#Course Name

Name: #Student Name

Introduction

Most of the organisations nowadays are vulnerable to cyber threats because of the technology advancement, networks, social media. Data breaches are typical cyber attacks which have a huge impact on the organisations that store their sensitive customer data in the form of huge chunks in company servers without proper safety measures. (Hammouchi et al., 2019, Pg.No 12) Cyber security criminals target the high volumes of sensitive and valuable customer data maintained by organisations and use this data for their own advantage.To prevent these ransomware attacks, the organisations must take preventive measures to stop ransomware attacks, such as adopting sophisticated security systems, regularly assessing risks, threat intelligence tools and educating employees to spot phishing efforts. (Miranda et al., 2021, Pg.No 22)Regular hardware and software updates need to be done in order to reduce the risks. Every organisation should have a predesigned incident response strategy which includes steps like restoring data from safe backups, reporting the breach to law authorities, and isolating compromised systems to apply during attacks.

PROBLEM

In the era of organisations facing constant threats and having vulnerabilities to cyber attacks. It is really important to perform advanced analytics to identify patterns, trends, and vulnerabilities in cybersecurity attacks. (Beazley et al., 2019, Pg.No 5) In this project through Exploratory Data Analysis I mainly tried to understand the patterns of cybersecurity attacks, logical ports that have been attacked along with the common type of attacks in various time periods in a day. (Nicodemo & Satorra, 2020, Pg.No 51)The main goal is to improve the protection against vulnerabilities by analysing the origin and type of attack.

Data Set

For representing the cybersecurity attacks I chose a dataset from Kaggle. This dataset contains several variables which provide the information related to the nature of the attacks and .

Attributes:

- Attack category : Type of registered cybersecurity attack
- Attack subcategory: A subcategory of the type of cybersecurity attack registered
- Attack Name: The technical name for the cybersecurity attack
- Time: Start and end date of the attack in timestamp format
- Protocol: The protocol used for the attack.
- Source IP: IPv4 address where the attack came from.
- Source Port: The logical port where the attack came from.
- Destination IP: Destination IPv4 address.
- Destination Port: Logical destination port

Tools: Python and Jupyter Notebook.

Cleaning

First I created a copy of the database df_before_cleaning before cleaning for reference.

Dropping unnecessary columns: The time column is splitted into Start time and End time.

In [8]:
```
df[['Start time','End time']] = df['Time'].str.split('-',expand=True) #split the time into start and end time
df.head()
```

Out[8]:

SourcePort	DestinationIP	DestinationPort	AttackName	AttackReference	.	Time	Start time	End time
13284	149.171.126.16	80	Domino Web Server Database Access: /doladmin.n...	-	.	1421927414-1421927416	1421927414	1421927416
21223	149.171.126.18	32780	Solaris rwalld Format String Vulnerability (ht...	CVE 2002-0573 (http://cve.mitre.org/cgi-bin/cv...	.	1421927415-1421927415	1421927415	1421927415
23357	149.171.126.16	80	Windows Metafile (WMF) SetAbortProc() Code Exe...	CVE 2005-4560 (http://cve.mitre.org/cgi-bin/cv...	.	1421927416-1421927416	1421927416	1421927416
13792	149.171.126.16	5555	HP Data Protector Backup (https://strikecenter...	CVE 2011-1729 (http://cve.mitre.org/cgi-bin/cv...	.	1421927417-1421927417	1421927417	1421927417
26939	149.171.126.10	80	Cisco IOS HTTP Authentication Bypass Level 64 ...	CVE 2001-0537 (http://cve.mitre.org/cgi-bin/cv...	.	1421927418-1421927418	1421927418	1421927418

In [9]:
```
df['.'].unique()# there is no benfit for this columns
```

Out[9]:
```
array(['.'], dtype=object)
```

Later '.'and 'Time' are dropped as they are not required anymore.

In [10]:
```
df = df.drop(['.', 'Time'],axis=1)# Drop the two "." and 'time'
df.head()
```

Out[10]:

·col	SourceIP	SourcePort	DestinationIP	DestinationPort	AttackName	AttackReference	Start time	End time
tcp	175.45.176.0	13284	149.171.126.16	80	Domino Web Server Database Access: /doladmin.n...	-	1421927414	1421927416
udp	175.45.176.3	21223	149.171.126.18	32780	Solaris rwalld Format String Vulnerability (ht...	CVE 2002-0573 (http://cve.mitre.org/cgi-bin/cv...	1421927415	1421927415
tcp	175.45.176.2	23357	149.171.126.16	80	Windows Metafile (WMF) SetAbortProc() Code Exe...	CVE 2005-4560 (http://cve.mitre.org/cgi-bin/cv...	1421927416	1421927416
tcp	175.45.176.2	13792	149.171.126.16	5555	HP Data Protector Backup (https://strikecenter...	CVE 2011-1729 (http://cve.mitre.org/cgi-bin/cv...	1421927417	1421927417
tcp	175.45.176.2	26939	149.171.126.10	80	Cisco IOS HTTP Authentication Bypass Level 64 ...	CVE 2001-0537 (http://cve.mitre.org/cgi-bin/cv...	1421927418	1421927418

Finding null values: Using isnull() I found there are 4476 missing values in the '**Attacksubcategory**' column and 51745 missing values in the '**AttackReference**' column. AsAttacksubcategory column is not mostly used I replaced the null values with value NotRegistered.

