Business Intelligence – Final Exam

Name: Asma Abid Karim

ERP ID; 19685

Step 1: Download any ONE of the following datasets related to different sectors globally:

Internet Firewall Data DataSet: https://archive.ics.uci.edu/ml/datasets/Internet+Firewall+Data

Step 2: Acquire domain knowledge (if you don't have) and then write a problem statement which you will solve using BI (just one liner).

The data is from UCI, machine learning repository it contains data related to internet firewall. It contains the information on different source and destination ports, the bytes sent, bytes received. The NAT source and destination ports are also given to underline data about network address translation. Total number packets that were sent and what number was received, it also covers how much time it takes.

Problem statement:

To identify the pattern in the sending and receiving of packets. Time taken by a information to be transferred from source to its destination, attributes affecting this journey – number of bytes it contain, where it comes from where it is going etc.

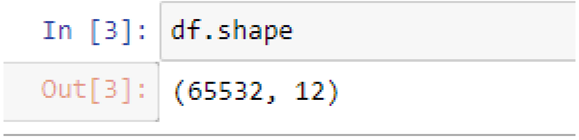
Step 3: Load and Transform data (mention all steps of transformation very briefly)

Columns:

There are total of 12 columns and 65532 rows. Following are the names given below. First, we rename them then understand the data they carry.

- Source Port
- Destination Port
- NAT Source Port
- NAT Destination Port
- Action
- Bytes
- Bytes Sent
- Bytes Received
- Packets
- Elapsed Time (sec)

- pkts_sent
- pkts_received

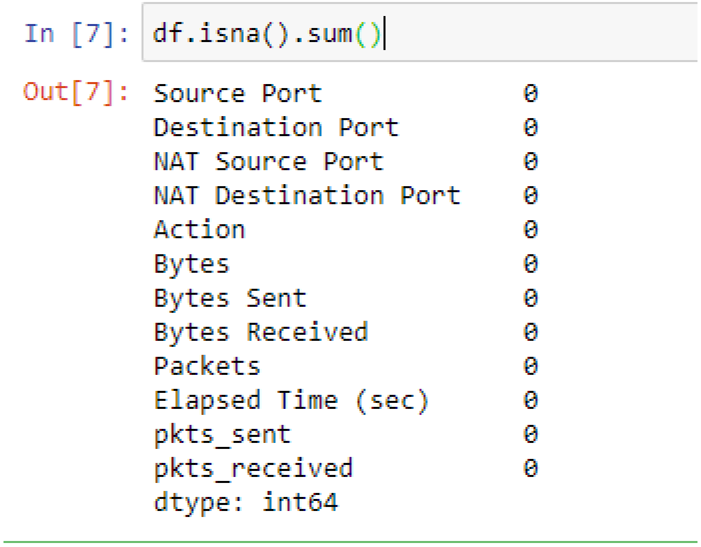


```
In [3]: df.shape

Out[3]: (65532, 12)
```

Missing Values:

The data is rather clean, and the attributes do not have missing values which need to dealt with as show below:



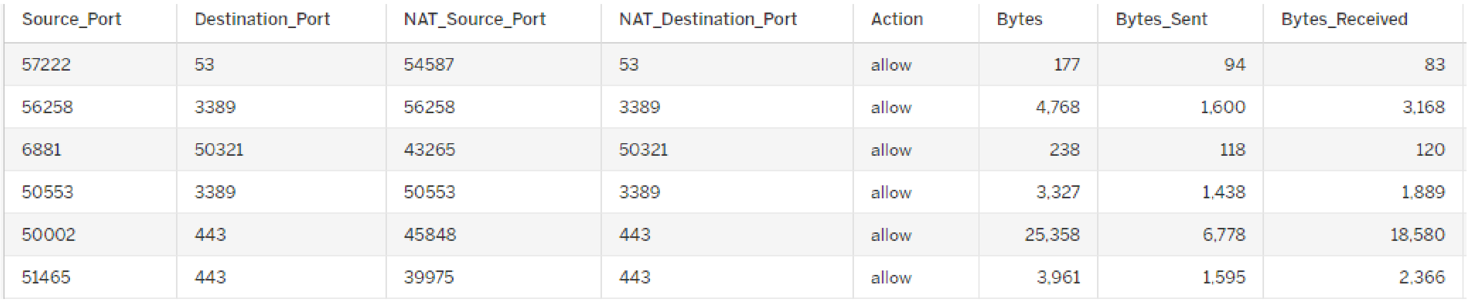
```
In [7]: df.isna().sum()

Out[7]: Source Port             0
        Destination Port        0
        NAT Source Port         0
        NAT Destination Port    0
        Action                  0
        Bytes                   0
        Bytes Sent              0
        Bytes Received          0
        Packets                 0
        Elapsed Time (sec)      0
        pkts_sent               0
        pkts_received           0
        dtype: int64
```

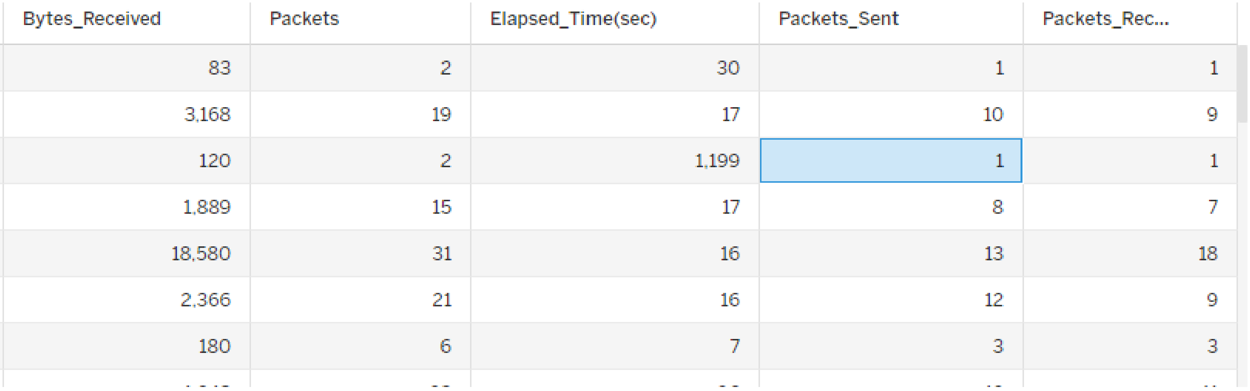
Renaming Columns:

- Source Port to Source_Port
- Destination Port to Destination_Port
- NAT Source Port to NAT_Source_Port
- NAT Destination Port to NAT_Destination_Port
- Bytes Sent to Bytes_Sent

