Characterizing Volumetric DDoS Attacks in IoT Network Traffic

George Mason University
AIT-580 | Prof. Dr. Alejandro Álvarez

Jonathan Wilson
George Mason University
Fairfax, Virginia
jwilso87@gmu.edu

Abstract—This paper explores cybersecurity within Internet of Things (IoT), focusing specifically on the challenge posed by Distributed Denial of Service (DDoS) attacks. The analysis utilizes a dataset consisting of real-world network traffic captures obtained from IoT devices. The main objectives are to characterize DDoS attacks in IoT traffic, shedding light on their behavior as it relates to volume, time and connection traits. A background on the subject matter and related works on the topic are also provided as well as details of the methodologies and technology used to inform the processes taken during research. The findings underscore the significance of understanding DDoS attack patterns for effective detection and mitigation strategies. Lastly, the paper discusses the implications of the research findings and as well as its challenges and limitations.

Keywords—IoT, Cyber Security, DDoS, AWS, Cloud, Python, R, SQL, Spark, Big Data

I. Introduction

The internet of things (IoT) has the potential to revolutionize the world. It's considered one of the six disruptive technologies with the potential to impact society well into the future according to the National Intelligence Council (NIC) of the United States [1]. Today IoT is used in critical life support systems that rely on real-time data essential of life preservation in healthcare. In industry IoT devices are used as monitoring and metering systems to deliver insights for power grids and manufacturing. Sensors placed at major traffic congestion locations or intersections can be used to monitor traffic flow. Cars and homes connected to phones and other mobile devices collect and analyze data to increase energy efficiency. Big name companies such as Google, Intel, IBM, AWS, and Cisco already have their own IoT products[1]. While IoT is revolutionizing the world it does not come without its limitations and problems. Possibly the most important issue to address is cyber security and in particular one of the most notorious cyber attacks distributed denial of service (DdoS) [7].

The overall goal of this research project is to characterize what DDoS attacks look like in IoT traffic thus better understanding their behavior. By understanding the nature of DDoS attacks cyber security professionals can learn to prevent these attacks in real time by implementing detection and mitigation solutions such as machine learning (ML). The main objectives and flow of this paper will be to:

1. **Characterize DDoS by traffic volume** - Determine what anomalies exist in traffic based on the volume indicators such as packets and the number of bytes.
2. **Characterize DDoS traffic temporally** - Pinpoint the time those anomalies happened, how often they happened and for how long they were sustained.
3. **Characterize DDoS by connection behavior** - Narrow down or isolate the anomaly in question and characterize it by looking at IP connection, port connection and connection history.

This paper is divided up into several sections. Background will lay a foundation for the content with the aim of making the subsequent sections more accessible. Related work will look at current research on the topic of IoT security and DDoS attacks. Dataset explains the dataset that was used for analysis. Technology Overview explains the workflow and tech stack used for this project, Analysis dives deeper into the findings and analysis methods, the Discussion provides an overview of the findings and lastly Future Work lists several next steps that can be taken.

II. Background

While a cyber attack on your personal computer might be a local catastrophe a cyber attack on power grid systems could cause a regional or even global catastrophe. One famous state sponsored cyber attack called Stuxnet targeted PLCs[2] on an Iranian nuclear facility wrecking havoc and setting Iran back from its nuclear objectives for many years [9].

IoT devices have many potential vulnerabilities. For any IoT system there are typically three layers:

1 IoT product offerings of Google [2], Intel [3], IBM [4], AWS [5], and CISCO [6].

2 PLC - programmable logical controller while not typically connected to the internet leverages similar technology and protocols as does IoT devices which raises the concern that if a standalone device can be attacked IoT devices connected to the internet have a higher risk profile and security is of an even greater concern [8].

1) perception layer such as physical sensors sensing the environment

2) network layer where all the telecommunications happen

3) application layer or the side the user interfaces with [10]. IoT devices also have a wide range of connectivity methods ranging from near field communications (NFC) to cellular and satellite networks in space each with their own protocols [10]. The lack of protocol standardization of this technology presents numerous problems although there are many efforts to fix this [11]. With all this complexity (heterogynous mixture of layers and non-standard protocols) leaves IoT devices vulnerable to a wide range of cyber attack vectors [7]. Each type of device has their own attack surface which makes cyber security in IoT such a complex endeavor but for the purpose of this paper we will focus on the network layer and DDoS cyber attacks.

The most common cyber attacks are Denial of Service and Distributed Denial of Service (DDoS[3]) [7]. The goal of these attacks are to disrupt the normal function of a targeted system. A DDoS attack starts by scanning for vulnerable devices (open ports for example) and sending information back to a database. Loaders then create new bots by connecting to these vulnerable devices and download a malware onto them. Once the malware is on the device these devices become part of a botnet awaiting commands from a command and control server (C&C) [12]. Once recruited to the botnet these devices are used to do damage to higher value targets. Mirai, a botnet mainly comprising IoT devices (IP cameras and home routers), unleashed massive DDoS attacks in 2016 infecting hundreds of thousands of IoT devices exploiting the many vulnerabilities of IoT devices and used them to target higher value assets such as Twitter, Spotify, Netflix, and GitHub [13]. This attack demonstrated the risks inherent in IoT ecosystems and also how IoT devices are being used for attacks as well as being attacked [14]. Mirai exceeded 600 Gbps in volume [14] aggregated from hundreds of thousands of devices placing it in the volumetric type of DDoS attack which is one of three types of DDoS attacks.

There are 3 main types of DDoS attacks :

1. Volumemetric (flooding) attacks
2. Protocol attacks
3. Application attacks [15]

This paper will focus on the first, volumetric DDoS attacks and attempt to characterize them. This kind of task is often implemented by threat hunters. Part of threat hunting is to understand an attacker's tactics, techniques, and procedures or TTPs which involves looking at patterns in network behaviors [16]. One benefit of understanding these TTPs is being able to create ML models that can be used for early malware detection specific for IoT devices. Algorithms such as anomaly detection could pick up indicators of compromise or even serve as an early detection system that could provide valuable information to security analysts without them having to manually look at the data themselves. While these types of analyses have been done before in various research (see Related Works) there is still much to be done in IoT security.

III. Related Work

Several studies have endeavored to classify and understanding network behaviors though research in IoT cyber security is still new and presents several limitations and barriers to overcome as mentioned here [17]. The Mirai attack, as previously discussed, serves as a motivation for further study. The research done by [14] sheds light on Mirai's emergence and growth and tracks Mirai's growth, composition, and evolution, pinpointing the timing of infections and the botnet's activity periods. The study also outlines Mirai's phases of infection, from rapid initial spread to eventual decline, providing insights into the temporal dynamics of DDoS attacks. These two points demonstrate the importance of studying the temporal traits of DdoS attacks hence objective 2. Connection history, IP and port connections are deemed as import for the NetSight platform which captures packet histories and uses them to understand network behavior [18]. This project uses PCAP files fomatted with Zeek conn.log which contains packet information which is different but similar to NetSight (see Dataset). And lastly this survey [17] highlights the importance traffic volume has as part traffic classification.