In [11]:
```
df.isnull().sum() # check from null value
```

Out[11]:
```
Attackcategory            0
Attacksubcategory      4476
Protocol                  0
SourceIP                  0
SourcePort                0
DestinationIP             0
DestinationPort           0
AttackName                0
AttackReference       51745
Start time                0
End time                  0
dtype: int64
```

In [12]:
```
df["Attacksubcategory"] = df["Attacksubcategory"].fillna("Not Registered")
```

In [13]:
```
df.isnull().sum()
```

Out[13]:
```
Attackcategory            0
Attacksubcategory         0
Protocol                  0
SourceIP                  0
SourcePort                0
DestinationIP             0
DestinationPort           0
AttackName                0
AttackReference       51745
Start time                0
End time                  0
dtype: int64
```

For the AttackReference column, I analysed each category individually to identify which had the highest percentage of null values. The analysis revealed that the Reconnaissance category has the

largest proportion(90.11%) of null values.

In [13]:
```
df.isnull().sum()
```

Out[13]:
```
Attackcategory           0
Attacksubcategory        0
Protocol                 0
SourceIP                 0
SourcePort               0
DestinationIP            0
DestinationPort          0
AttackName               0
AttackReference      51745
Start time               0
End time                 0
dtype: int64
```

In [14]:
```
print(df[pd.isnull(df['AttackReference'])]['Attackcategory'].value_counts()) #to know which attack category have
```

```
Fuzzers           30297
Reconnaissance    18538
Analysis           1657
Shellcode           761
Generic             351
Backdoor             68
DoS                  56
Worms                12
Exploits              5
Name: Attackcategory, dtype: int64
```

In [15]:
```
# Percentage of missing values in 'Attack Reference' per Attack Category
((df[pd.isnull(df['AttackReference'])]['Attackcategory'].value_counts()/df['Attackcategory'].value_counts())*100)
```

Out[15]:
```
Reconnaissance    90.117155
Fuzzers           88.172638
Analysis          85.721676
Shellcode         49.383517
Worms              6.936416
Generic            1.729405
Backdoor           1.622137
DoS                0.222957
Exploits           0.007185
Name: Attackcategory, dtype: float64
```

Removed Duplicates: Identified and removed 6 duplicate rows from the dataset.

In [17]:
```
df[df.duplicated()].shape  # check from duplicated
```

Out[17]:
```
(6, 11)
```

In [18]:
```
print('The Dimensions before dropping duplicated rows: ' + str(df.shape))
df = df.drop(df[df.duplicated()].index)
print('The Dimensions after dropping duplicated rows: ' + str(df.shape))
```

```
The Dimensions before dropping duplicated rows: (178031, 11)
The Dimensions after dropping duplicated rows: (178025, 11)
```

Invalid Port Filtering:Filtered rows where SourcePort or DestinationPort had invalid values outside the range 0-65535.Standardized text in the Protocol and Attackcategory columns to uppercase and Merged categories Backdoors -> Backdoor since they have same purpose.

In [19]:
```
invalid_SourcePort = (df['SourcePort'] < 0) | (df['SourcePort'] > 65535)
invalid_DestinationPort = (df['DestinationPort'] < 0) | (df['DestinationPort'] > 65535)
df[invalid_SourcePort | invalid_DestinationPort].head()
```

Out[19]:

	Attackcategory	Attacksubcategory	Protocol	SourceIP	SourcePort	DestinationIP	DestinationPort	
174347	Generic	IXIA	udp	175.45.176.1	67520	149.171.126.18	53	Microsoft_DNS_Ser
174348	Exploits	Browser	tcp	175.45.176.3	78573	149.171.126.18	110	Microsof
174349	Reconnaissance	HTTP	tcp	175.45.176.1	71804	149.171.126.10	80	Domino Web
174350	DoS	Ethernet	pnni	175.45.176.3	0	149.171.126.19	-753	Cisco IPS Ju
174351	Fuzzers	OSPF	trunk-1	175.45.176.0	73338	149.171.126.13	0	Fuzzer: OSP

In [20]:
```
df = df[~(invalid_SourcePort | invalid_DestinationPort)].reset_index(drop=True)
```

In [21]:
```
df.shape
```

Out[21]:
```
(174341, 11)
```

In [22]:
```
print("Attack category:",df['Attackcategory'].unique()) # there is duplicated such as tcp and TCP
print('Protocol:',df['Protocol'].unique()[:15]) # Backdoor vs Backdoors
```

```
Attack category: ['Reconnaissance' 'Exploits' 'DoS' 'Generic' 'Shellcode' 'Fuzzers' 'Worms'
 'Backdoors' 'Analysis' 'Backdoor']
Protocol: ['tcp' 'udp' 'Tcp' 'UDP' 'ospf' 'sctp' 'sep' 'mobile' 'sun-nd' 'swipe'
 'pim' 'ggp' 'ip' 'ipnip' 'st2']
```

In [23]:
```
df['Protocol'] = df['Protocol'].str.upper().str.strip()
df['Attackcategory'] = df['Attackcategory'].str.upper().str.strip()
df['Attackcategory'] = df['Attackcategory'].str.strip().replace('BACKDOORS','BACKDOOR')

df.head()
```

Feature Engineering: Converted Start time and End time to datetime objects and also added derived features like Duration, hour, Month, and Day.