- Bytes Received to Bytes_Received
- Elapsed Time (sec) to Elapsed_Time(sec)
- pkts_sent to Packets_Sent
- pkts_received to Packets_received



Source_Port	Destination_Port	NAT_Source_Port	NAT_Destination_Port	Action	Bytes	Bytes_Sent	Bytes_Received
57222	53	54587	53	allow	177	94	83
56258	3389	56258	3389	allow	4,768	1,600	3,168
6881	50321	43265	50321	allow	238	118	120
50553	3389	50553	3389	allow	3,327	1,438	1,889
50002	443	45848	443	allow	25,358	6,778	18,580
51465	443	39975	443	allow	3,961	1,595	2,366



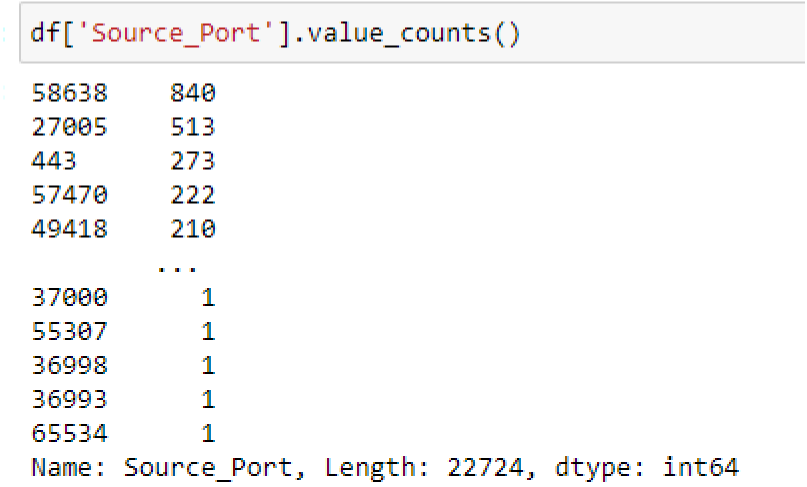
Bytes_Received	Packets	Elapsed_Time(sec)	Packets_Sent	Packets_Rec...
83	2	30	1	1
3,168	19	17	10	9
120	2	1,199	1	1
1,889	15	17	8	7
18,580	31	16	13	18
2,366	21	16	12	9
180	6	7	3	3

Understanding Columns:

Source_Port:

- Numerical column converted to a String type column
- The source port shows the source of the packets that are sent over the internet. The source port serves analogues to the destination port, it is utilized by the supplying host to assist keep track of new incoming connections and current data streams.
- It is converted into string because don't have any aggregation function involved with the port number rather, they can be treated as a category which can be repeated multiple times.

- Since it converted to string, we see how many different source ports are possible. As seen in the picture below there are a few ports number that are repeated several times and few occur just once or twice.

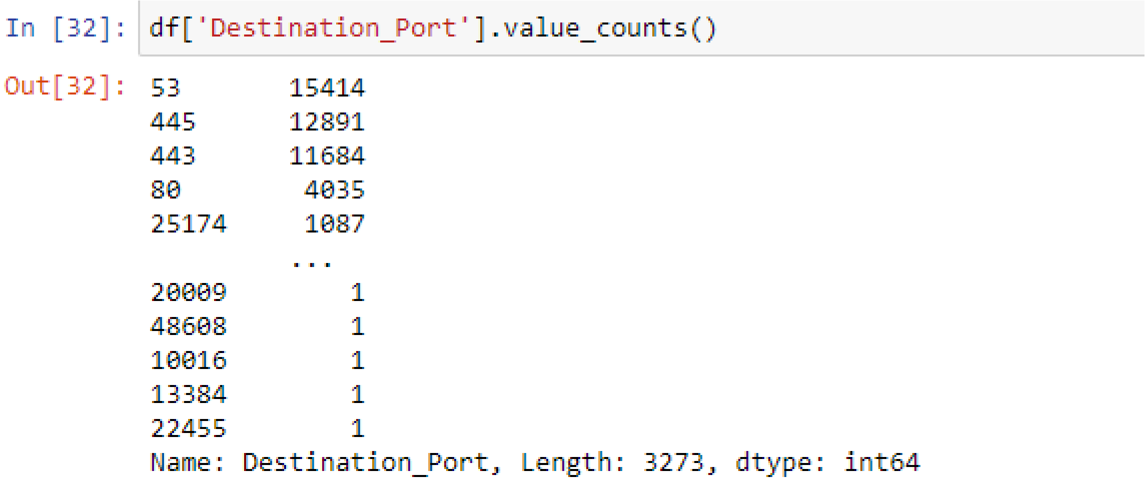


```
df['Source_Port'].value_counts()

58638    840
27005    513
443      273
57470    222
49418    210
        ...
37000      1
55307      1
36998      1
36993      1
65534      1
Name: Source_Port, Length: 22724, dtype: int64
```

Destination_Port:

- Numerical column converted to a String type column
- The destination port shows the destination of the packets that are received over the internet coming from some source. The port, terminal, or refining system where the Product to be supplied hereunder will be discharged is referred as a Destination Port.
- It is converted into string because don't have any aggregation function involved with the port number rather, they can be treated as a category which can be repeated multiple times.
- Since it converted to string, we see how many different destination ports are possible. As seen in the picture below there are a few ports number that are repeated several times and few occur just once or twice.

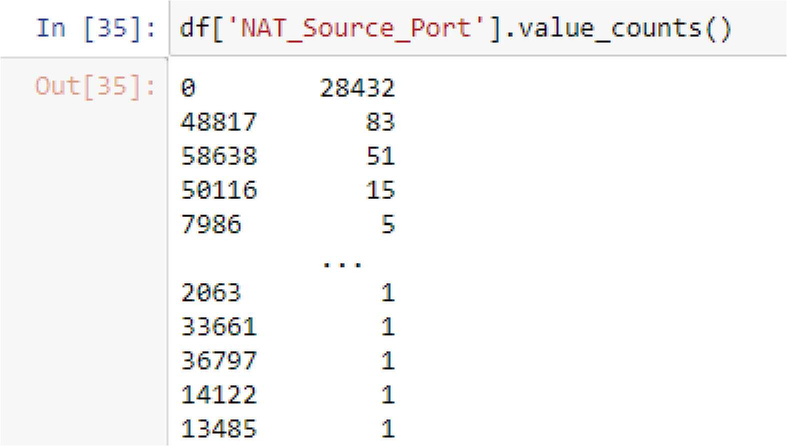


```
In [32]: df['Destination_Port'].value_counts()

Out[32]: 53       15414
         445      12891
         443      11684
         80        4035
         25174     1087
                  ...
         20009        1
         48608        1
         10016        1
         13384        1
         22455        1
         Name: Destination_Port, Length: 3273, dtype: int64
```

NAT_Source_Port:

- Numerical column converted to a String type column
- The source Port number of a packet leaving a Juniper Networks device is translated via source NAT. Source NAT is a network access technique that allows hosts with private IP ports to connect to a public network. This is commonly used to redirect arriving packets with a public address/port destination to a private IP address/port within your network.
- It is converted into string because don't have any aggregation function involved with the port number rather, they can be treated as a category which can be repeated multiple times.
- Since it converted to string, we see how many different NAT source ports are possible. As seen in the picture below there are a few ports number that are repeated several times and few occur just once or twice.

