International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 10, pp 551-558, October-2023

https://doi.org/10.47001/IRJIET/2023.710072

NetNexus: Intelligent Network Monitoring Device for Small Business

[1]Thriyashi Silva, [2]Helani Herath, [3]Vihan Udawela, [4]Kavindu Rathnayake, [5]A.N.Senarathne, [6]Dinithi Pandithage

[1,2,3,4,5,6]Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka
Authors E-mail: [1]thriyashiradika@gmail.com, [2]helaniherath@gmail.com, [3]vihan.udawela@gmail.com, [4]kavinduviraj3@gmail.com, [5]amila.n@sliit.lk, [6]dinithi.p@sliit.lk

***Abstract* - Network security of small businesses is pivotal with evolving technology. Presently most small business users tend to use the Internet as their main source of marketing, storing sensitive information, and performing business processes. Compared to large companies, small business networks often neglect information security. Most of the existing solutions that integrate advanced technologies like machine learning are not cost-effective for the target audience. To address the security concerns, this research proposes an innovative approach named 'NetNexus' an intelligent network monitoring device built on a Raspberry Pi module. The proposed model provides an Access control list that is integrated with a malware detector with 85% accuracy, a brute force attack detection, and a user-behavior-based quota allocation model with 91% accuracy. Furthermore, the NetNexus device is integrated with a device security mechanism with a Physically Unclonable function to protect the device from tampering and cloning. NetNexus is targeted at small businesses that neglect the need for network security due to the high expenses in implementation. This product has focused on a cost-effective approach which will cost around 1500 USD, and alternatively, it is efficient as only one single device is capable of securing the network and managing the performance.**

Keywords: Intelligent Network Monitoring, Malware Detection, Access Control List, Quota Allocation, Brute Force, Network Security.

I. INTRODUCTION

The small business networks have advanced significantly over the years evolving from peer-to-peer connections to complex cloud-based systems. Digital transformation in business has increased productivity as well as brought difficulties such as the requirement of strong security measures to guard against security threats. Furthermore, there are various security tools and software at industrial levels ensuring information security, but the most important fact to highlight in small businesses is that these implementations are expensive and require skillful technical teams to operate. As a motive for addressing these problems, NetNexus presents a cost-effective, plug-and-play intelligent network monitoring device. With the rapid development of technology, attackers are trying to attack systems and networks a lot. Large-scale businesses can secure their systems and networks using high technical security mechanisms which most of the time need high power and technical knowledge to handle, and they are costly. But small-scale businesses lack the financial capacity to invest in advanced technological solutions. That makes small-scale businesses more vulnerable. Therefore, it is important to have a solution that is suitable for the scope, budget, and time of the business.

The proposed NetNexus is built on a Raspberry Pi board with four main components which aid in securing and optimizing small business networks. Access Control List with Malware Detection Model is developed to deliver protection to the network by identifying malware utilizing both machine learning and rule-based techniques. A smart honeypot is deployed to detect brute-force attacks. A user-behavior-based quota management module is deployed to allocate quota to users. The device security mechanism is used to protect NetNexus from tampering by utilizing physically unclonable functions.

This paper layout is as follows: Section 2 discusses the related work covering the main components of the proposed NetNexus Device. The implementation of the device is addressed in Section 3. Section 4 is dedicated to discussing the results obtained and finally, Section 5 concludes the work and discusses the potential future improvements.

II. LITERATURE REVIEW

A) Access Control List with Malware Detection System

The utilization of firewalls for the purpose of implementing an access control list is a significant technical approach to safeguarding network security. The complexity of network architecture continues to evolve, and the dynamic nature of business requirements and access control policies are frequently changing which leads to inaccurate policies. These

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 551-558, October-2023
https://doi.org/10.47001/IRJIET/2023.710072

can result in many security problems, particularly the unavailability of services due to wrong access policies[1].

An important factor in computer systems is the malware detection. According to [2] , presently attackers commonly employ polymeric malware, which is a form of malicious software that shows a dynamic nature, consistently altering its identifiable attributes. This adaptive behavior is intended to deceive detection mechanisms that rely on conventional signature-based approaches.