In [24]:
```
df['Start time'] = pd.to_datetime(df['Start time'], unit='s')
df['End time'] = pd.to_datetime(df['End time'], unit='s')
df['Duration'] = ((df['End time'] - df['Start time']).dt.seconds).astype(int)
df['hour'] = df.apply(lambda row: '0'*(2-len(str(row['Start time'].hour)))+str(row['Start time'].hour)+':00:00',
df['Month'] = df['End time'].dt.month
df['Day'] = df['End time'].dt.day

df
```

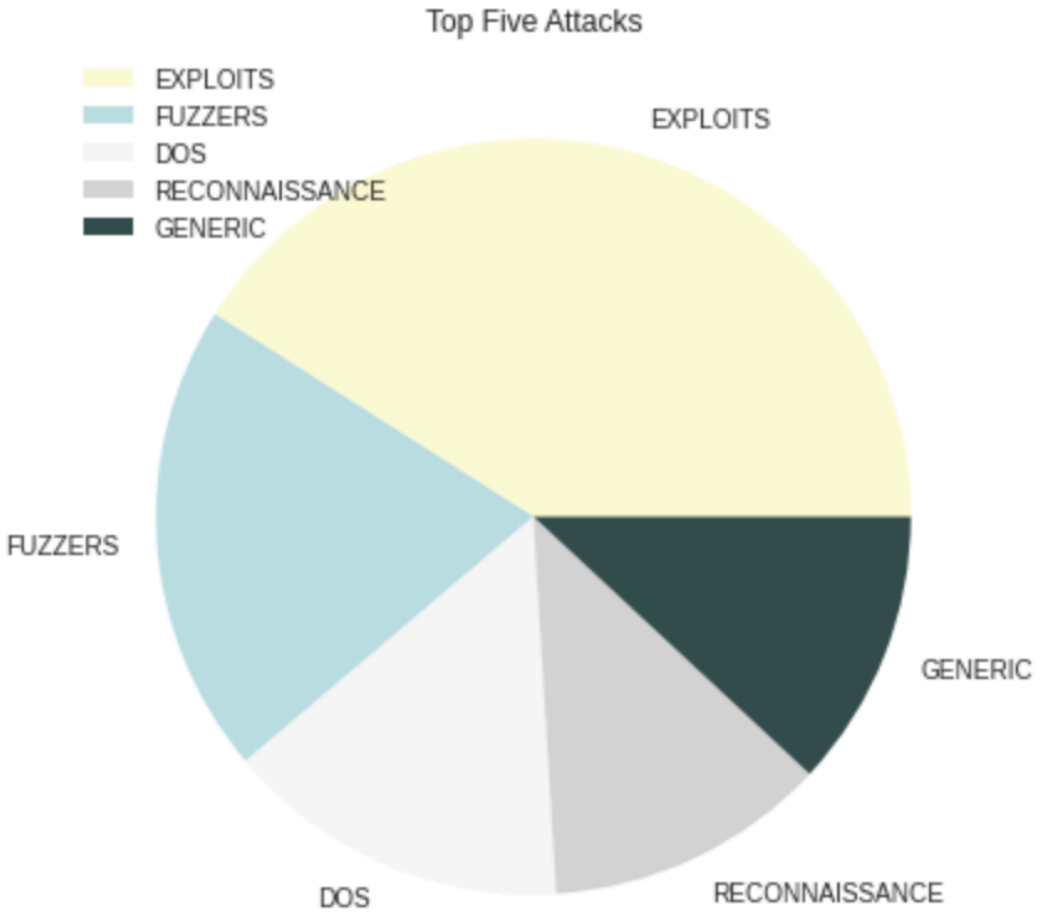
Out[24]:

rt	AttackName	AttackReference	Start time	End time	Duration	hour	Month	Day
0	Domino Web Server Database Access: /doladmin.n...	-	2015-01-22 11:50:14	2015-01-22 11:50:16	2	11:00:00	1	22
0	Solaris rwalld Format String Vulnerability (ht...	CVE 2002-0573 (http://cve.mitre.org/cgi-bin/cv...	2015-01-22 11:50:15	2015-01-22 11:50:15	0	11:00:00	1	22
0	Windows Metafile (WMF) SetAbortProc() Code Exe...	CVE 2005-4560 (http://cve.mitre.org/cgi-bin/cv...	2015-01-22 11:50:16	2015-01-22 11:50:16	0	11:00:00	1	22
5	HP Data Protector Backup (https://strikecenter...	CVE 2011-1729 (http://cve.mitre.org/cgi-bin/cv...	2015-01-22 11:50:17	2015-01-22 11:50:17	0	11:00:00	1	22
0	Cisco IOS HTTP Authentication Bypass Level 64 ...	CVE 2001-0537 (http://cve.mitre.org/cgi-bin/cv...	2015-01-22 11:50:18	2015-01-22 11:50:18	0	11:00:00	1	22

Exploratory Data Analysis

Hypothesis: Identifying the most common types of cybersecurity attacks.