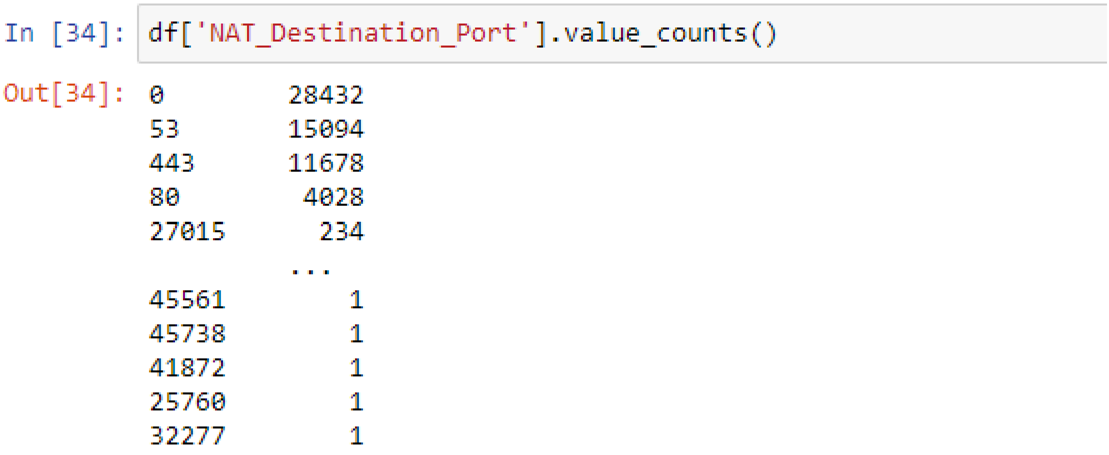


```
In [35]: df['NAT_Source_Port'].value_counts()

Out[35]: 0        28432
         48817       83
         58638       51
         50116       15
         7986         5
                  ...
         2063         1
         33661        1
         36797        1
         14122        1
         13485        1
```

NAT_Destination_Port:

- Numerical column converted to a String type column
- The destination addresses of packets travelling through the Router is changed via destination NAT. Incoming packets with an external address or port destination are often forwarded to an internal IP address or port within the network using destination NAT. It is widely used to provide a service with a publicly visible IP address that is situated on a private network.
- It is converted into string because don't have any aggregation function involved with the port number rather, they can be treated as a category which can be repeated multiple times.
- Since it converted to string, we see how many different NAT destination ports are possible. As seen in the picture below there are a few ports number that are repeated several times and few occur just once or twice.

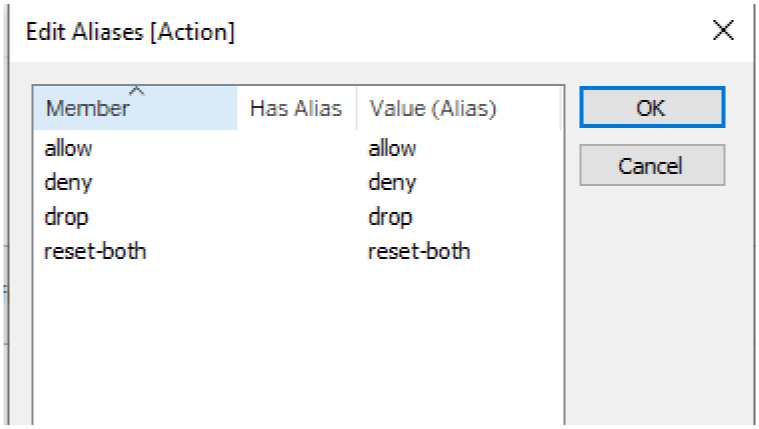


```
In [34]: df['NAT_Destination_Port'].value_counts()

Out[34]: 0        28432
         53       15094
         443      11678
         80        4028
         27015      234
                  ...
         45561        1
         45738        1
         41872        1
         25760        1
         32277        1
```

Action:

- Categorical column
- Have four categories as follow:
    - allow
    - deny
    - drop
    - reset-both



Edit Aliases [Action]

Member	Has Alias	Value (Alias)
allow		allow
deny		deny
drop		drop
reset-both		reset-both

These are firewall rules and actions

- allow: Explicitly allows traffic that matches the rule to pass, and then implicitly denies everything else.
- drop: the firewall discards the packet and sends no response back to the source host that sent the packet.
- deny: Explicitly blocks traffic that matches the rule.
- reset-both: This action will inject a RST packet into the tcp stream, breaking the connection. So, a connection exists, a threat is detected and blocked, and a RST is sent to end the session. A TCP Reset (RST) packet is used by a TCP sender to indicate that it will neither accept nor receive more data.

Bytes:

- Numerical column
- This is the sum of bytes send and received
- This column may not have much significant if use Bytes_Sent, Bytes_Received

Bytes_Sent:

- Numerical column.
- Number of bytes send by a source to a destination.

Bytes_Received:

- Numerical column.
- Number of bytes received by a destination, send by a source.

Elapsed_Time(sec):

- Numerical column.
- Time calculated in seconds.
- Time taken by a packet to travel from source to a destination.