According to [3], a framework for malware has been created named Cuckoo. This sandbox is utilized for the purpose of conducting dynamic analysis on files. The main purpose of this sandbox is to generate a comprehensive report that is derived from the activities of the runtime system. The utilization of Chi Square and random forest techniques is prevalent in the process of selecting significant features. The results indicate that the decision tree classifier has attained the highest level of accuracy compared to the other classifiers. For malware detection, Josh McGiff et al. [4] have taken into account a combination of hardware attributes and permission data. Combining these two types it has been determined that improves the model's performance.

B) Smart Bruteforce Honeypot

Password-based authentication mechanisms are the most common authentication mechanisms used to identify a specific user to the system or the network. This has led to password guessing and cracking becoming frequent attacks for attackers. Feldmier and Kard [5] found out that the password creation time is weaker than the time of password cracking. Tanvi Gautham and Anurag Jain[6] highlight that there can be several reasons for an attacker to do a brute force attack. Main reasons are to gain access unauthorized, to recover passwords in case of forgotten and to exploit the system or the network.

The models used to predict brute force attacks in NetNexus are Naive Bayes Model and NLP (Natural Language Processing) Model. Among them, the Naive Bayes Model is the main model. Naive Bayes model has a probability and statistical model based on classification algorithms. According to [7] the following equation is used in the Naive Bayes model to predict brute force attacks.

$$P\ (c\ |\ x)\ =\ (P\ (c\ |\ x)\ P(c)\ )/P(x) \qquad (1)$$

$$P\ (c\ |\ X)\ =\ P\ (x1\ |\ c)\ xP\ (x2\ |\ c)\ x\ \ldots xP\ (xn\ |\ c)\ xP(c)$$

The flexibility of the model makes it helpful for models with large data sets. Naive Bayes model can be used for binary and multiclass classification problems. Another advantage of using the Naive Bayes model is, it does not need to train a lot and can be used to discrete and continuous data.

In aspects of network security, honeypots are used to lure attacks to an isolated environment where the attacker thinks he is in the real network or system. The dynamic deployment of honeypots has been researched by Warren Z Cabral, Craig Valli, Leslie F Sikos, and Samuel G Wakeling[8]. They have identified machines with their high-level features like IP addresses, MAC addresses, and TCP Stack fingerprint details. The honeypot chosen for this component is Cowrie honeypot which is a medium interaction honeypot. It does not have a high connection with the actual system, and it is more similar to the actual environment. Cowrie can act as a SSH server where it is using an approach called emulation. According to [8], Cowrie has an emulated PowerShell session and changeable file system where the attackers can easily lure to the honeypot. The honeypot is hosted in an AWS EC2 instance. Hosting a honeypot in the cloud has many advantages the honeypot can be hosted separately without having any interaction with the actual environment, the honeypot does not need any physical space, and since it was hosted in a free instance there is not any cost. According to Rahul Saini and Rachna Bail[9], the EC2 instance permits clients to have virtual machine environments in the cloud. For this component, an Ubuntu virtual machine is used as well.

C) User-behavior-based Quota Management

The internet is one of the most critical components presently. Due to the rapid expansion of digitally enhanced content and the increasing demands of Internet computing in recent years, users frequently perceive a lack of adequate bandwidth to fully meet their requirements. However, the underlying issue lies with management's failure to identify specific bandwidth-consuming and unproductive applications.

Internet quota management has become important as it aids in managing Internet usage while reducing network congestion and increasing the speed of Internet services. The research [10], compares and analyzes two schemes which are quota scheduling (QS) and time-of-day pricing (TDP). The evaluations have shown that TDP has the best improvement in managing the quota. Dynamic pricing methods emphasize the behavior and the system dynamics. Specifically, it considers the user population as the average of user arrivals and departures [11].

In static quota allocation, the data is allocated to users manually by an administrator at the beginning of the month [12]. And in dynamic quota allocation, the system will monitor Internet usage and allocate the data dynamically [13]. The research NetNexus has come up with a machine learning model to automatically allocate quota according to the user's behavior.

