ISSN: 0974-8571 Vol.13 No. 2 December, 2021

International Journal of Computational Intelligence in Control

IoT Security Orchestration using a Chatbot

S. Balachandar [1]

[1] Research Scholar, VTU-RC , Department of MCA CMR Institute of Technology , Bengaluru - 560037

Dr. R. Chinnaiyan [2]

[2] Professor & Supervisor , VTU-RC , MCA , Department of MCA CMR Institute of Technology, Bengaluru - 560037

Abstract. Chatbot-assisted Internet of Things (IoT) security analysis is the practice of employing chatbots to improve and simplify IoT device security examination. The main data sources for analysis are the logs obtained from various IOT gateways, IOT Edge servers, or direct devices. The chatbot, also known as a virtual assistant, is a cloud-based program that provides ongoing assistance to users, the command center team, or regulators regarding the device's metadata, current data pattern, and characteristics. Additionally, the bot can use some of the models that have been implemented on top of this log data to determine if there has been any unusual data pattern or packet flooding within a specified time frame.

Keywords: *chatbot, GPT, TCP packet, IOT Edge, Cloud Platform, Distributed Database, IOT Gateway, anomaly detection*

I. INTRODUCTION

Forbes estimates that by the end of 2024, there will be over 207 billion linked IoT devices, increasing the value of the industry for both consumer and enterprise IoT.[1]. There are a lot of security risks, and vulnerabilities can appear in many places, including hardware, software, web apps, and network services. Protecting vital infrastructure and private information utilized by the IoT network is of utmost importance. A centralized storage is necessary for the data emitted from various networks, such as the Edge[2] or the Cloud, in order to gather logs and messages. This not only aids in protecting these devices from unknown attacks, such as Denial of service or Botnets, but also helps in understanding the security risk from various device end points and network zones. With the use of a chatbot[4], we can retrieve the information we need to anticipate security breaches by answering questions based on the logs we've stored centrally. This will allow us to see things from a security standpoint and gather any extra data that may be necessary. Our next step is to examine new models that we implemented into our chatbots. Section II examines the reviews written by various writers and published by various publishers. These reviews discuss the importance of chatbot models, as well as the risks and models associated with IoT security, in identifying attack patterns and preventing vulnerabilities.

We shall go over the overall strategy and its parts in Section III. In Section IV, we will discuss

the use of chatbots to implement various IoT security models. Problem relevance, high-level solution mapping, and recognized challenges will be discussed in Section V. In section VI, we will discuss the worries and factors to think about during deployment. We shall offer our suggestions and last thoughts in section VII.

***Problem statement*:** In a large-scale centralized IoT system (like a Smart City or any industry using thousands of devices), it is always challenging to gather and query various metadata details of IoT devices and analyze the data points relevant to security.

2. Data Collection, Data Analysis

Data Collection: Following data sources are identified for analyzing the IOT Security Analysis.

1. Device Logs: In Kaggle[12] There are few preprocessed dataset for network based intrusion detection system in IoT Devices kept Ultrasonic Sensor with Arduino and NodeMCU used to monitor the network and collect the network logs. Node MCU with ESP8266 wifi module was used to send data to the server through WiFi network.
2. Network Logs[13]: Sourced from https://www.kaggle.com/datasets/jacobvs/ddos-attack-network-logs The dataset contains around 2,100,000 labelled network logs from various types of network attacks.The types of network attacks logged are: UDP-Flood, Smurf, SIDDOS, HTTP-FLOOD, & Normal traffic

Data Analysis: We took the sample of 477,426 records with different Frame lengths from blank to 3484 bytes.

3. PROPOSED Algorithm:

 # Load Data

file_path = 'networklogs_data.csv' df = pd.read_csv(file_path)
 # Choose Columns for Anomaly Detection columns_for_zscore = ['frame.len', 'tcp.len'] # Calculate Z-Scores for Selected Columns

z_scores = calculate_z_scores(df, columns_for_zscore) # Define Z-Score Threshold
z_score_threshold = 3.0 # Identify Anomalies
anomalies = detect_anomalies(z_scores, z_score_threshold) # Display or Process Anomalies
print_anomalies(df, anomalies)

Output:

Anomalous rows:

	frame.number	frame.time	frame.len	eth.src	\
2006	2007	123746764912430	705	167275820076079	
2021	2022	123746855970591	1424	167275820076079	

2022	2023	123746858594675	683	167275820076079
2033	2034	123746942019478	646	37559677479822
2038	2039	123747031838176	596	37559677479822
...	...	...	...	...
103277	103278	130251445329932	705	167275820076079
103289	103290	130251530436142	839	167275820076079
103298	103299	130251601625670	646	37559677479822
103314	103315	130251673137431	741	37559677479822
103315	103316	130251674075789	394	37559677479822

Isolation Forest: This technique generates decision trees at random to identify and isolate outliers. Isolation Forests are different from conventional decision trees in that they don't use attribute values to divide data; instead, they pick a feature at random and then choose a split value between the feature's minimum and maximum values.