Multiple studies have worked toward creating machine learning solutions to classify and detect malicious activity on networks. In [19] a neural network based approach for detecting DDoS attacks was implemented. The model was based on an multi-layer perception and used to classify various patterns distinguishing between what was normal vs what was considered a threat [19]. Deep learning and HetIoT (Heterogeneous Internet of Things) environments are looked at in [20]. They explore the usage of CNN as a good solution for classifying introducing HetIoT-CNN IDS mentioned in [21] which is lightweight design (low complexity) compared to conventional IDSs [20]. The main point here is that work is that researchers are using machine learning solutions to classify network traffic in IoT systems all of which require identifying patterns in network traffic.

Several limitations and hurdles are noted in a survey of traffic classification in IoT networks [17]. In the article they mention many of the studies they surveyed the data were synthetically generated in a lab leading to issues with correct representation of real world scenarios [17]. The survey also admits classification research in IoT is still an emerging area and will require more work in the future to tackle real world threats posed by malware on IoT systems [17]. When it comes to DDoS specifically [15] mentioned the idiosyncratic nature of DDoS makes it difficult to distinguish between legitimate packets and malicious packets due to how the packets are aggregated which makes characterizing DDoS such a difficult task [15]. Hence, it is imperative to acknowledge the two

3 The distinction between DoS and DDoS is the word distributed. Distributed means the attacker uses multiple sources as vectors of attack as opposed to just one.

primary limitations inherent in this study. For one, the dataset used (see Dataset) was created in a lab setting. And two, trying to characterize DDoS attacks is a difficult. Despite these limitations we will proceed anyway and consider solutions to these limitations at a future date.

IV. Dataset

The dataset for this research comes from the iot23 dataset created at CTU University's Stratosphere Laboratory funded by Avast Software published in 2020 [22]. To simulate a real world scenario it consists of 20 malware and 3 benign traffic captures on several different IoT devices running between 2018-2019. Each scenario contains several million records and totaling together roughly 325 million records. The attacks originated from Raspberry Pi devices and the victim hosts consisted of 3 different IoT devices:

- Philips HUE smart LED lamp
- Amazon Echo home intelligent personal assistant
- Somfy smart doorlock

The 23 captures (called scenarios) come from Zeek conn.log files which were extracted from the original PCAP[4] files [24]. This dataset also labels and names the type of malware which aids the research of network behaviors with the motivation to develop machine learning algorithms that can be used in cyber security applications [22].

The following tables show the data features with their associated data types. Table 1 contains original iot23 dataset. Table 2 contains the transformation of the iot23 dataset which consists of some name changes, dropped features, a modification to the timestamp, and an additional feature that classifies traffic volume (low, normal, high and very_high) for each feature containing packets and bytes. The method used for obtaining traffic_volume_category will be explained in the analysis section.

Table 1

Original Dataset (iot23)			
Feature	**Data Type**	**Feature**	**Data Type**
field-ts	Ratio	resp_bytes	Ratio
Uid	Nominal	conn_state	Nominal
id.orig_h	Nominal	local_orig	Nominal
id.orig_p	Nominal	local_resp	Nominal
id.resp_h	Nominal	missed_bytes	Ratio
id.resp_p	Nominal	history	Nominal
proto	Nominal	orig_pkts	Ratio
service	Nominal	resp_pkts	Ratio
duration	Ratio	orig_ip_bytes	Ratio
orig_bytes	Ratio	resp_ip_bytes	Ratio
tunnel_parents	Nominal	label	Nominal
detailed_label	Nominal		

Table 2

Transformed Dataset			
Feature	**Data Type**	**Feature**	**Data Type**
timestamp[a]	Interval	resp_bytes	Ratio
connection_uid	Nominal	conn_state	Nominal
source_ip	Nominal	local_orig	Nominal
source_port	Nominal	local_resp	Nominal
destination_ip	Nominal	missed_bytes	Ratio
destination_port	Nominal	conn_history	Nominal
conn_proto	Nominal	orig_pkts	Ratio
app_proto_service	Nominal	resp_pkts	Ratio
conn_duration	Ratio	orig_ip_bytes	Ratio
orig_bytes	Ratio	resp_ip_bytes	Ratio
malware_name	Nominal	traffic_volume_category[b]	Ordinal
label	Nominal		

a. Timestamp (field-ts) was converted from unix time to use 24 hour based time.

b. This is a new feature created to categorize traffic based on traffic volume. See analysis.

Some features deserve some explanation while others are obvious. Anything with the prefix source means the originator and destination means the responder. Source is where the connection started and the destination is who the source is connecting to. Anything with prefix orig means the originator (source) while resp means the responder (destination). Proto represents protocol, conn for connection and pkts for packets. Some very import features to define are conn_history and orig_bytes. conn_history is the connection history which is a string characters each character with a meaning. Essentially theses strings of characters tells us the behavior of the connection i.e. a history of what happened at an instant in time. orig_bytes represents the number of bytes that came from the source or originator. These two features will be important in working toward our objectives stated above. For more information on the features see the Zeek logs documentation [25].

4 PCAP files stand for packet capture and are files containing network traffic packets obtained by a network analyzer [23]

V. Technology Overview

This project used both local computing resources (Figure 1) as well as resources in AWS cloud (Figure 2). Several processes were conducted during the project namely, data processing, data cleaning, statistical analysis, EDA, and the creation of an a relational database using a schema. A workflow for the project is explained in the next several paragraphs.

The workflow starts with downloading the data from the iot23 dataset website onto the local machine. As the dataset was sufficiently large (43 GB) Apache Spark (Pyspark) was used for most of the data processing and cleaning where processing the entire dataset was required. The decision for choosing Pyspark was for two reasons. First, scalability into the future so if there was ever a need to expand or enrich the dataset with more data it could be done without having to worry about memory limitations of pandas. Second is due to differences in memory consumption. Pyspark's data transformations are lazy loaded into memory and thus do not store all the data in memory as pandas does [26], [27]. There are options to scale large datasets with pandas [28], however, for this project it was decided to stick with Spark. Apache Spark is also designed specifically for big data projects and in particular distributed data so if there was a desire to scale the data analysis in the cloud it could easily be done through the usage of something like AWS EMR[5] or Google Dataproc.

The etl.py or pyspark script created smaller data aggregations saved as CSV files to a reports directory that were then used downstream by other programs. One such program is a load_sql.sql script which creates database and a table provided with a data schema. This load_sql.sql script was used in a database in AWS RDS a relational database and a connection established between a local MySQL workbench instance AWS RDS. Within MySQL various queries were run against a particular report such as the connection history report. The other program that uses the reports directory extensively is the Rstudio program. Rstudio is where most of the EDA, analysis and data visualizations were created. Rsudio provides code blocks and a library called reticulate that enables R and Python to be run in the same script so both R and Python we used during the project in the same environment. R was used for creating the visualizations for traffic flow by volume analysis. Python, pandas and networkx were used for many of the time series and network graphs.