The top five attack categories are Exploit, Fuzzers, DoS, Reconnaissance, and Generic. With the help of the frequency of different attack categories we can improve security measures against these specific attacks the most which will help the organizations to protect their customers data from ransomware attackers.



Color representation:

Lemonchiffon:Exploit

Powderblue:Fuzzers

Lightgray:DoS

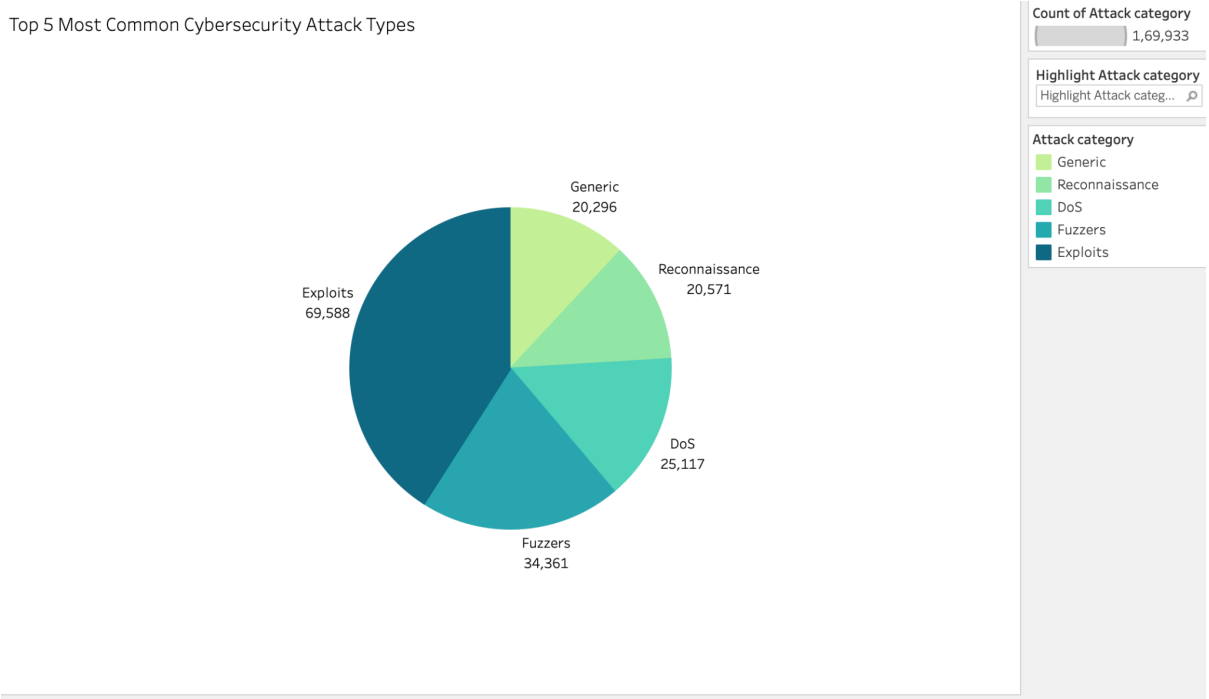
Darkslategray:Reconnaissance

darkseagreen:Generic

Tableau Visualizations:

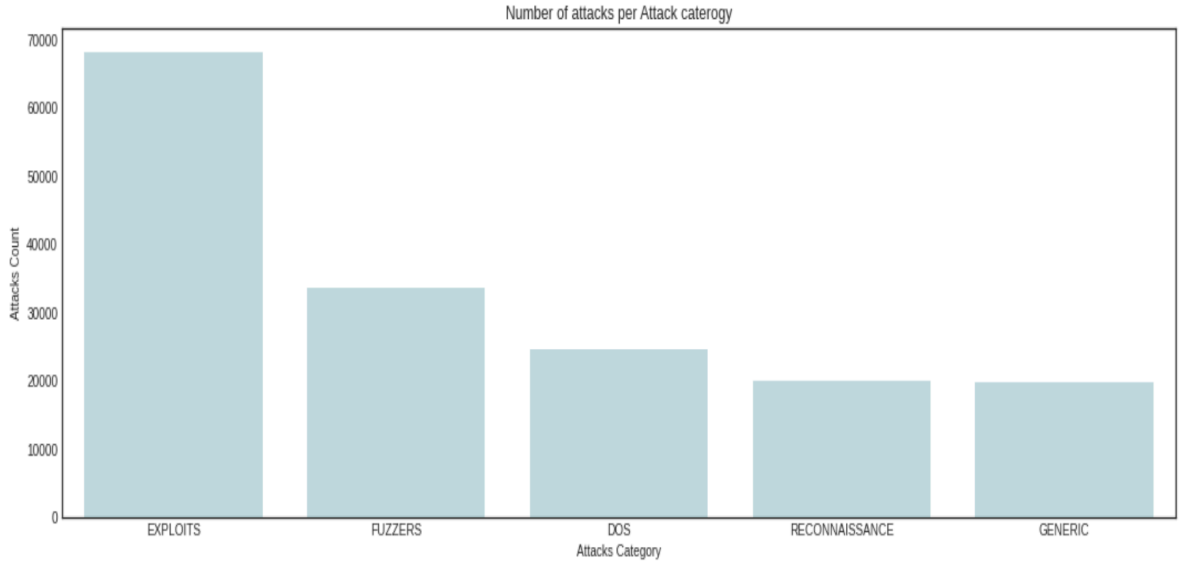
I used a pie chart to visualise the top five most common attack categories. The analysis revealed that the top five attack categories are Exploit, Fuzzers, DoS, Reconnaissance, and Generic. By finding the count of each attack category in the dataset,I found which attacks are most frequent.I choose pie chart