Packets_Sent:

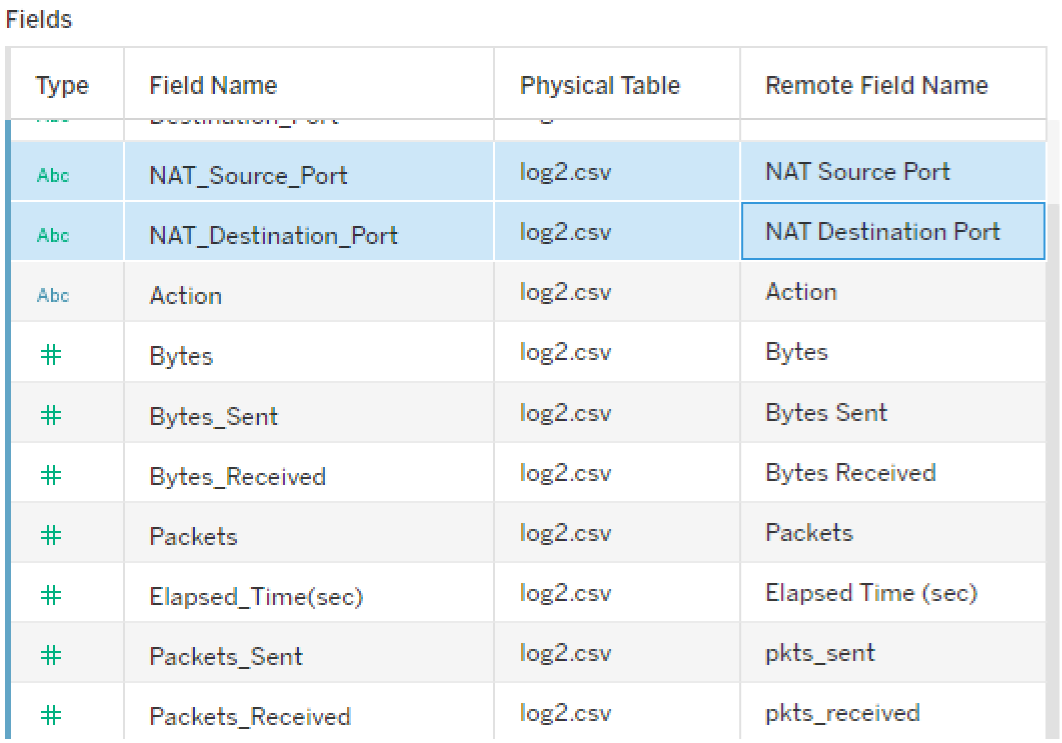
- Numerical column.
- Number of packets send by a source to a destination.
- A typical packet contains perhaps 1,000 or 1,500 bytes. This just an average number and can differ for different packets.

Packets_Received:

- Numerical column.
- Number of packets received by a destination, send by a source.

Packets:

- Numerical column
- This is the sum of Packets send and received
- This column may not have much significant if use Packets_Sent, Packets_Received



Fields

Type	Field Name	Physical Table	Remote Field Name
Abc	Destination_Port	log2.csv	
Abc	NAT_Source_Port	log2.csv	NAT Source Port
Abc	NAT_Destination_Port	log2.csv	NAT Destination Port
Abc	Action	log2.csv	Action
#	Bytes	log2.csv	Bytes
#	Bytes_Sent	log2.csv	Bytes Sent
#	Bytes_Received	log2.csv	Bytes Received
#	Packets	log2.csv	Packets
#	Elapsed_Time(sec)	log2.csv	Elapsed Time (sec)
#	Packets_Sent	log2.csv	pkts_sent
#	Packets_Received	log2.csv	pkts_received

Step 4: List down dimensions (with different types as done in class) along with KPIs (to be extracted from transformed data)

Dimensions (time):

- Elapsed_Time

Since we only have Elapsed_Time, it’s taken as time dimension.

Dimensions (normal):

- Source_Port
- Destination_Port
- NAT_Source_Port
- NAT_Destination_Port

Measure:

- Bytes_Sent
- Bytes_Received
- Packets_Sent
- Packets_received
- Bytes (This is not important to be analyzed and can be dropped)
- Packets (This is not important to be analyzed and can be dropped)

Step 5: List down some potential analyses (can be identified from Step 4)

Time Dimension: Analyze KPI’s across time dimension

1. What are temporal trends of KPI’s across Elapsed_Time

Normal Dimension:

2. Analyze KPI’s across Source_Port, Destination_Port, NAT_Source_Port, NAT_Destination_Port

Questions:

Over the span of collected data which are the major sources and destinations of packets.

How many bytes are there per packets?

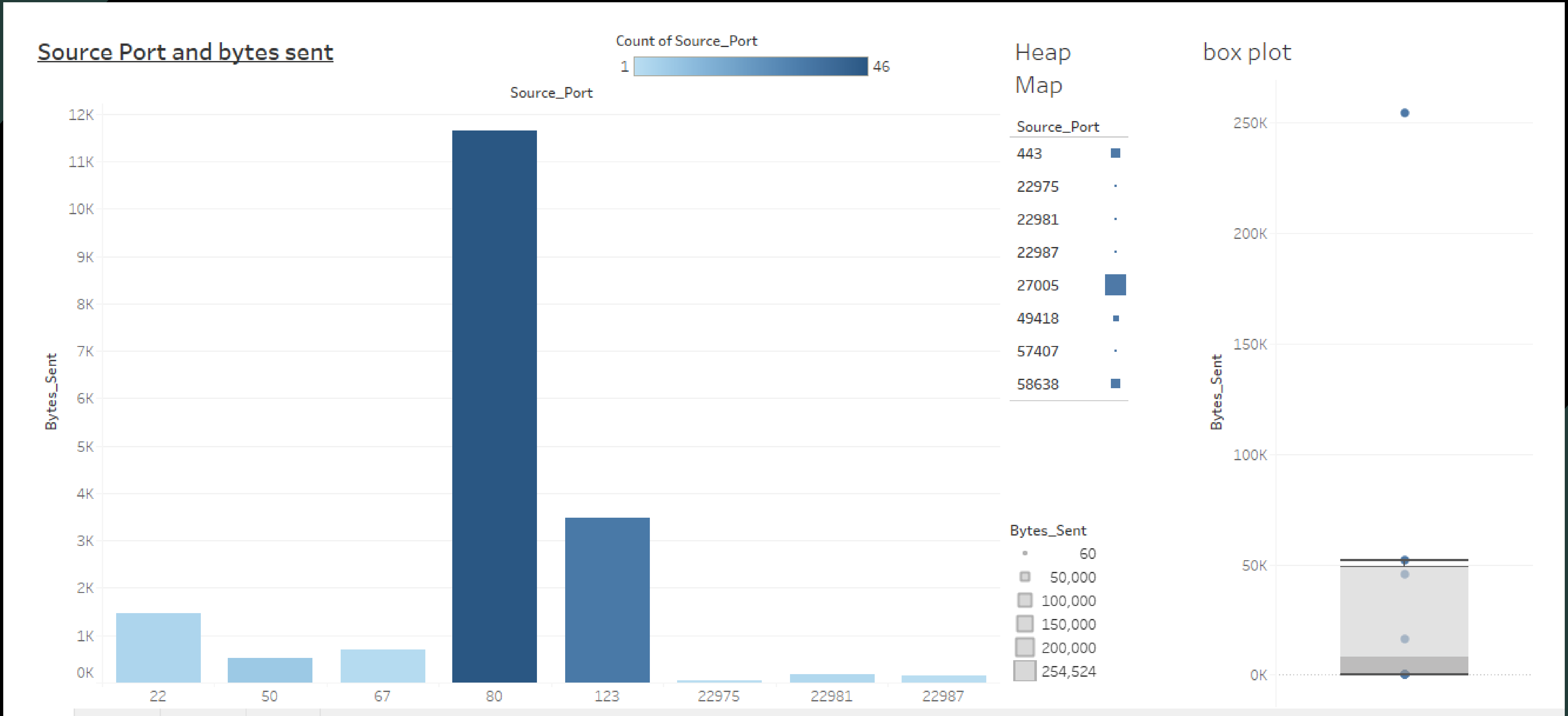
Is there a pattern between the sending and receival of packets – a repeating activity.