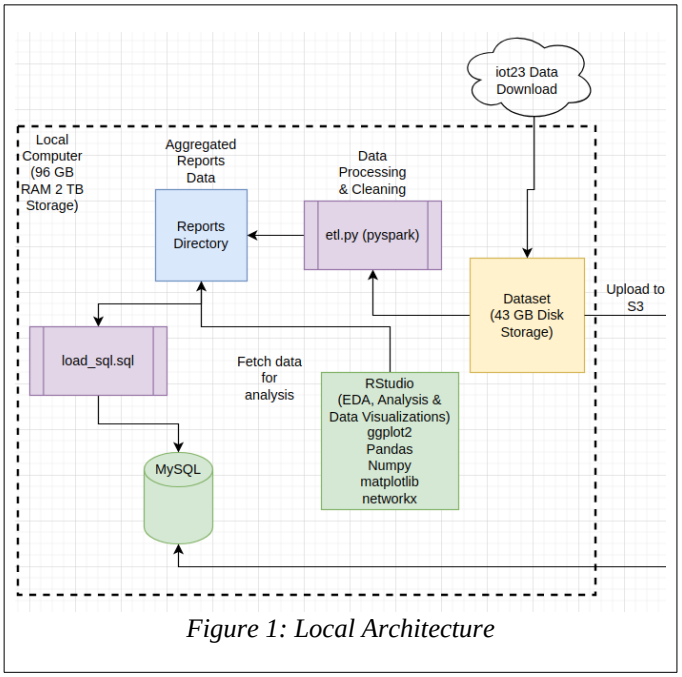

Figure 1: Local Architecture

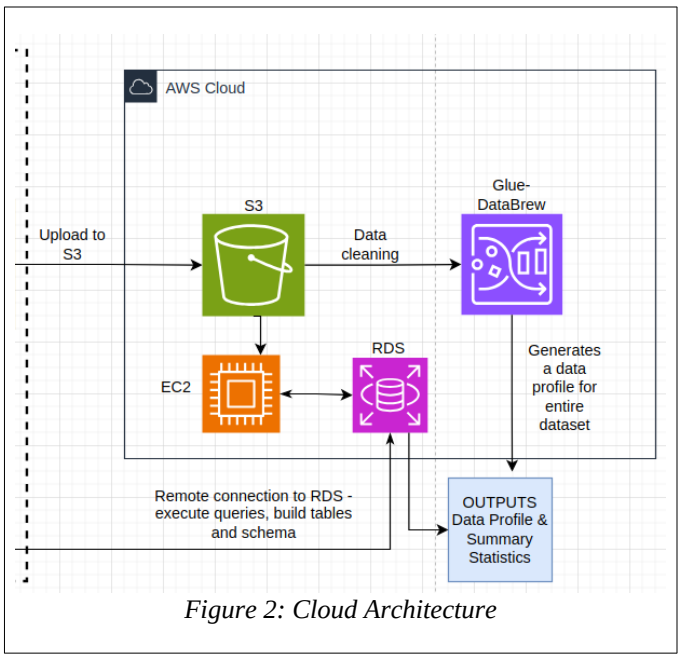

Figure 2: Cloud Architecture

In the cloud portion of this workflow the iot23 dataset is uploaded to an AWS S3 bucket and then ingested into AWS Glue using a Glue script that preforms data cleaning (using the same pyspark code to do the data cleaning) and processing. After that process is complete a CSV file of the cleaned data is saved to S3. Once that was complete Glue DataBrew created a dataset which was used to create a data profile. This data profile provides most of the summary statistics for the project such as the box-plots and correlation matrix.

5 AWS EMR is a web service specifically designed for distributed big data environments such as Hadoop and Apache Spark. Google Dataproc is Google's flavor of a similar web service.

VI. Analysis

This analysis starts by looking at some summary statistics of the data to explore and understand the data better. Next we dive into the main objectives of this research project by looking at traffic volume for inbound and outbound packets and bytes with the objective of discovering anomalies in the data. Next we narrow in on that particular anomaly by zooming in on when that anomaly happened and look at how long it was sustained. Lastly we look at other characteristics of those particular events by analyzing connection behaviors. For each part of this analysis we will be comparing benign network traffic and malicious traffic labeled as DDoS.

A. Summary Statistics

Looking at Figure 5 and 5 we can get an idea of the overall distribution of malware and benign traffic types. There is an overall total of about 325 million records in this dataset. Out of those 30.86 million of those records are labeled as benign while the rest (324.61 million) are labeled as malware. Focusing in on Figure 4 again the traffic types we are interested for this analysis are boxed in red (Benign and DDoS). DDoS traffic accounts for 6% (19.54 million records) of the traffic. Figure 3 shows a correlation matrix for all the numeric type features. It's not surprising that there is an almost perfect correlation between bytes and packets for both resp_ip_bytes and resp_ip_pkts and origin_ip_bytes and orig_pkts (.99 & .81 respectively). This is because the more bytes you are sending will require more packets to send those packets[6]. It's interesting to note that there is no correlation in almost all of the other features. This is most likely due to the vast amount of zeros in the data as will be seen in the next section of this analysis. Figure 6 shows the distributions of the connection protocol used. For this analysis we will be primarily interested in TCP traffic and which consists of 99% of the traffic in this dataset and all of it for DDoS labeled traffic. Figure 7 shows the destination ports with port 22 being the most popular destination. Table 3 is a table of summary statistics for all the traffic volume related features which will be useful for when we do the volume analysis. Something to point out are the large max values compared to the mean and the large standard deviations relative to the mean indicating data are widly spread out. The last visualizations in this summary statistics section are Figures 8 and 9 which look a little closer at the spread of the data using boxplots. The boxplots show the median as either all zeros or really low in comparison with the larger outlier values. We will address some of these issues in the traffic volume analysis.