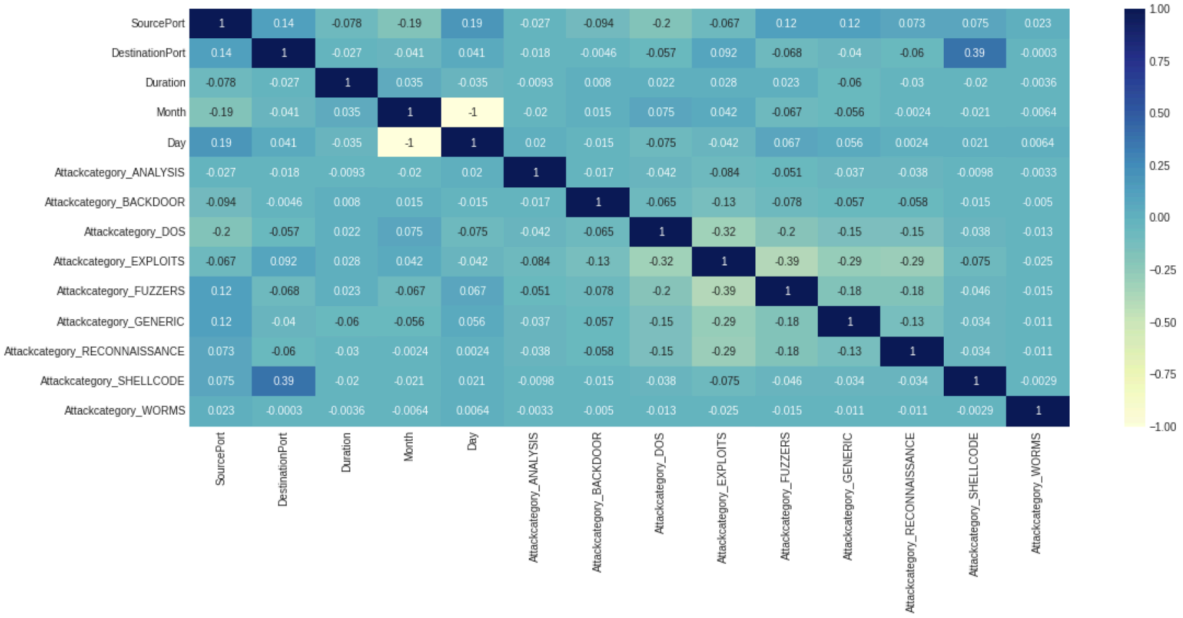
because it effectively displays relative proportions. I used Lighting Bluegrass colours to improve visual clarity and it also helps to differentiate the categories easily.