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 551-558, October-2023
https://doi.org/10.47001/IRJIET/2023.710072

D) Device Security with Physically Unclonable Function (PUF)

According to [14], Lightweight Authentication protocols can be used to secure IoT devices using Physical Unclonable Functions. Minimal user intervention can be implemented to IoT devices using PUF Authentication. The methodology can be used to solve problems with existing security controls in PUF and to solve vulnerabilities in PUF. There is a direct interaction with the challenge response protocol. Server hacking attacks and machine learning-based attacks can be resolved from this methodology.

John Chou [15] was able to solve the problem of securing the key of PUF at the hardware level. There can be so many vulnerabilities that happen if the security key of the PUF was unable to be secure. Another problem John Chou understood was, even though we can implement security patches to software, it is hard to implement them in hardware. So, it is important to implement correct security measures for hardware at the initial level. PUF-based, which is an integration of PUF, and anti-fuse can be used to enhance security at the initial level of hardware. Therefore, it is important to secure the root key for the system.

According to [16], IoT devices are important for people to do their work in less time and efficiently. Since most IoT devices are based on hardware devices they can be victims of hardware-based attacks. Although there have been protocols built for the security of hardware, they are complicated due to the authentication process. The proposed methodology uses key exchanging protocols and lightweight authentication protocols to solve those problems.

Breaken[17], identifies recently proposed key agreement schemes which are used in IoT. Since they are vulnerable to man-in-the-middle attacks and replay attacks they have identified an alternative methodology for the key agreement schemes. The proposed key agreement scheme is an efficient one and two IoT devices can communicate with each other when they are connected to the same authentication server. Furthermore, this alternative provides identity-based authentication and repudiation.

III. METHODOLOGY

An overview of the proposed NetNexus device is showcased in Fig 1 and it consists of the four main components. The main objective of NetNexus is to develop a network monitoring device to secure the network while ensuring the performance and quality of service. The proposed product has the capability of denying malicious traffic and allowing legitimate traffic while creating a web client for users, as well as luring attackers while accurately detecting their information, increasing the performance since users have been allocated for data quotas smoothly, and making sure the device is working attack freely by having PUF authentication. Every functionality has a different development approach. Our device’s ideal hardware is Raspberry Pi 4 running the Raspbian operating system. Python was used as the most ideal for machine learning development requirements along with other crowd sources, packages, and libraries.

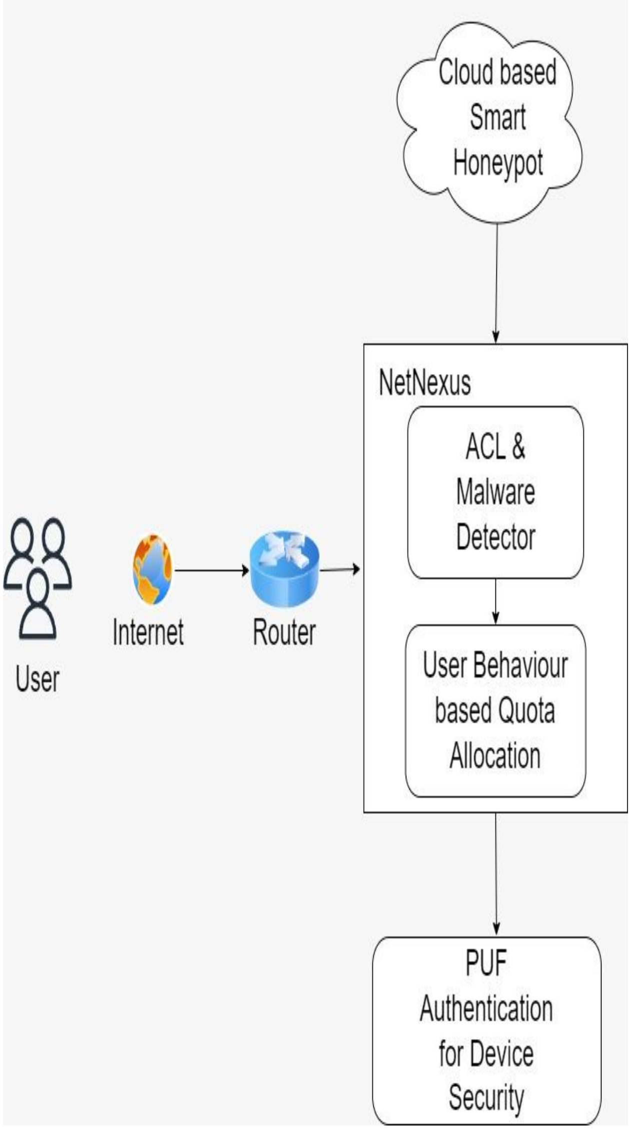
Cloud based Smart Honeypot
NetNexus
ACL & Malware Detector
User Behaviour based Quota Allocation
User
Internet
Router
PUF Authentication for Device Security