CHARTS AND ANALYSIS

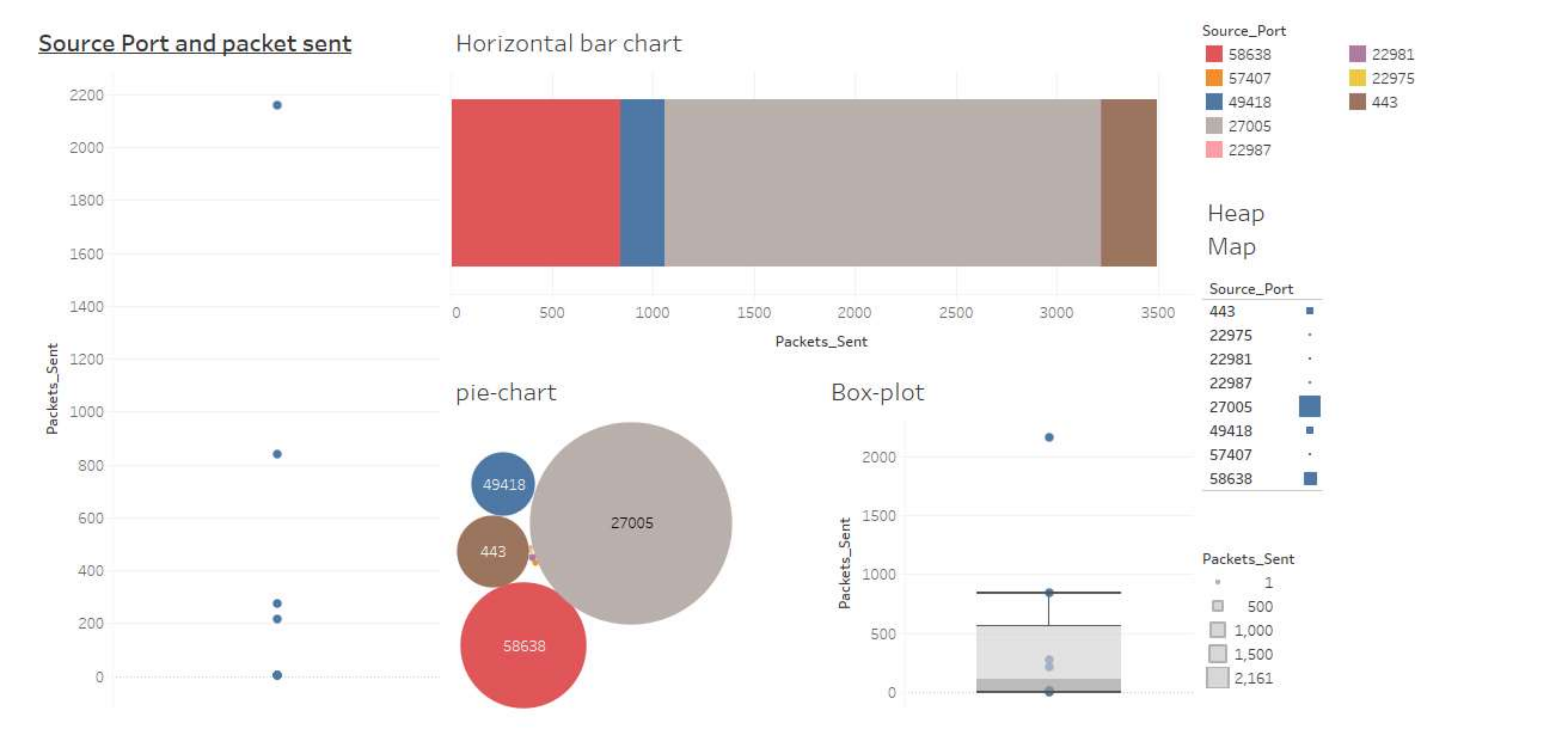
Step 6: Draw BI charts in Tableau

and explain each page (how it solves the problem).