Hypothesis: Identifying the Attack Category with the Highest Number of Attacks

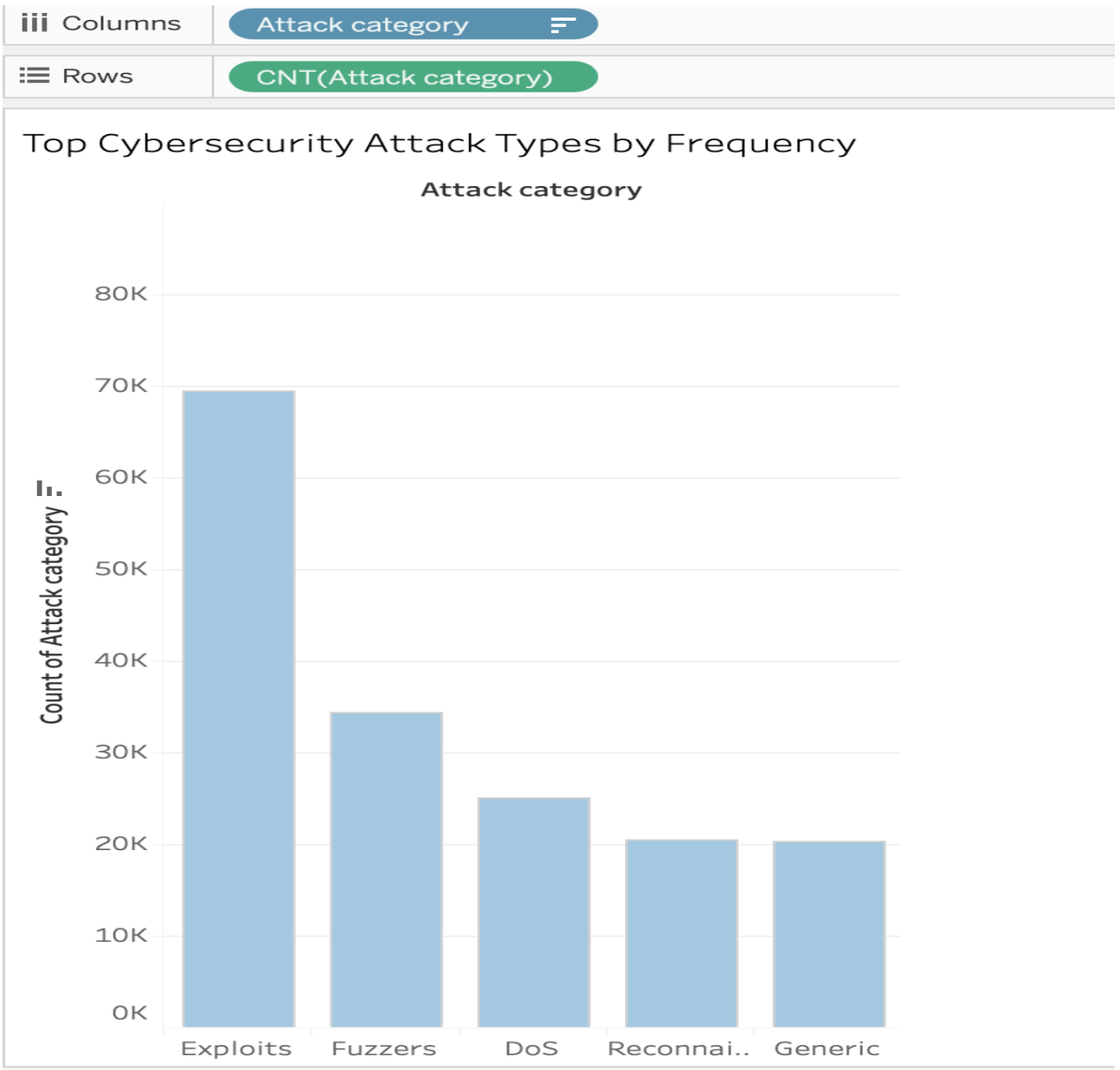
The Exploit category has the highest number of attacks compared to the others. Finding the attack category has the highest number of attacks helps in resource allocation and prioritising the counter measures.I used a bar chart to display the exact count of attacks for each category, highlighting the dominance of Exploit attacks. The analysis revealed that the Exploit category has the highest number of attacks compared to the others. I used the attack category and count measure to find the count of each category. Using a filter I visualized the count of the top five attacks frequency. I choose bar charts because they are best for comparing numerical values across categories. I used a simple blue tone to maintain consistency.





The correlation analysis shows that even though the relationship among the variables is weak there exists a strong monotonic relation between them.

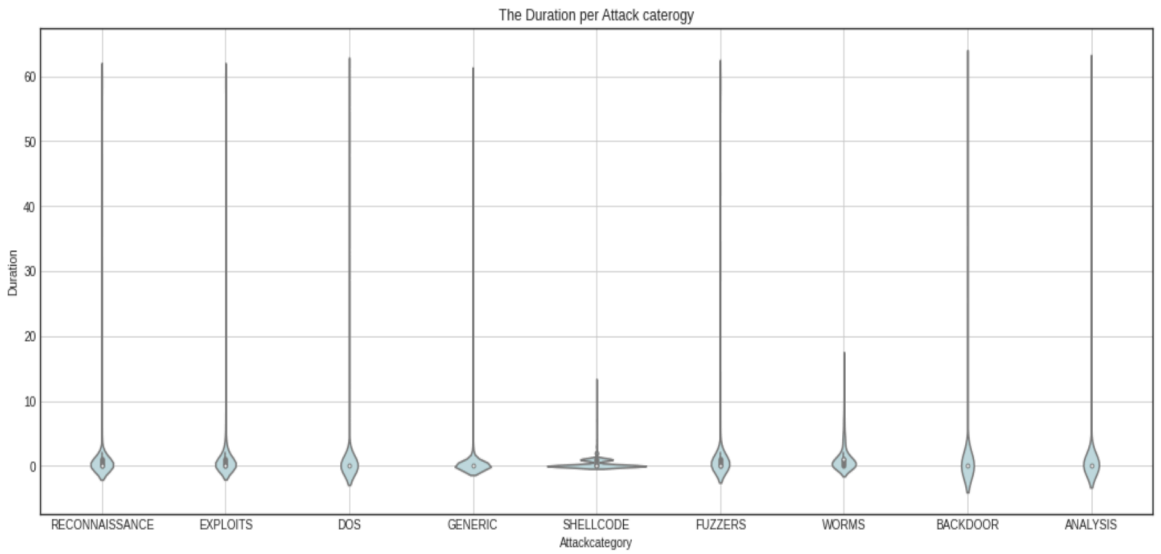
Tableau Visualizations:



Hypothesis: Understanding the duration of attacks for each category.

There were two types of attacks that do not record times greater than 20 seconds, the shellcode and the worms. Most distributions are normal distributions, except for the one found in shellcode, which has two peaks, indicating a bimodal distribution .