Figure 1: Proposed NetNexus Architecture

A) Access Control List with Malware Detection

By offering a novel strategy that incorporates malware detection capabilities, the methodology used in this research study aims to solve the limits of conventional Access Control Lists (ACLs) and thereby improve network security. ACLs and firewalls were thoroughly reviewed throughout the project's first phase, which revealed how inadequate they were at efficiently fending off the evolving threat of malware.

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 551-558, October-2023
https://doi.org/10.47001/IRJIET/2023.710072

A state-of-the-art ACL framework that focuses on integrating malware detection algorithms into the access control paradigm was developed to close this gap. There were several crucial milestones in the implementation process. To better assist customers, a user-friendly console with a streamlined process was created. This console featured a safe login page and a user dashboard that was a central location for many features.

A key component of the console, the dashboard, offered customers a variety of insightful data. Accurate bandwidth measurement was made possible by real-time bandwidth monitoring made possible by packet transfer to a speed test server. Users were able to successfully optimize their network consumption because the technology tracked data usage patterns and controlled allocation details. Additionally, a special function allowed users to manually block particular websites, helping to create a customized and secure network environment.

Admins were provided with improved site-blocking tools through a customized Domain Name System (DNS) approach. Banned site URLs were made to point to the local host by host file manipulation, blocking access to illegal content. Using an advanced machine learning method, the project's malware detection component was strengthened. A feedforward neural network was the method of choice, and it was quite successful in identifying and reducing malware risks.

The 'netifaces' library from Python was used in the technical implementation to enable easy access to network interfaces and to make it easier to gather pertinent network data. Utilizing Electron.js and React JS, the client application was painstakingly designed to ensure cross-platform compatibility and give users a user-friendly and responsive experience. The client-side application and the backend components may now communicate easily with the establishment of a reliable Python Flask backend server.

B) Smart Bruteforce Honeypot

The Smart Brute Force Detector of NetNexus is a cloud-based honeypot that can deviate attackers from real networks and notify users about brute force attacks. The model is connected through VirusTotal to check the attackers' IP address. The final results will be notified to the user and they will be saved in logs for future purposes. The component is hosted in the AWS cloud platform using an EC2 instance to avoid performance loss in Raspberry Pi. Fig 2 is the flow diagram of the Smart Brute Force Detector.

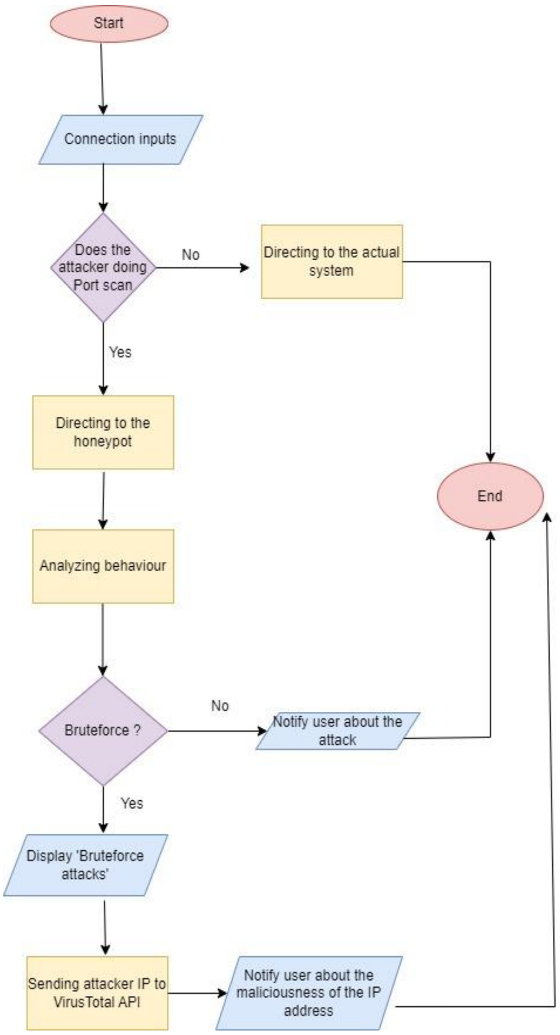
Start
Connection inputs
Does the attacker doing Port scan
No
Directing to the actual system
Yes
Directing to the honeypot
End
Analyzing behaviour
Bruteforce ?
No
Notify user about the attack
Yes
Display 'Bruteforce attacks'
Sending attacker IP to VirusTotal API
Notify user about the maliciousness of the IP address