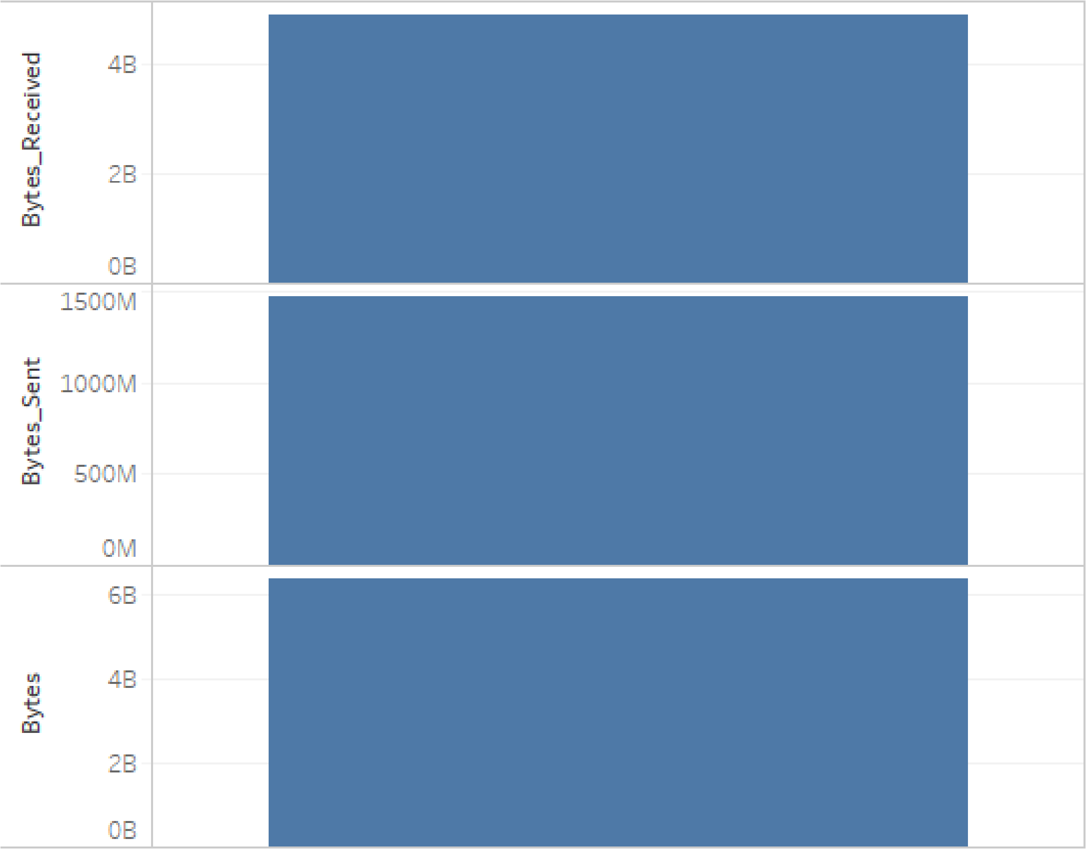
The source port in plotted against bytes sent by each of it. Random ports are plotted. The same thing is displayed through heat map and box plot. We see from random source port some have a lot of bytes sent other not so much this shows that these source port are busy or are regular senders. This also shows the important source port. Other assumption be that that they are combined meaning not one person own them rather a lot of parties receive information from them.



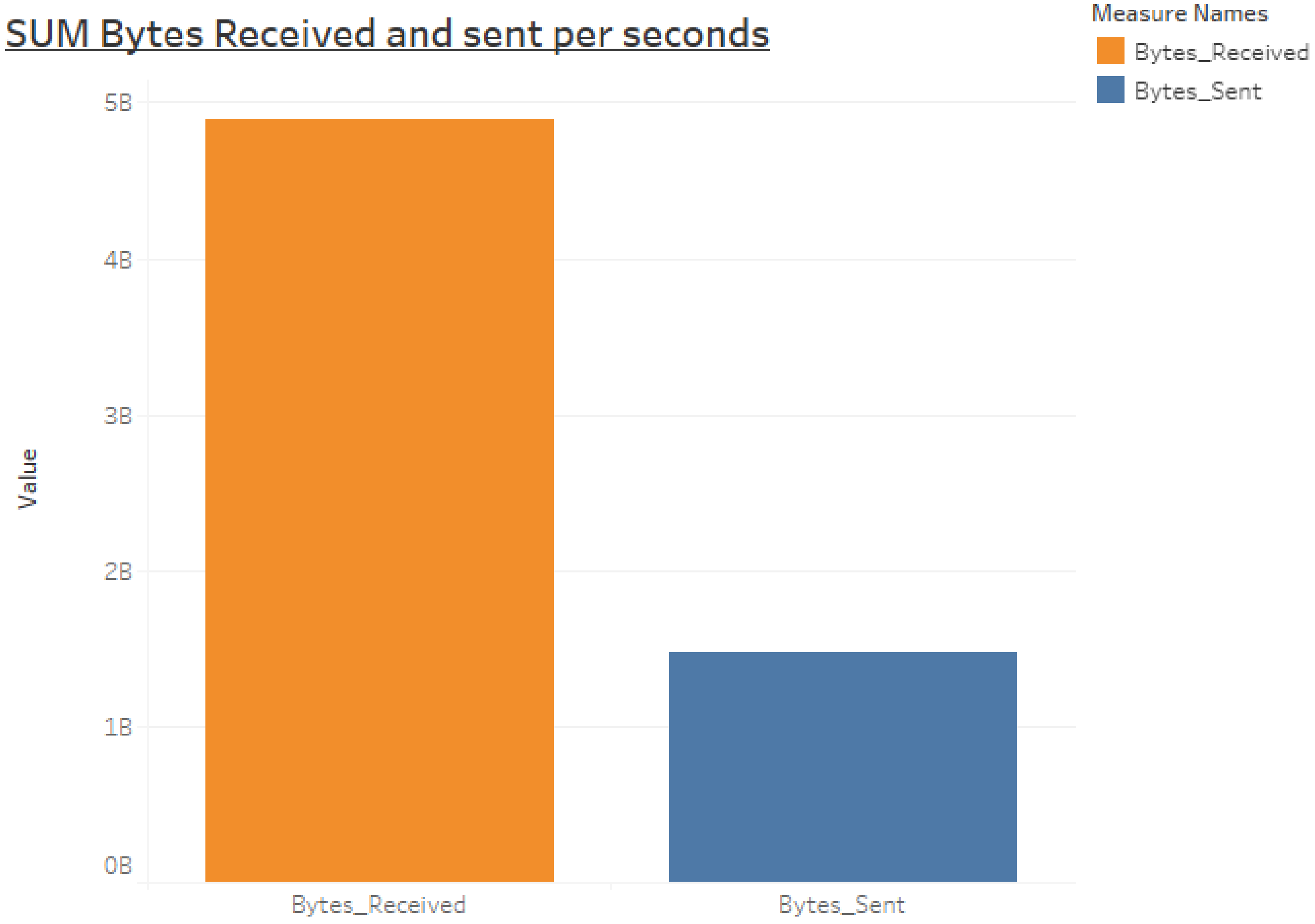
The source port in plotted against packet sent. Here we see use the plot as mentioned in the document above with largest times existence. So, we see a pattern where source port 27005 sent the most sum of packets over the span of the data. This again can mean rather than been a private port its public. The box plot also shows this port to be an outlier. The second leading is 58638 which also be a public sending server port from data is received by numerous regular customers. The firewall examine these ports the most.