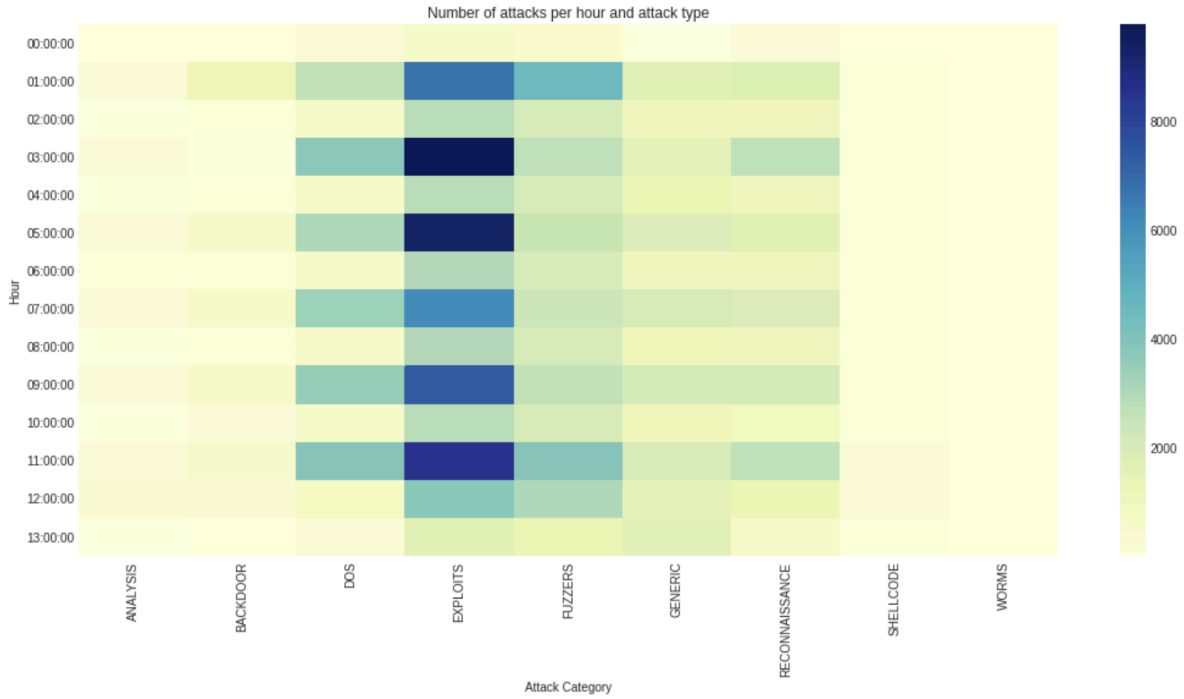
I used a violin plot to visualise the distribution of attack durations for each category. I used a violin plot because of its ability to combine density estimation and box plot features which provides a detailed view of data spread, skewness, and mode.



Hypothesis: Identifying patterns and targeted machines in cybersecurity attacks.

I used a heat map to visualise the Attack Frequency by Hour and Type and Destination IP and Category. The analysis revealed that there is a specific pattern in the attacks, especially for Denial of Service and Exploit attacks. The machine with the IPv4 address 149.171.126.17 has been attacked the most. Conversely, while worms, shellcode and generic attacks are not directed at particular machines, Denial of Services, Exploits and Backdoor attacks are clearly targeted towards specific servers.The attacks were made with more intensity at odd hours. I chose a heat map because it's best for showing patterns over two variables (hour and category). Inorder to convert time(unix timestamp) into readable format I created a calculated field :Start Time using formula LEFT([Time], FIND([Time], '-') - 1) to get the first timestamp of the time range and then created another calculated field Readable Time using formula DATEADD('second', INT([Start Time]), #1970-01-01#) to format the unix timestamp to a readable date and time format.

The **“Blue-Green Sequential”** colour scheme changes from light (low intensity) to dark (high intensity), highlighting the clarity of attack trends.