Figure 2: Brute force Honeypot

The honeypot which is used for this component is the Cowriehoneypot which is a medium interaction honeypot. The honeypot is customizable hence it can be configured as a real system according to the user's preferences. Cowrie honeypot is able to detect port scans, attacker tools, password guessing, Secure Shell (SSH) based attacks, etc. Cowrie is generating a log which was obtained as the input for the Brute Force prediction model and VirusTotal based model.

Natural Language Processing (NLP) and Naive Bayes models are used to predict brute force attacks in this component. The Naive Bayes model is good for security-related predictions while the NLP model is used for the feature engineering part. Popup notification will be sent to the user when a brute force attack is detected from the honeypot.

VirusTotal is an online platform to detect the maliciousness of IP addresses and domain names. It can detect various malware such as viruses, worms, trojans, etc. NetNexus has been connected to VirusTotal and got their API to check the maliciousness of the IP addresses that have been detected by the honeypot. This function helps to identify whether the attacker has already been identified as a malicious one.

C) User-behavior-based Quota Management

The objectives of developing a user-behavior based Quota Management are as follows; capturing user behavior and validating the user credentials via the captive portal of NetNexus, using the machine learning model to dynamically allocate quota, finding appropriate data sets to train the model, completing the training phase, allocating quota through a router, testing, and integration phase, as well as updating the allocated quota amount in the interface. The dataset utilized to train the model was found from the UCI which is a machine learning repository. From the data set, the identification of the customers who visited frequently and rarely were able to depict. The machine learning model used is Recurrent Neural Network (RNN). The reasons that prove that RNN was the most appropriate are as follows: it can handle sequential data, considers the current and previous inputs, and can be used for a series of data predictions. The algorithm used is the Long short-term memory (LSTM), this algorithm has the capability of classifying and learning sequential data to depict an output. This methodology is used to classify the user's data usage according to a certain time. Then the machine learning model that was built will predict the amount of quota for each user. And finally, the allocated quota will be updated in the main database of NetNexus. The flow of this system is also shown in Fig 3.

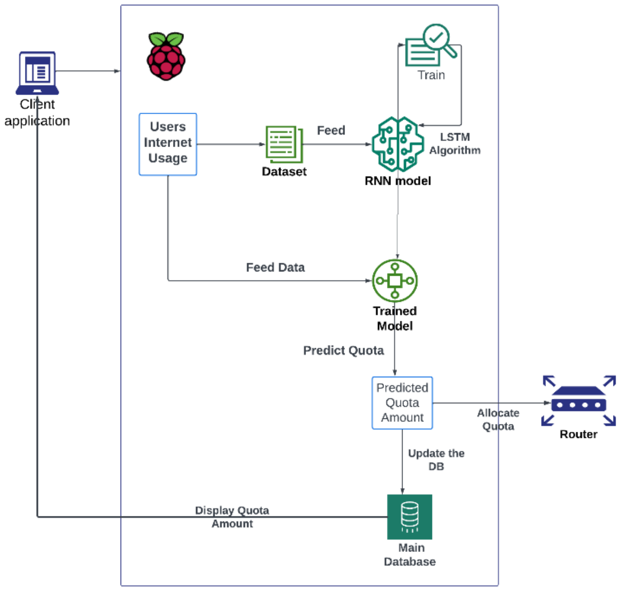

Figure 3: Quota Allocation System Diagram

The User ID which is extracted from the verification portal, user's data usage amount, date and time are the types of data that will be utilized to train the machine learning model. One of the limitations identified is that this system is for internal network users only.