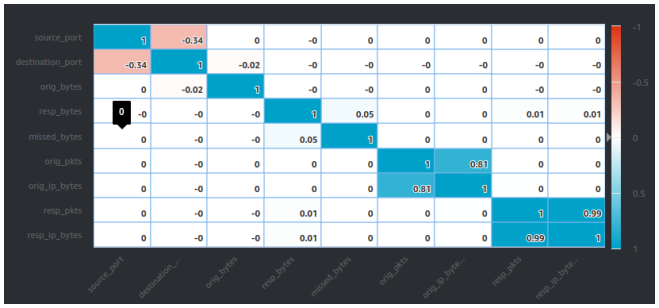

Figure 3: Correlation Matrix

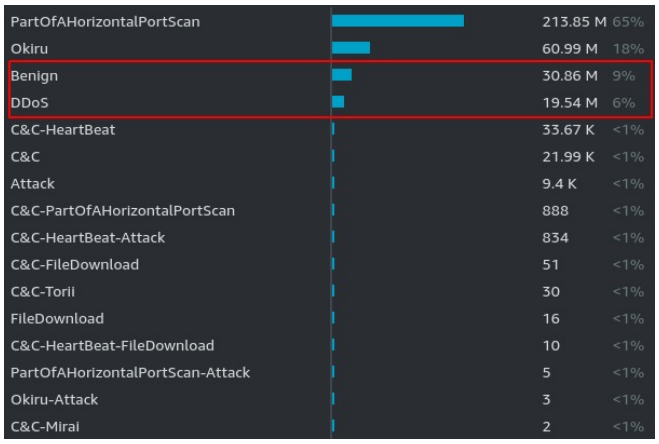

Figure 4: Network Traffic Distribution

Figure 5: Benign vs Malicious Traffic Distribution

Figure 6: Connection Protocol

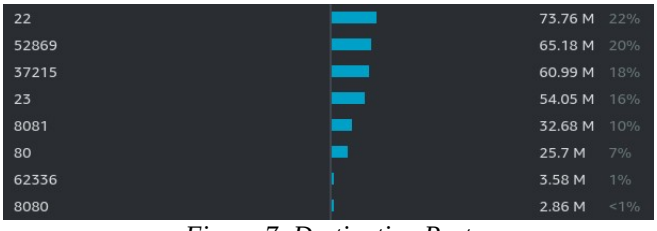

Figure 7: Destination Port

6Packet switching involves the packaging of data (bytes) into smaller packets that can then be routed across a network. For more info read about MTUs in this RFC [29]

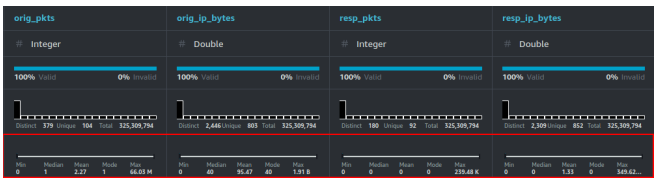

Figure 8: Boxplot For Traffic Flows

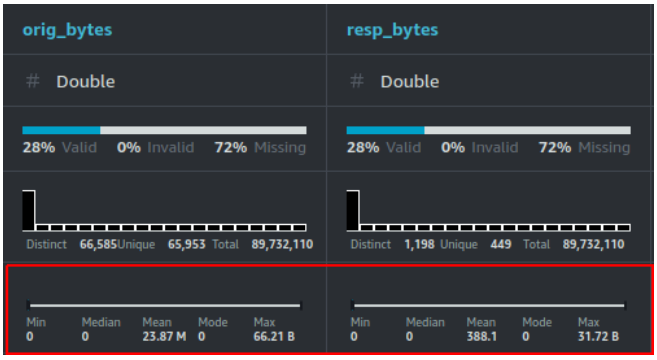

Figure 9: Boxplot For Traffic Flows-2

Table 3

summary	orig_bytes	resp_bytes	orig_pkts	orig_ip_bytes	resp_pkts	resp_ip_bytes
count	89732239	89732239	325309923	325309923	325309923	325309923
mean	23865207.28	388.11	2.27	95.47	0	1.33
stddev	902225805.56	3355943.33	4855.63	190715.96	13.41	19398.62
min	0	0	0	0	0	0
max	66205578295	31720511878	66027354	1914793266	239484	349618679

B. Anomalies In Traffic Based On Volume

For this part of the analysis we look at classifying the traffic by low, normal, high and very high to determine any kind of pattern or anomaly. Determining these classifications was through the use of Algorithm 1 below. The reason for this algorithm was there needed to be some kind of threshold or cutoff value for determining the various traffic flow volumes that leveraged that data itself and not some arbitrary value. Although it could be argued that this algorithm is arbitrary, however, it at least gives us something to base our categories off of. And in reality calculating network flow thresholds are highly dependent on the traffic under investigation. There are methods that exist such as CMU's Timothy Shimeall [30] but to keep things simple Algorithm 1 was decided.

1. Calculate quantiles q25, q75, q95 for traffic volumes:
   - If q75 equals 0:
     - Exclude all zero values from the calculation
     - Recalculate q25, q75, q95 from the non-zero values

2. For each traffic volume entry in Column A:
   - If the entry is less than or equal to q25:
     Set the category to "low"
   - Else if the entry is less than or equal to q75:
     Set the category to "normal"
   - Else if the entry is less than or equal to q95:
     Set the category to "high"
   - Else:
     Set the category to "very_high"

Algorithm 1:

The algorithm shows how each traffic type was placed into categories based on quartiles. Traffic types included originator and responder packets and bytes (6 types or features). Then there were 4 different traffic volume categories. The low category captures the 25th percentile values. Normal consists of values between the 25th and 75th percentile. High is anything between the 75th and 95th percentile. Lastly, very high takes on the very tip of the data at the 95th and above. As mentioned earlier there are many values that are zero and the overall spread of the data are large given the large standard deviation compared to the means. To overcome this the algorithm checks if the 75th quartile is zero and proceeds to exclude all zeros from the data. The dataset is transformed using this algorithm creating new features with the extension traffic_volume_category. For example, resp_pkts_traffic_volume_low represents the low category for responder packets. And so on. See Table 4.

Table 4

Determined Thresholds			
Feature	**Q25**	**Q75**	**Q95**
orig_bytes	90	22,005,606,846	38,687,204,445
resp_bytes	48	233	2079
orig_pkts	1	2	2
orig_ip_bytes	40	80	80
resp_pkts	1	2	16
resp_ip_bytes	40	146	2259

With a category in place we can low look at the various traffic types and classify them into low to very high traffic volumes and analyze any anomalies. The different colors should be noted here. Blue represents traffic that has been labeled as benign and red is for malware (DDoS).

Looking at all of the plots below we notice there is nothing out of the ordinary in terms of traffic. Benign traffic flow volumes seem to be sighted in every traffic type and every volume category with no significant spikes in malware. Figure 13 for originator bytes is the only real noticeable anomaly. The malware labeled DDoS is exceptionally high for both the high and very high traffic flow categories. Given that this number of bytes is coming from an originator indicates a possible attack using a volume based attack. With this information we have learned that DDoS can be identified by a high number of bytes coming from the orig_bytes feature of traffic flows. This makes sense given what DDoS attacks are. This could been seen as an anomaly in traffic flow as the number counts for high and very high category are very high compared to the rest of the data in originator bytes. Now that we have identified which feature contains the anomaly lets identify the timing of those DDoS events.