SUM Byte Comparision between sent, received and total

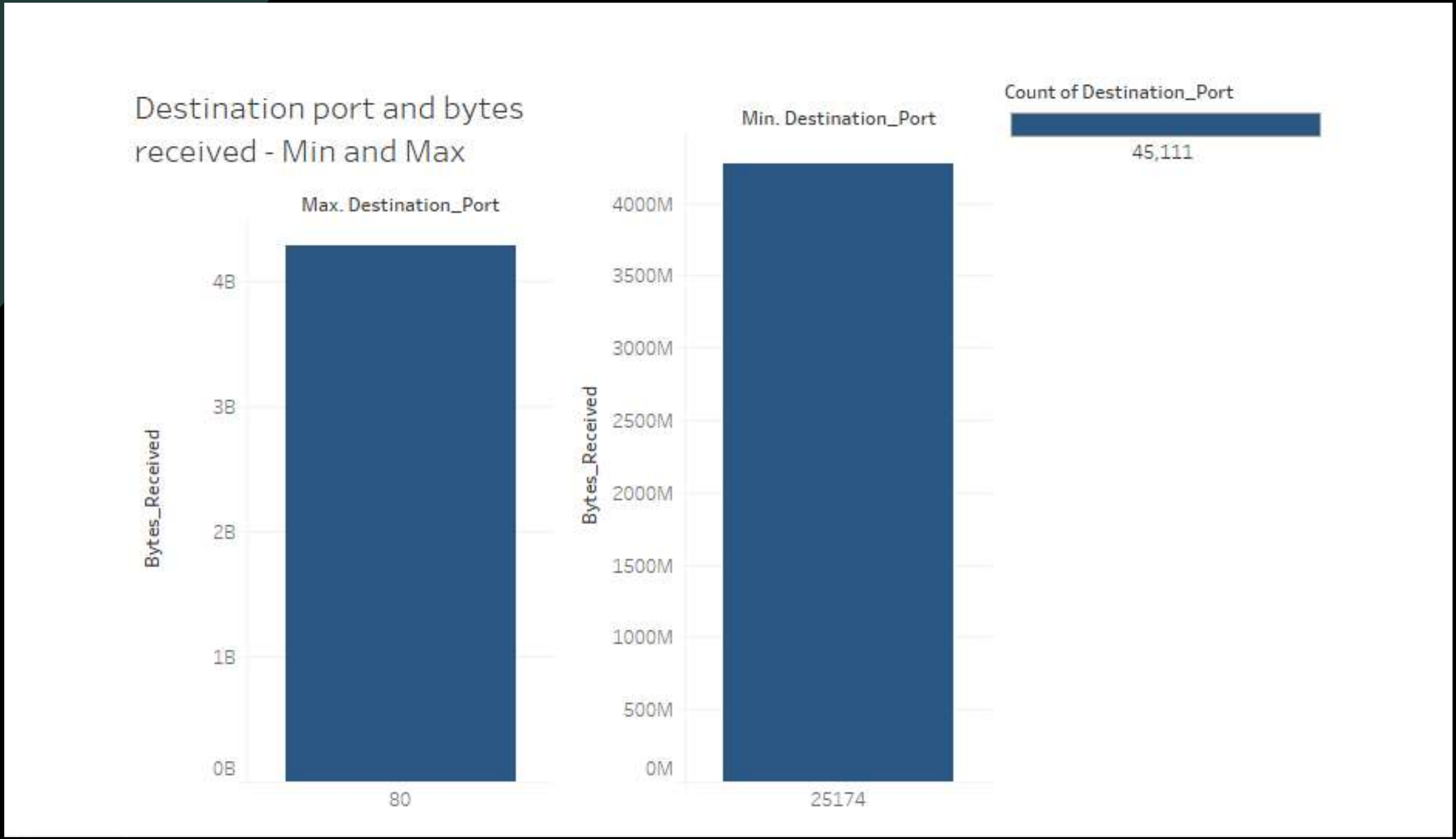


SUM Bytes Received and sent per seconds

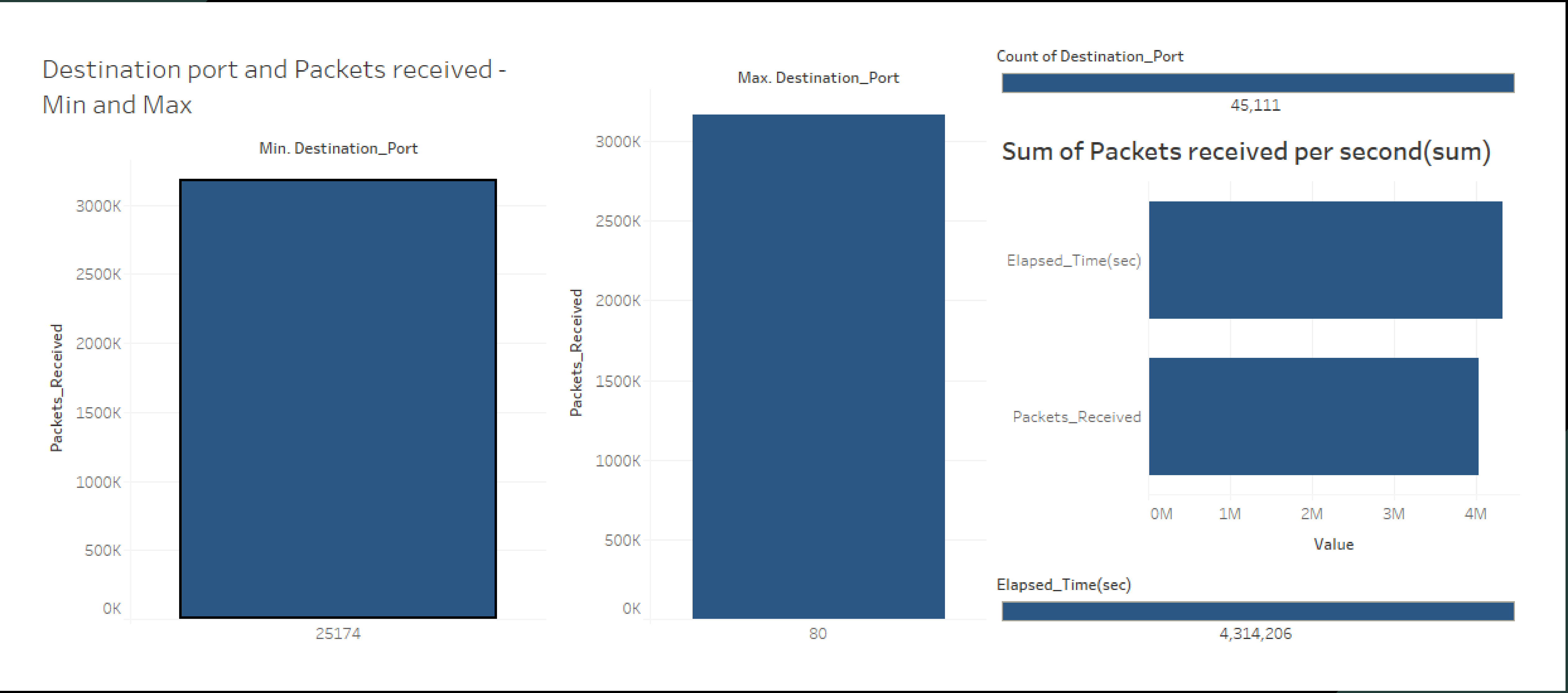


On previous slide -- The sum of byte – total, sent and received are compared. The sent bytes are 1500M and received are 4b there sum is the total. The socking thing is we are receiving more which means there are some port from which are getting receiving data, but we are not able to record their sources. This can be alarming in a firewall data. And then for the sum of elapsed time we compute the sum of bytes sent and received and again see the same pattern. These lead to alarming signs. There are people receiving with no source data being sent to them.