D) Device Security with Physically Unclonable Function (PUF)

The main objective of this component is to secure the data stored in the proposed NetNexus as shown in Fig 4. Physical Unclonable Functions (PUFs) are a category of hardware security. These have utility in a diverse range of security-oriented applications, encompassing tasks such as the production of cryptographic keys, authentication protocols, and the implementation of anti-counterfeiting measures.

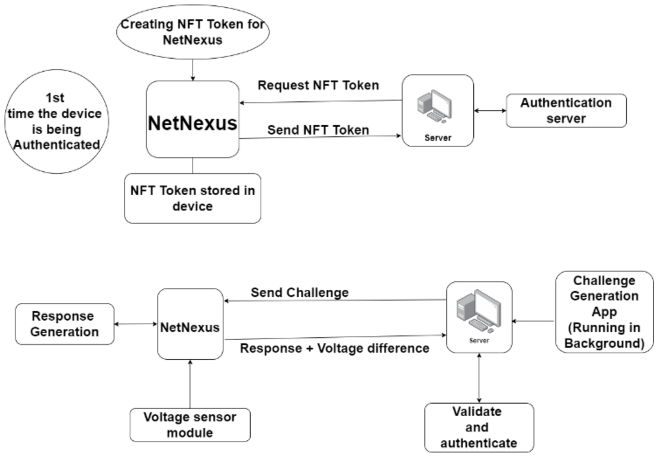
Creating NFT Token for NetNexus
1st time the device is being Authenticated
NetNexus
Request NFT Token
Send NFT Token
Server
Authentication server
NFT Token stored in device
Response Generation
NetNexus
Send Challenge
Response + Voltage difference
Server
Challenge Generation App (Running in Background)
Voltage sensor module
Validate and authenticate

Figure 4: Device Security Architecture

Here in this component when an attacker tries to clone the NetNexus device, it will be nearly impossible since PUF authentication is done using the Voltage fluctuations of the Raspberry Pi, and an authentication server is used. When the attacker tries to connect to the Network with the cloned device the initial authentication will be done via the NFT Token. The NetNexus device has an NFT Wallet with an NFT Token stored in it. And the authentication server will verify if the Device is a legit device or not. Next, a challenge-response will be carried out using the physical unclonable functions of the NetNexus device. Here, the challenge will be generated from the authentication server and then sent to the NetNexus Device (Raspberry Pi). Next, a response will be generated using the sent challenge and the voltage fluctuations of the device during the generation of the response and then it will be verified using the authentication server. Since the voltage fluctuations are not the same in the device it will be varied. Therefore, it will be ensured that the device is authentic, and will provide a unique and tamper-proof device.

IV. RESULTS AND DISCUSSIONS

The device NetNexus provides the user with a single device that has the capability of monitoring the network intelligently while securing and elevating the performance.

This is compromised with four advanced features into a single Raspberry Pi. NetNexus has a user-friendly environment that the target market can easily understand and interact with. The testing stage was performed separately for each module.

A) Access Control List with Malware Detection

The technique utilized in this component is coupled with theoretical innovation with real-world application. The initiative set the path for more effective defense against the changing environment of cyber threats by incorporating malware detection capabilities into ACLs, which not only contributed to personalized network security but also improved protection against them. The ML model has an accuracy level of 85% in detecting malware.

B) Smart Bruteforce Honeypot

In this component, the accuracy level of the function was sufficient to deliver to the target audience. Apart from the accuracy level, to ensure the reliability of the function, the confusion matrix, recall score, precision function, and F1 score have also been tested. The following Fig 5 depicts the accuracy and reliability of the Smart Brute Force Honeypot.