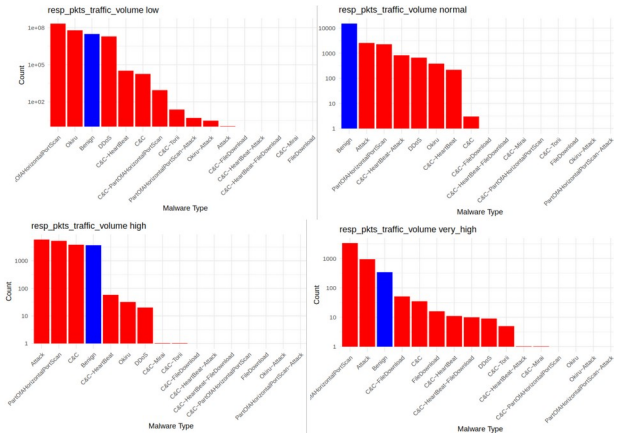

Figure 10: Responder Packet Volume Distribution

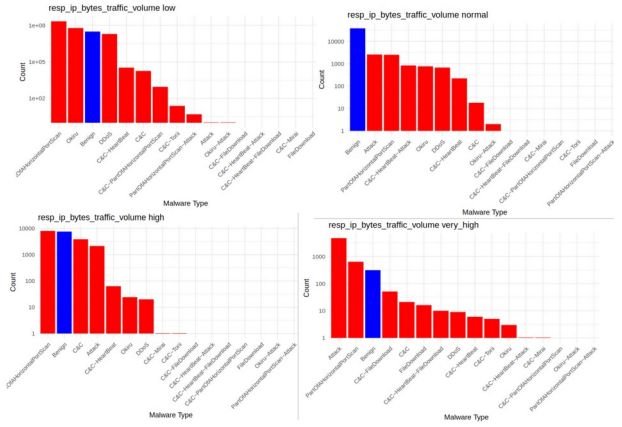

Figure 11: Responder IP Bytes Volume Distribution

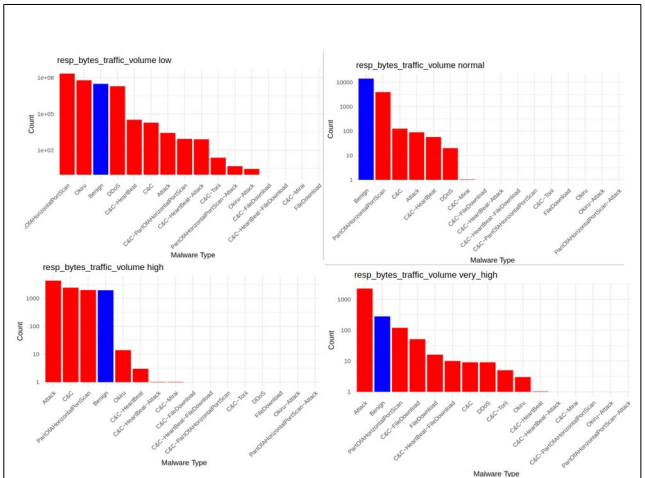

Figure 12: Responder Bytes Volume Distribution

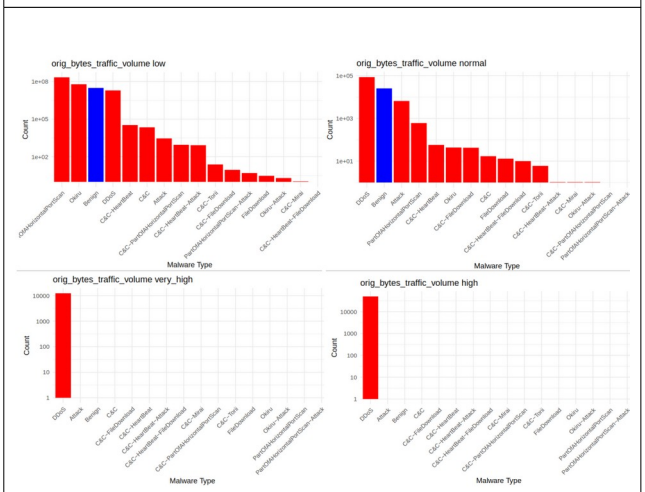

Figure 13: Originator Bytes Volume Distribution

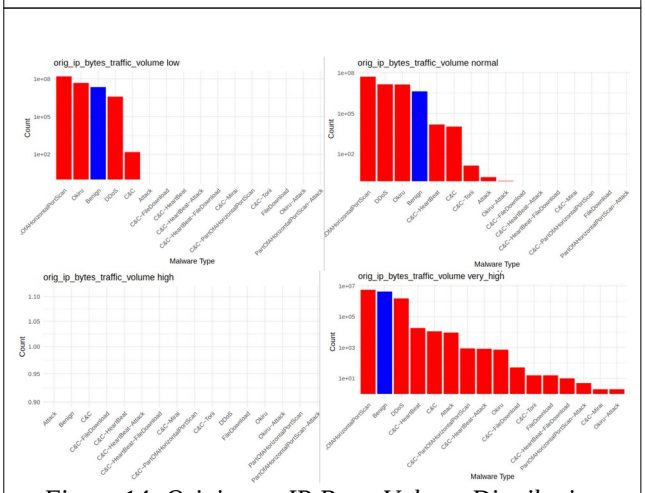

Figure 14: Originator IP Bytes Volume Distribution

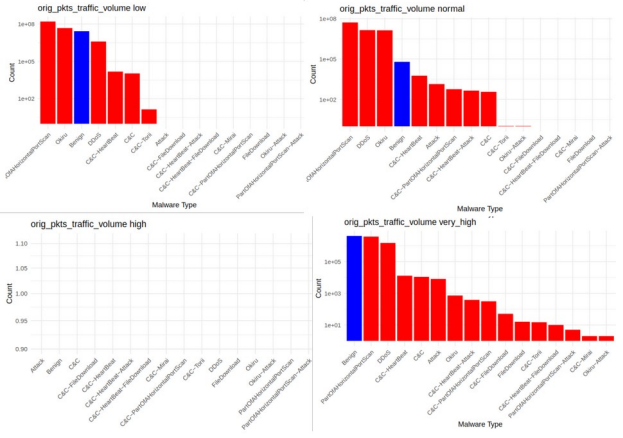

Figure 15: Originator Packet Volume Distribution

C. *Timing of Anomalies*

In this part of the analysis instead of looking at all of the malware we look at only DDoS types of malware as well as Benign traffic. In the previous section we discovered that orig_bytes was worth exploring more given that there was a suspected anomaly event, however, this time we analyze it temporally and by volume. The main objective here is to determine when DDoS happened, how long it happened, how often it happened, and how long they were sustained.

If we look at the time series plot in Figure 16 we can see about when DDoS events happen. The time spans over the entity of the data collection period 2018-05-09 to 2019-09-20 with a range of about 1 year and 4 months. If we use orig_bytes as a guide to narrow our investigation down we see that two notable events with large spikes which we will call attack 1 and attack 2. Attack 1 happens in Dec of 2018 while attack 2 happens the following month in Jan of 2019. Since it's hard to visualize what's happening we can zoom into that these particular points in time to see if we notice anything interesting. Already we can see that these spike events don't happen for too long. But just for how long exactly is what we would like to know next.