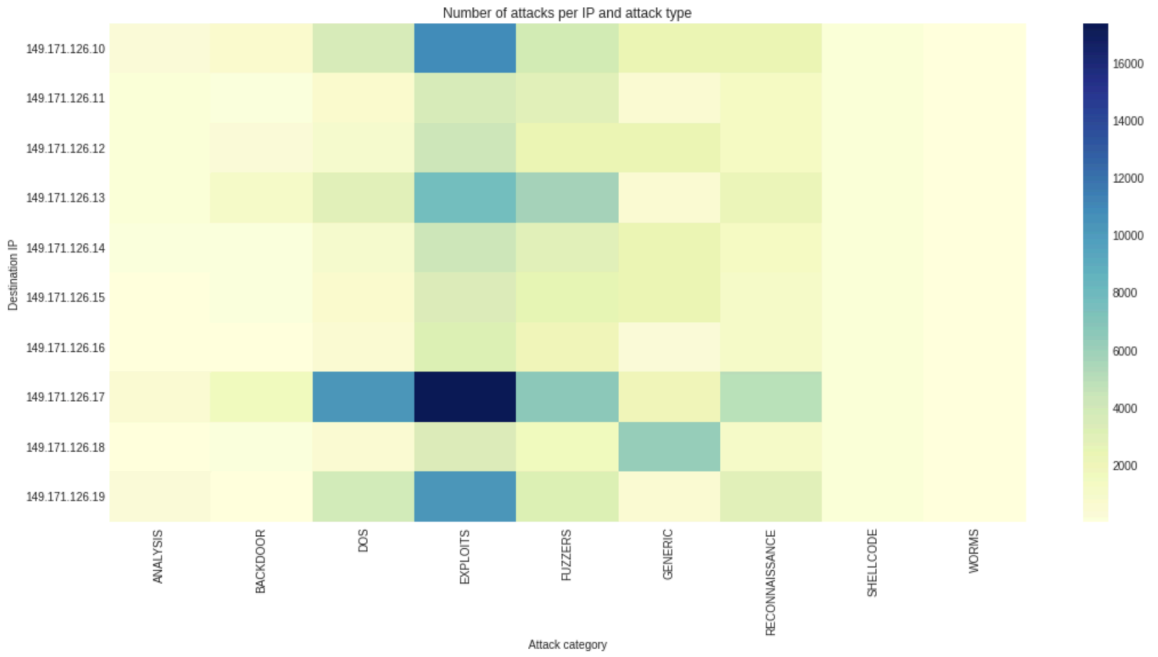
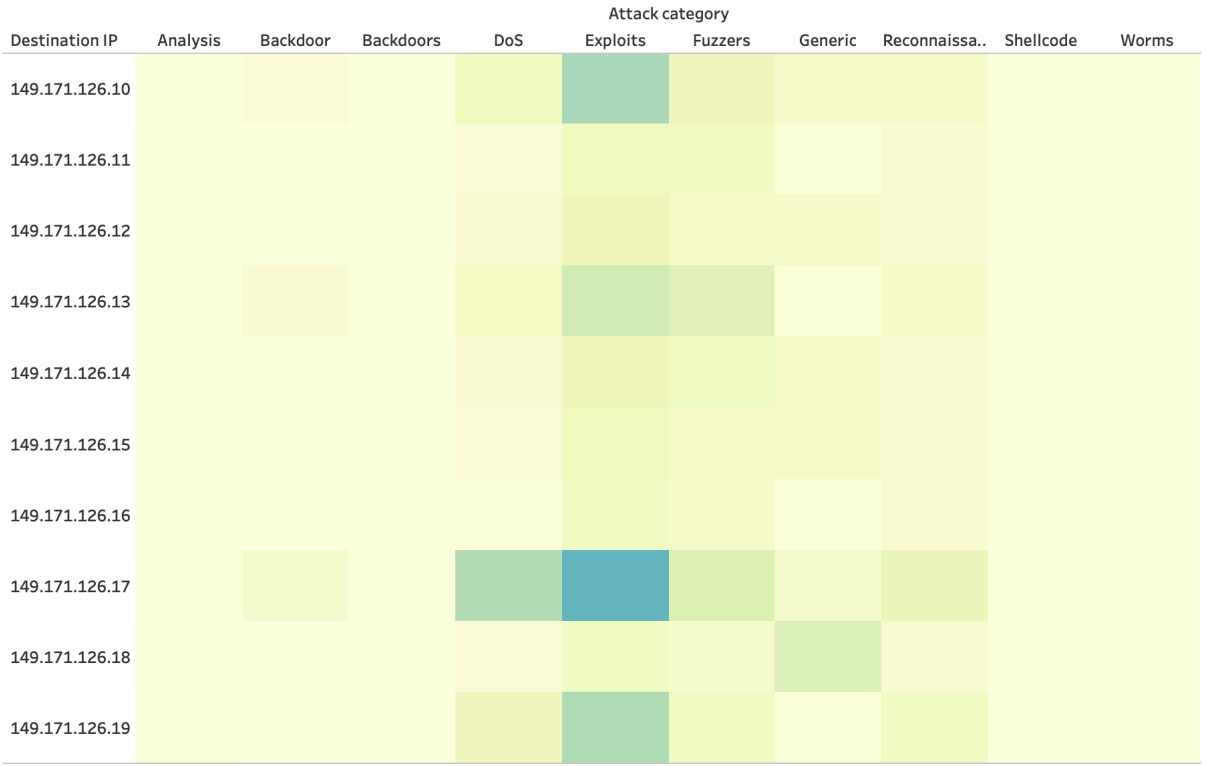


Tableau Visualizations:

Attack Frequency by Hour and Attack Type.

Attack Frequency by IP Address and Attack Type.



Conclusion

In the end through this project I analysed that the most common types of cyber attacks are Exploit, Fuzzers, DoS, Reconnaissance, and Generic with each attack ranging its duration from 1 min to except the Exploit and worms range to 20 sec. The most accessed port is 149.171.126.17 where Denial of Services, Exploits and Backdoor attacks are clearly targeted and were made with more intensity at odd hours.

References

Beazley, C., Gadiya, K., Rakesh, v. K. U., & Roden, D. (2019, April). Exploratory Data Analysis of a Unified Host and Network Dataset. *2019 Systems and Information Engineering Design Symposium*. 10.1109/SIEDS.2019.8735640

Hammouchi, H., Cherqi, O., & Mezzour, G. (2019, January). Digging Deeper into Data Breaches: An Exploratory Data Analysis of Hacking Breaches Over Time. 10.1016/j.procs.2019.04.141

Miranda, J., Chapaala, V. R., & Churi, P. (2021, Feb). Exploratory data analysis for cybersecurity. *World Journal of Engineering*. 10.1108/WJE-11-2020-0560

Nicodemo, C., & Satorra, A. (2020, September). Exploratory data analysis on large data sets: The example of salary variation in Spanish Social Security Data. *BRQ Business Research Quarterly*. 10.1177/2340944420957335