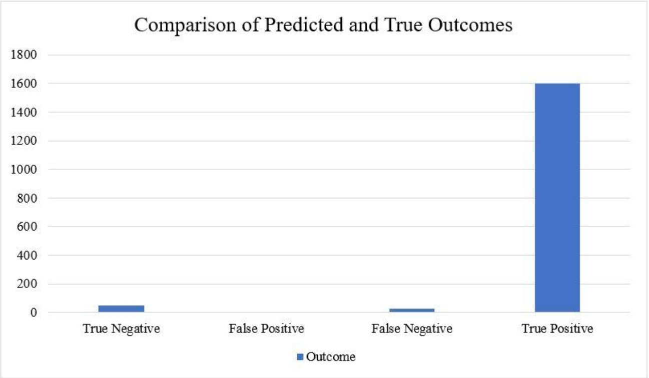

Figure 5: Accuracy Level of the smart honeypot

C) User-behavior-based Quota Management

The accuracy rate of the RNN model is 91%, this proves that this model accurately analyzes the user behavior and predicts the correct quota amount. The following Fig 6 depicts the model loss function, which gradually comes down as shown in the graph, this shows that the accuracy of the model increases and the loss of the model loss function decreases.

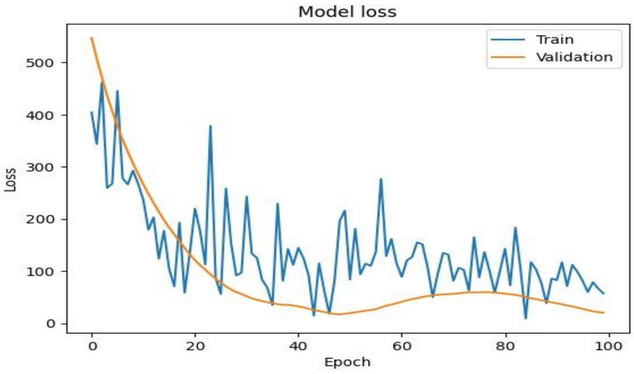

Figure 6: Model Loss Chart

D) Device Security with Physically Unclonable Function (PUF)

A CentOS 7 machine is used as the authentication server and the challenge generation will be done using a python virtual environment in the device. Using Python code, the challenge response will be validated. Using a Python virtual environment, an authentication client is created. The response will be generated using the challenge that was sent by the authentication server and the voltage fluctuation that occurred during the generation of the response to the challenge.

V. CONCLUSION AND FUTURE WORK

NetNexus has provided a solution for a number of issues that small-scale businesses face like unprotected networks and internet quota problems for their employees and customers. This proposed device is a low-powered, cost-effective intelligent network monitoring device that consists of four major functionalities. The access control list with a malware detection model integrated with machine learning has an accuracy rate of 85%. The smart honeypot detects brute force attacks using the Natural Language Processing model with a precision rate of 1.000% and a higher rate of true positives. The quota management module was developed using a Recurrent neural Network model with an accuracy level of 91%. NetNexus has its own device security mechanism which detects device cloning and tampering attacks. Apart from the security, NetNexus makes sure that the user interface is easy to access and the performance of the device. The technologies that were utilized in NetNexus are machine learning models like RNN and Feed forwarding, and the Cowrie honeypot to detect brute-force attacks.

Our proposed device, NetNexus, can be used highly efficiently to identify cyber-attacks at small-scale business networks while ensuring data integrity and confidentiality, according to the findings of tests that were successfully conducted. In conclusion, the suggested affordable network security solution, which also offers ease of setup, has huge

ISSN (online): 2581-3048
Volume 7, Issue 10, pp 551-558, October-2023
https://doi.org/10.47001/IRJIET/2023.710072

market potential. Future work will be to improve the device according to the feedback of the target market. As the next phase of this device, it is planned to expand the honeypot to detect DDOS attacks, to integrate the device with artificial intelligence to allocate quota and user assistance which would provide more business value for the product.

REFERENCES

[1] Z. Chen, W. Zhang, and G. Wang, “Design of algorithm for verifying correctness of firewall access control list,” 2020, pp. 190–193. doi: https://doi.org/10.1109/CCET50901.2020.9213149.

[2] U. V. Nikam and V. M. Deshmuh, “Performance Evaluation of Machine Learning Classifiers in Malware Detection,” in 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), pp. 1–5. doi: https://doi.org/10.1109/ICDCECE53908.2022.9793102

[3] K. Sethi, R. Kumar, L. Sethi, P. Bera, and P. K. Patra, “A Novel Machine Learning Based Malware Detection and Classification Framework,” in 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), pp. 1–4. doi: https://doi.org/10.1109/CyberSecPODS.2019.8885196.