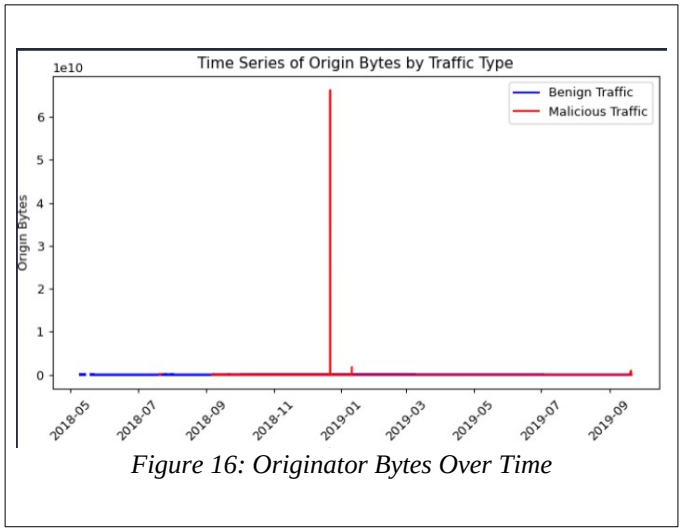

Figure 16: Originator Bytes Over Time

Let's start with attack 1. The conical looking plot in Figure 17 resembles a chaotic mess of seemingly random spikes in the number of originator bytes sent to responder device. It should also be noted that the originator bytes is scaled due to the large amount of bytes relative to the rest of the traffic. The spike lasts for about 15 min with the largest spike lasting only 10 min. The event begins on Dec 21st 2018 from 22:07:59 and goes to about 22:08:15. Cloudflare[7] Radar's DDoS Attack Trends for 2022 Q3 also seem to confirm that the majority (94%) of network-layer attacks based on volume of bytes sent end within 20 minutes [31].

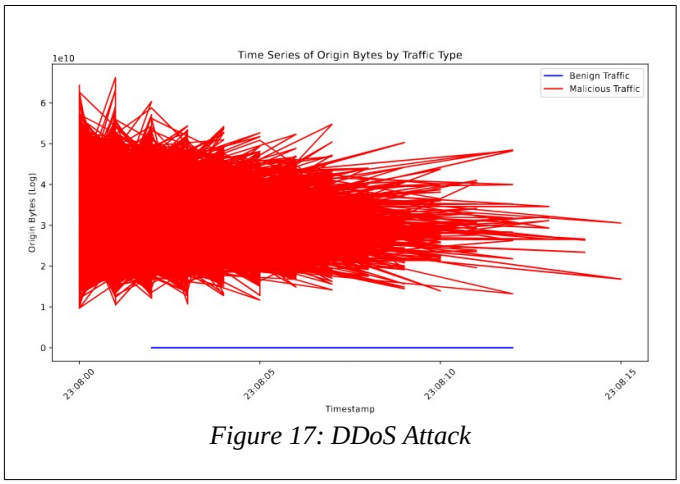

Figure 17: DDoS Attack

Attack 2 (Figure 18) is a bit different. On Jan 10th 2019 there is a spike that happens between the hours of 14:00 and 18:00 lasting 4 hours. This attack lasted a bit longer but as you recall in Figure 16 the spike in originator bytes is small in comparison to attack 1. The shape of the spike is also different as it exhibits a steady linear line rather than the more chaotic mess that attack 1 had. These differences reveal varied attacks

7 Cloudflare Radar is a tool for getting insights into global internet usage and is often used for cyber threat analysis and in this case it offers valuable information into global DDoS trends.

strategies employed by the attacker thus showing not all DDoS attacks are created equal.

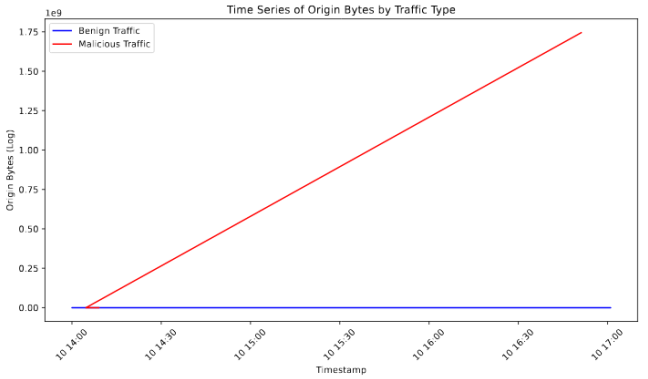

Figure 18: DDoS Attack #2

In the next section we will look at the various characteristics associated with these attacks to understand the behaviors exhibited during these very short time frames. In particular we want to understand the IP address network connections, port connections and the overall connection history and see what that can tell us about the nature of DDoS attacks.

D. Traits of DDoS Attacks

This section will be broken up into 5 subsections. The first two look at IP network for both attacks individually. The next two sections will look at port connections for both attacks individually. The last section will cover connection histories for all time, during attack 1 and for attack 2. Our goal for this section is to understand the traits or characteristics of DDoS in three domains, IP network, port connections and connection history.

1) Attack 1 – IP Network

Figure 19 shows the IP network graph during the time of attack 1. The IP address labeled as a blue node (192.168.1.196) is the originator of the attack and the white node (209.97.190.136) is the victim. The number 65535 is the number of unique connections the attacker established with the victim. Given that this graph only covers about 15 min 65535 is a pretty significant amount.

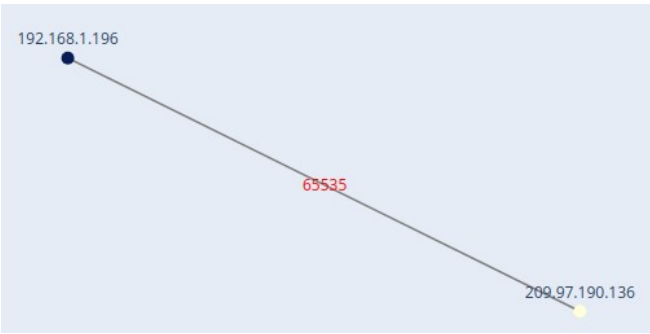

Figure 19: Attack 1 - IP Network

2) Attack 2 – IP Network

The IP network for Attack 2 is shown in Figure 20 and the same color scheme applies as did in attack 1. However, in this attack we have two clusters. Cluster one at the top left consists of only one unique connection to the white node. While the second cluster at the bottom right has 65803 connections.

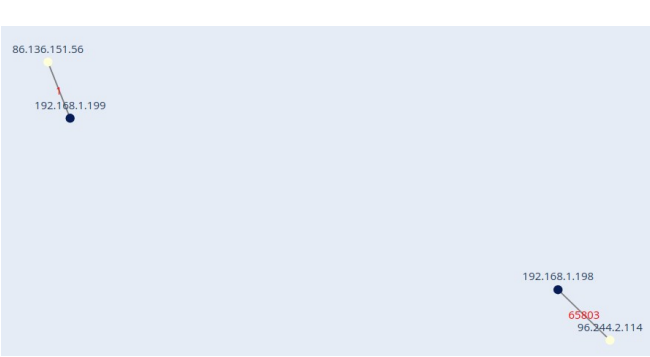

Figure 20: Attack 2 - IP Network

3) Attack 1 – Ports

Things start to get interesting from this point on. The network graph in Figure 21 shows many originator port connections (the blue nodes) to a single destination port 123. However, not all of the ports are displayed in this graph mostly because if we tried we wouldn't see any of the ports just a thick blue circle with port 22 in the middle. The actual number of unique ports was actually 65535. This is something to take

note of because the total number of available ports[8] is 65535. That means this attacker used all of the available ports to connect with victim node thus overwhelming them with traffic in a very short period of time.