Average Path Length in Trees: The key point is that outliers should have shorter path lengths on average. An outlier is more likely to be a data point that is rapidly isolated (with a short path length).
o Tree Ensemble: These randomly selected trees are assembled into an ensemble via the Isolation Forest algorithm. An anomalous score is calculated for each data point by calculating the mean distance traveled by each tree.
An anomaly score is derived from the average path length; a lower score suggests a higher possibility of an anomaly.The ability to scale: isolation When compared to other algorithms, such as k-means or density-based approaches, forests are less affected by the data's dimensionality and can scale to big datasets.

Output: Anomalous rows:

	frame.number	frame.time	frame.len	eth.src \
8	9	123722750551006	288	167275820076079
17	18	123722861240051	288	167275820076079
26	27	123722974641916	288	167275820076079
36	37	123723084768869	288	167275820076079
45	46	123723199143947	288	167275820076079
...	...	...	...	
			...	
313056	60624	125272303944069	207	101775857169652

313061	60973	125282281866981	204	102351086050890
313062	60630	125271955934775	207	101714973144285
313065	59007	125223126994653	221	98706942643254
313069	61636	125302234764452	199	103580253098726

	eth.dst	ip.src	ip.dst	ip.proto	ip.len
8	87971959760497	1921680121	192168035	6.0	274.000000
17	87971959760497	1921680121	192168035	6.0	274.000000
26	87971959760497	1921680121	192168035	6.0	274.000000
36	87971959760497	1921680121	192168035	6.0	274.000000
45	87971959760497	1921680121	192168035	6.0	274.000000
...	...	...	...	...	...
313056	167275820076079	1255436009	1921680121	6.0	193.522087
313061	167275820076079	1299743963	1921680121	6.0	190.960211
313062	167275820076079	1250746317	1921680121	6.0	193.793244
313065	167275820076079	1019047804	1921680121	6.0	207.189999

...

4. Recommendations:

- Further Analysis: Continue to analyze the characteristics of the anomalies detected by each model to gain insights into their performance.
- Fine-tuning: Consider fine-tuning the models or exploring other anomaly detection algorithms to optimize performance further.

We have integrated this model output into our Database and it helps us to get the abnormal rows based on the score value.

5. Implementation of Chatbot

The chatbot is crucial for end-users (e.g., monitoring users, command center personnel, etc.) to query the IOT log using natural language, as described in Figure 1 in section III. With the help of the chatbot's virtual assistant, users may gather all the information they need regarding a device's or network's security, as well as any other data they specify, within a predetermined time frame.

For this, we looked to Open AI's Generative Pre Training article.[15][16] Using user input, the GPT models (particularly GPT-3) may produce writing that sounds natural. The Application Programming Interface (API) that OpenAI provides is the usual means by which users engage with GPT models. Natural language understanding and generation can be integrated into apps, goods, or services using the API, which allows developers to access GPT.

For our chatbot need, we will be utilizing "Completion Calls," which use the "Text Completion" type. The most recent iteration of OpenAI's GPT series is GPT-3, which stands for Generative Pre-trained Transformer 3 [16]. Using deep learning techniques, GPT-3 is able to comprehend input text and produce output that is remarkably similar to human writing. To improve its conversational abilities, it employs a method known as "Reinforcement Learning from Human Feedback (RLHF)". Listed in Table 2 below are a few of the models included in GPT3.5[17]

Model	Description	Context Window	Training Data Upto
gpt-3.5-turbo-1106	Latest Model supports JSON and returns a maximum of 4096 tokens	16,385 Tokens	September 2021
gpt-3.5-turbo	Currently points to gpt-3.5-turbo-0613	4096 Tokens	September 2021

Table2: GPT 3.5 Models

Considering our use case which needs to query the IOT Data from YugabyteDB or Elastic Search, we used a LangChain[18] Model. LangChain is a emerging solution which helps us in the querying process and extracting information from YugabyteDB. With its advanced NLP algorithms, it easily converts the user input queries to structured query language.

Figure 2 – Data Flow of Chatbot using langchain Framework.