[4] J. McGiff, W. G. Hatcher, J. Nguyen, W. Yu, E. Blasch, and C. Lu, “Towards Multimodal Learning for Android Malware Detection,” in 2019 International Conference on Computing, Networking and Communications (ICNC), pp. 432–436. doi: https://doi.org/10.1109/ICCNC.2019.8685502.

[5] L. Bošnjak, J. Sres, and B. Brumen, Bruteforce and dictionary attack on hashed realworld passwords. 2018, pp. 1161–1166. doi: https://doi.org/10.23919/MIPRO.2018.8400211.

[6] T. Gautam and A. Jain, “Analysis of brute force attack using TG — Dataset,” in 2015 SAI Intelligent Systems Conference (IntelliSys), pp. 984–988. doi: https://doi.org/10.1109/IntelliSys.2015.7361263.

[7] H. TAŞÇI, S. GÖNEN, M. A. BARIŞKAN, G. KARACAYILMAZ, B. ALHAN, and E. N. YILMAZ, “Password Attack Analysis Over Honeypot Using Machine Learning Password Attack Analysis,” Turkish Journal of Mathematics and Computer Science, Aug. 2021, doi: https://doi.org/10.47000/tjmcs.971141.

[8] W. Cabral, C. Valli, L. Sikos, and S. Wakeling, “Advanced Cowrie Configuration to Increase Honeypot Deceptiveness,” 2021, pp. 317–331. doi: https://doi.org/10.1007/9783030781200_21.

[9] R. Saini and R. Behl, An Introduction to AWS—EC2 (Elastic Compute Cloud). 2020, pp. 99–102. doi: https://doi.org/10.15439/2020KM4.

[10] S. i Chu and S. c Chang, “TimeofDay Internet Access Management: Virtual Pricing Vs. Quota Scheduling,” in 2006 10th IEEE Singapore International Conference on Communication Systems, pp. 1–6. doi: https://doi.org/10.1109/ICCS.2006.301386.

[11] M. Siew, D. Cai, L. Li, and S. Quek, “Dynamic Pricing for ResourceQuota Sharing in MultiAccess Edge Computing,” IEEE Transactions on Network Science and Engineering, vol. 7, no. 4, pp. 2901–2912, doi: https://doi.org/10.1109/TNSE.2020.3003051.

[12] Internet Access Gateway (IAG), SANGFOR, 2022. https://www.sangfor.com/Uploads/File/IAM%20Bandwidth%20Man (accessed Aug. 01, 2023).

[13] GuruhAryotejo and M Mufadhol, “Static and dynamic alliance: the solution of reliable internet bandwidth management,” vol. 1217, no. 1, pp. 012126–012126, May 2019, doi: https://doi.org/10.1088/1742-6596/1217/1/012126.

[14] S. Yoon, B. Kim, and Y. Kang, “Multiple PUFbased lightweight authentication method in the IoT,” in 2021 International Conference on Information and Communication Technology Convergence (ICTC), pp. 1198–1200. doi: https://doi.org/10.1109/ICTC52510.2021.9620972.

[15] Root of Trust: PUFrt | PUF-based Security IP SolutionsPUFsecurity, Mar. 02, 2022. https://www.pufsecurity.com/products/pufrt/#:~:text=PUFrt%E2%80%99s%20Hardware%20Root%20of%20Trust%20is%20ushering%20in (accessed Aug. 13, 2023).

[16] Y. Zheng, W. Liu, C. Gu, and C. H. Chang, “PUFBased Mutual Authentication and Key Exchange Protocol for PeertoPeer IoT Applications,” IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 4, pp. 3299–3316, doi: https://doi.org/10.1109/TDSC.2022.3193570.

[17] A.Braeken, “PUF based authentication protocol for IoT,” Symmetry, vol. 10, p. 352, Aug. 2018, doi: https://doi.org/10.3390/sym10080352.

Citation of this Article:

Thriyashi Silva, Helani Herath, Vihan Udawela, Kavindu Rathnayake, A.N.Senarathne, Dinithi Pandithage, "NetNexus: Intelligent Network Monitoring Device for Small Business" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 10, pp 551-558, October 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.710072

****