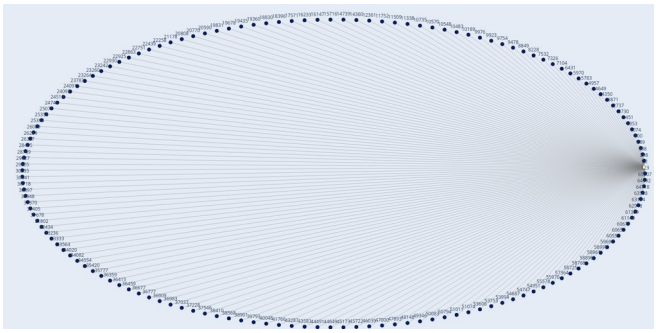

Figure 21: Attack 1 - Port Connections

4) Attack 2 – Ports

For attack 2 there is not much new that happens with port connection other than one of the victims in the attack had only one port connection.

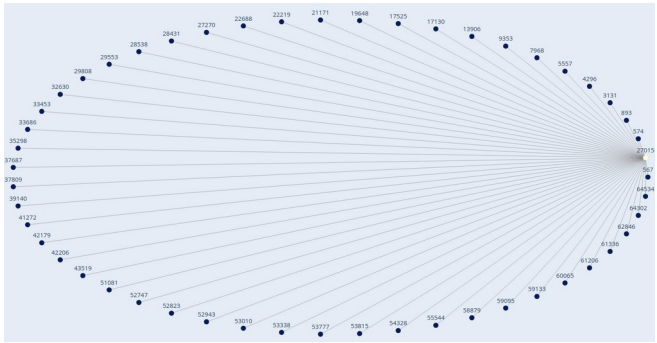

Figure 22: Attack 2 - Port Connections

5) Connection History

Zeek logs contain strings of letters or flags that represent the connection history of a given entity. These flags have special meanings and show what kind of behavior was exhibited. Figure 23 contains the unique value counts for connection history over the entire duration that network traffic was collect for traffic labeled as DDoS and benign. For background information, an upper case character comes from the originator while the lower case indicates it came from the responder. Overall the S had the most occurrences. According to Zeek docs [33] S is a SYN without the ACK[9] this could indicate a scanning campaign by the attacker though this would require further analysis. What's interesting to note for our purpose of volume based DDoS attacks are the flags D, DT, and DTT. Again referring back to Zeek's documentation of connection history flags a D flag represents packet with payload or data and T represents retransmitted packet. While a D can only be sent one time T can be sent many times. In fact T works on a logarithmic scale where a TT could represent an event that happens at least 10 times and TTT 100 times [33]. Looking at Figures 24 and 25 we can see these events take place for attack 1 and 2 for DDoS only. Benign traffic did not exhibit this behavior. The combination of DTT or even DT in such a short period of time again leads to the indication of a flooding attack. Since attack 1 was for for such a short period of time we see more occurrence of DTT while attack was more of a sustainment of just the flag D.

conn_history	total_count
S	13654866
C	3592643
I	2119476
D	105420
DTT	65534
Sr	660
^c	40
F	21
ShADdfFa	13
SI	10
ShADdattFfR	8
CCCC	6
ShADdfF	5
ShR	5
CCC	2
ShADdf	1
ShADdFaf	1
ShADdtatFfR	1
DT	1

Figure 23: Connection History Over All Time

conn_history	total_count
DTT	65531
DT	1

Figure 24: Attack 1 Connection History

8 Computer networking ports are like doors for the various different programs on your computer system. Each port allows connection that that program. Port 53 for example is used for DNS which converts your URL domains back and forth between the IP address and the string domain you see in your browser. For TCP/IP there is only a total number of 65,5535 such ports. See [32] for more information.

9 For more information on SYN and ACK see the RFC for TCP [34].

conn_history	total_count
D	65804

Figure 25: Attack 2 Connection History

VII. Discussion

The findings of this research offer valuable insights into the intricate landscape of cybersecurity in IoT devices, particularly in relation to the prevalence and characteristics of volumetric DDoS attacks. By analyzing the volume trends in the data we were able to identify certain anomalies associated with DDoS labeled malware. It was learned that originator bytes sent to the responder spikes significantly in relation to the rest of the traffic. Next we drilled down on those anomalies identifying certain events we called attacks 1 and 2. During attack 1 we noticed that the attack lasted only briefly (15 min) while attack 2 demonstrated a different pattern lasting 4 hours. This indicated differences in styles of DDoS attacks and yet another reason why focusing on one are of analysis is not enough to correctly classify all types of DDoS attacks as can be recalled from [15]. Lastly, we looked at connection behaviors for each attack to understand the traits exhibited in DDoS attacks. Two are notable. First are the number of ports used by an attacker. During both attacks it was seen that all or almost all of the available port ranges were used by the attack to try and overwhelm the victims. The second trait was the connection histories of both attacks. The strings DT and DTT indicated that DDoS attacks leverage the strategy of retransmitting bytes it previously sent. Doing this over and over again combined with the number of ports used would indicate flooding of information thus causing the target systems to potentially fail.

Overall, this study revealed some valuable insights into DDoS attacks. While it did not cover the entire spectrum of various DDoS attacks or dive deeper into the analysis of the particular traits it does demonstrate areas that could be focused on for a future study. As mentioned in the Background and Related Works security in IoT is still young and there is a lot of room for more research. Securing our fragile IoT infrastructure is also imperative for a continuously interconnected and digitally driven society.

VIII. Future work

This research project only scratches the surface of what could be done and is limited in scope. Future work could include looking at more than one dataset, more sophisticated feature analysis, looking at different types of DDoS attacks such as Mirai and Okiru. Further time series analysis could look at data that contains mili or nano second precision to discover inflection points of when spikes occur potentially aiding detection algorithms. More sophisticated analysis in outlier or anomaly detection could be something to look at as well.

IX. References

[1] National Intelligence Council,Washington,DC, “Disruptive Civil Technologies: Six Technologies With Potential Impacts on US Interests Out to 2025,” 2008.

[2] “Cloud IoT Core | Google Cloud.” Accessed: Apr. 25, 2024. [Online]. Available: https://cloud.google.com/iot-core

[3] “Industrial Internet of Things (IIoT) Solutions - Intel.” Accessed: Apr. 25, 2024. [Online]. Available: https://www.intel.com/content/www/us/en/internet-of-things/industrial-iot/overview.html

[4] “IoT Solutions | IBM.” Accessed: Apr. 25, 2024. [Online]. Available: https://www.ibm.com/cloud/internet-of-things?utm_content=SRCWW&p1=Search&p4=43700064659068667&p5=p&gad_source=1&gclid=Cj0KCQjw_qexBhCoARIsAFgBlet_P7ysJs5otKQ-fwsisLV-aDEjvH1KERhf4GV-IbvTKqsNlRmff7AaAmf2EALw_wcB&gclsrc=aw.ds

[5] “Build Free IoT Solutions on AWS.” Accessed: Apr. 25, 2024. [Online]. Available: https://aws.amazon.com/free/iot/?gclid=Cj0KCQjw_qexBhCoARIsAFgBlevxqyTQpYNaOvv1SuiYqsszMEqBgeYybF0ITa06-ND-JEtAHlOj7MgaAqMfEALw_wcB&trk=d96365ed-3ce7-4dd5-9cab-102978dac4ce&sc_channel=ps&ef_id=Cj0KCQjw_qexBhCoARIsAFgBlevxqyTQpYNaOvv1SuiYqsszMEqBgeYybF0ITa06-ND-JEtAHlOj7MgaAqMfEALw_wcB:G:s&s_kwcid=AL!4422!3!651784491568!p!!g!!internet%20of%20things%20service!19852661753!145019262697

[6] “Internet of Things (IoT) Control Center - Cisco.” Accessed: Apr. 25, 2024. [Online]. Available: https://www.cisco.com/c/en/us/solutions/internet-of-things/iot-control-center.html

[7] R. Khader and D. Eleyan, “Survey of DoS/DDoS attacks in IoT,” *Sustain. Eng. Innov.*, vol. 3, no. 1, pp. 23–28, 2021.