Data flow seq#	Operations/Tasks	Component Involved
1	App written in Python which takes IOT Security questions of an device or network	Python/Flask

2	Pass the data through Langchain which invokes OpenAI call	Langchain Framework installed from https://python.langchain.com/docs/get_started/introduction
3	YugabyteDB stores the application data	YugabyteDB

Table 3: Component Mapping with data flow sequence

Output:

> *The network frame details whose normality values is "0" are: frame_number 98312, frame_time 1.30201E+14, frame_len 62, eth_src 8.7972E+13, eth_dst 1.67276E+14, ip_src 192168035, ip_dst 1921680121, ip_proto 6, ip_len 48, tcp_len 0, tcp_srcport 63987, tcp_dstport 80, value -99.*

6. Conclusion

In conclusion, this research has delved into the realm of IoT security orchestration, introducing the innovative concept of a "Cybernetic Sentinel" empowered by Generative AI. Through extensive investigation and experimentation, several crucial findings have been unearthed.

7. REFERENCES

1. Applying Chatbots to the Internet of Things: Opportunities and Architectural Elements, DOI:10.14569/IJACSA.2016.071119, November 2016International Journal of Advanced Computer Science and Applications 7(11)
2. Forbes - https://www.forbes.com/sites/bernardmarr/2023/10/19/2024-iot-and-smart- device-trends-what-you-need-to-know-for-the-future/?sh=2aab520f7f34
3. G Sabarmathi, R Chinnaiyan (2019), Envisagation and Analysis of Mosquito Borne Fevers: A Health Monitoring System by Envisagative Computing Using Big Data Analytics, Lecture Notes on Data Engineering and Communications Technologies book series (LNDECT, volume 31), 630-636. Springer, Cham
4. G. Sabarmathi and R. Chinnaiyan, "Investigations on big data features research challenges and applications," *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, 2017, pp. 782-786.
5. G. Sabarmathi and R. Chinnaiyan, "Reliable Machine Learning Approach to Predict Patient Satisfaction for Optimal Decision Making and Quality Health Care," *2019 International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, India, 2019, pp. 1489-1493
6. Hari Pranav A;M. Senthilmurugan;Pradyumna Rahul K;R. Chinnaiyan , "iot and Machine Learning based Peer to Peer Platform for Crop Growth and Disease Monitoring System using Blockchain," 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-5, doi:
7. M Swarnamugi, R Chinnaiyan (2019), IoT Hybrid Computing Model for Intelligent Transportation System (ITS), Proceedings of the Second International Conference on Computing Methodologies and

Communication (ICCMC 2018), 802-806.

8. M. Caroline Viola Stella Mary, G. Prince Devaraj, et al., "Intelligent Energy Efficient Street Light Controlling System based on IoT for Smart City," IEEE Xplore
9. M. Swarnamugi ; R. Chinnaiyan, "IoT Hybrid Computing Model for Intelligent Transportation System (ITS)", IEEE Second International Conference on Computing Methodologies and Communication (ICCMC), 15-16 Feb. 2018.
10. M. Swarnamugi; R. Chinnaiyan, "Cloud and Fog Computing Models for Internet of Things", International Journal for Research in Applied Science & Engineering Technology, December 2017.
11. Partha Pratim Ray, ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope".
12. Prasannjeet Singh, Mehdi Saman Azari1, Francesco Vitale, Francesco Flammini1, Nicola Mazzocca, Mauro Caporuscio, Johan Thornadtsson, Using log analytics and process mining to enable self-healing in the Internet of Things, DOI:10.1007/s10669-022- 09859-x
13. S. Balachandar, R. Chinnaiyan (2019), Internet of Things Based Reliable Real-Time Disease Monitoring of Poultry Farming Imagery Analytics, Lecture Notes on Data Engineering and Communications Technologies book series (LNDECT, volume 31), 615- 620. Springer, Cham
14. S.Balachandar , R.Chinnaiyan (2018), A Reliable Troubleshooting Model for IoT Devices with Sensors and Voice Based Chatbot Application, International Journal for Research in Applied Science & Engineering Technology,Vol.6,Iss.2, 1406-1409.
15. S.Balachandar , R.Chinnaiyan (2018), Centralized Reliability and Security Management of Data in Internet of Things (IoT) with Rule Builder, Lecture Notes on Data Engineering and Communications Technologies 15, 193-201.
16. S.Balachandar , R.Chinnaiyan (2018), Reliable Digital Twin for Connected Footballer, Lecture Notes on Data Engineering and Communications Technologies 15, 185-191.