[8] “PLC vs. IOT - PLC Systems.” Accessed: Apr. 26, 2024. [Online]. Available: https://www.plctable.com/plc-vs-iot/

[9] T. M. Chen and S. Abu-Nimeh, “Lessons from Stuxnet,” *Computer*, vol. 44, no. 4, pp. 91–93, 2011, doi: 10.1109/MC.2011.115.

[10] Samie, Farzad ; Bauer, Lars ; Henkel, Jörg, “IoT technologies for embedded computing: a survey,” presented at the 2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2016, pp. 1–10.

[11] Lee, Suk ; Bae, Mungyu ; Kim, Hwangnam, “Future of IoT Networks: A Survey,” *Appl. Sci.*, vol. Vol.7, 2017.

[12] S. U. Rehman, S. Manickam, and N. F. Firdous, “Impact of DoS/DDoS attacks in IoT environment: A study,” in *AIP Conference Proceedings*, Melville: American Institute of Physics, 2023.

[13] “Major DDoS attack on Dyn disrupts AWS, Twitter, Spotify and more - DCD.” Accessed: Apr. 26, 2024. [Online]. Available: https://www.datacenterdynamics.com/en/news/major-ddos-attack-on-dyn-disrupts-aws-twitter-spotify-and-more/

[14] Manos Antonakakis, Georgia Institute of Technology; Tim April, Akamai; Michael Bailey, *et al.*, “Understanding the Mirai Botnet,” in *26th USENIX Security Symposium*, Vancouver, BC, Canada, Aug. 2017, pp. 1093–1110. Accessed: Apr. 12, 2024. [Online]. Available: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis

[15] A. Lohachab and B. Karambir, “Critical Analysis of DDoS—An Emerging Security Threat over IoT Networks,” *J. Commun. Inf. Netw.*, vol. 3, no. 3, pp. 57–78, 2018.

[16] Kyle Chin, “What is TTP Hunting? | UpGuard.” Accessed: Apr. 14, 2024. [Online]. Available: https://www.upguard.com/blog/what-is-ttp-hunting

[17] H. Tahaei, F. Afifi, A. Asemi, F. Zaki, and N. B. Anuar, “The rise of traffic classification in IoT networks: A survey,” *J. Netw. Comput. Appl.*, vol. 154, pp. 102538-, 2020.

[18] Nikhil Handigol, Brandon Heller, Vimalkumar Jeyakumar, David Mazières, and and Nick McKeown, Stanford University, “I Know What Your Packet Did Last Hop: Using Packet Histories to Troubleshoot Networks,” in *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI ’14)*, Seattle, WA, USA, Apr. 2014, pp. 71–85. Accessed: Apr. 12, 2024. [Online]. Available: https://www.usenix.org/conference/nsdi14/technical-sessions/presentation/handigol

[19] E. Hodo *et al.*, “Threat analysis of IoT networks Using Artificial Neural Network Intrusion Detection System,” *arXiv.org*, 2017.

[20] J. H. Kalwar and S. Bhatti, “Deep Learning Approaches for Network Traffic Classification in the Internet of Things (IoT): A Survey,” *arXiv.org*, 2024.

[21] S. Mahadik, P. M. Pawar, and R. Muthalagu, “Efficient Intelligent Intrusion Detection System for Heterogeneous Internet of Things (HetIoT),” *J. Netw. Syst. Manag.*, vol. 31, no. 1, pp. 2-, 2023.

[22] Sebastian Garcia, Agustin Parmisano, & Maria Jose Erquiaga. (2020)., “IoT-23: A labeled dataset with malicious and benign IoT network traffic (Version 1.0.0) [Data set]. Zenodo.” 2020. Accessed: Apr. 01, 2024. [Online]. Available: http://doi.org/10.5281/zenodo.4743746

[23] G. Harris and M. C. Richardson, “PCAP Capture File Format.” Accessed: Apr. 14, 2024. [Online]. Available: https://www.ietf.org/archive/id/draft-gharris-opsawg-pcap-01.html

[24] “Zeek Logs.” corelight, 2020. Accessed: Apr. 04, 2024. [Online]. Available: https://f.hubspotusercontent00.net/hubfs/8645105/Corelight_May2021/Pdf/002_CORELIGHT_080420_ZEEK_LOGS_US_ONLINE.pdf

[25] “Zeek Documentation — Book of Zeek (git/master).” Accessed: Apr. 04, 2024. [Online]. Available: https://docs.zeek.org/en/master/index.html

[26] “Pandas vs PySpark DataFrame With Examples - Spark By {Examples}.” Accessed: Apr. 21, 2024. [Online]. Available: https://sparkbyexamples.com/pyspark/pandas-vs-pyspark-dataframe-with-examples/#google_vignette

[27] [27] “RDD Programming Guide - Spark 3.5.1 Documentation.” Accessed: Apr. 21, 2024. [Online]. Available: https://spark.apache.org/docs/latest/rdd-programming-guide.html

[28] “Scaling to large datasets — pandas 2.2.2 documentation.” Accessed: Apr. 21, 2024. [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/user_guide/scale.html

[29] “RFC 791: Internet Protocol.” Accessed: Apr. 15, 2024. [Online]. Available: https://www.rfc-editor.org/rfc/rfc791

[30] “Traffic Analysis for Network Security: Two Approaches for Going Beyond Network Flow Data.” Accessed: Apr. 27, 2024. [Online]. Available: https://insights.sei.cmu.edu/blog/traffic-analysis-for-network-security-two-approaches-for-going-beyond-network-flow-data/

[31] “Cloudflare Radar.” Accessed: Apr. 21, 2024. [Online]. Available: https://radar.cloudflare.com/reports/ddos-2022-q3#17-network-layer-ddos-attack-trends

[32] “Registered Port - an overview | ScienceDirect Topics.” Accessed: Apr. 24, 2024. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/registered-port

[33] “base/protocols/conn/main.zeek — Book of Zeek (v6.2.0).” Accessed: Apr. 24, 2024. [Online]. Available: https://docs.zeek.org/en/current/scripts/base/protocols/conn/main.zeek.html

[34] “RFC 9293 - Transmission Control Protocol (TCP).” Accessed: Apr. 24, 2024. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc9293