Another explanation can be that one sender is simply providing to many user again forming the idea of a public port.



Here we see the maximum and minimum value of the bytes received by a destination port. The maximum is received a port number 80. This led to the idea that port 80 has either some commercial or this port is shared a multiple different devices. And further routed by NAT destination port. The port with minimum number of bytes receives shows a normal pattern.

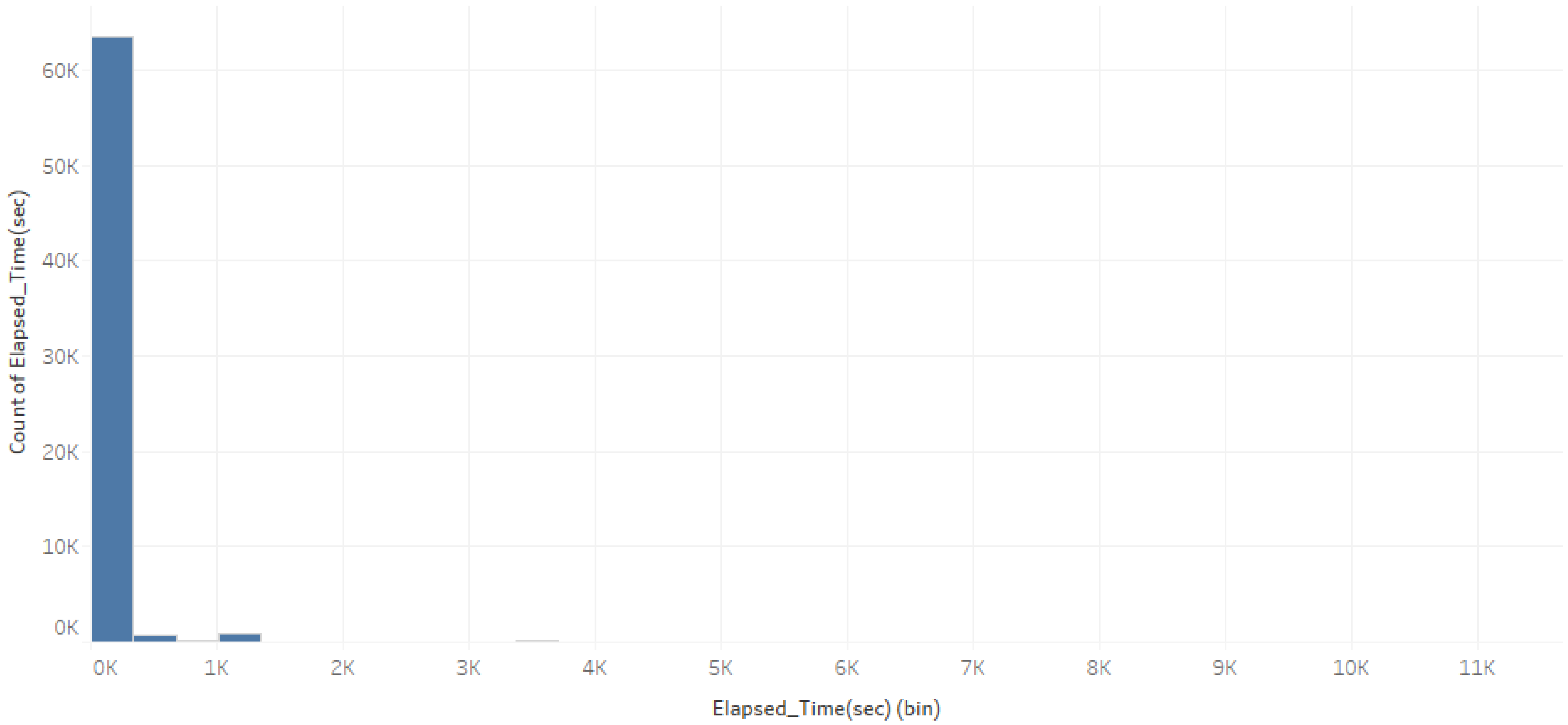


Destination port and Packets received - Min and Max

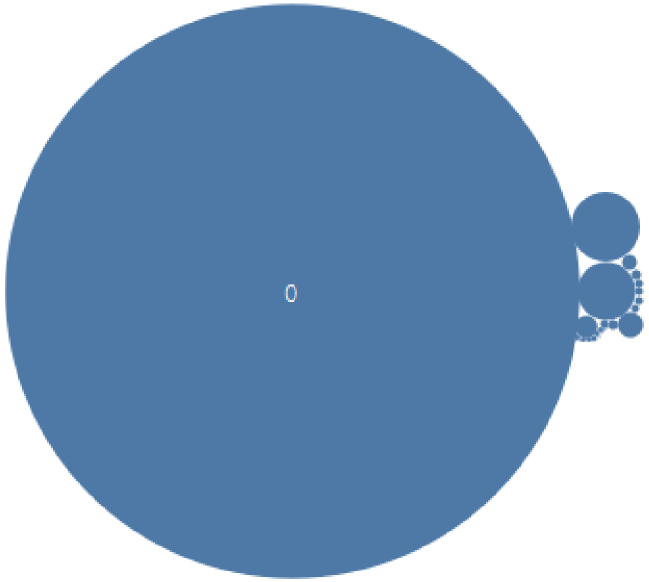
- **Same thing is plotted for packets received by a port. Both port from our list before as computed using python wrangling, we see which port receives the most packets and which receives the least.**
- **25174 receives least number of packets which is quite normal as we have seen before.**
- **But the astonishing fat is port 80 receives the greatest number of packets. This is simply due to the fact the greatest number of bytes were received there so the greatest number of packets belong there.**
- **This also clearly stated the ide more the bytes more the number of the packets.**

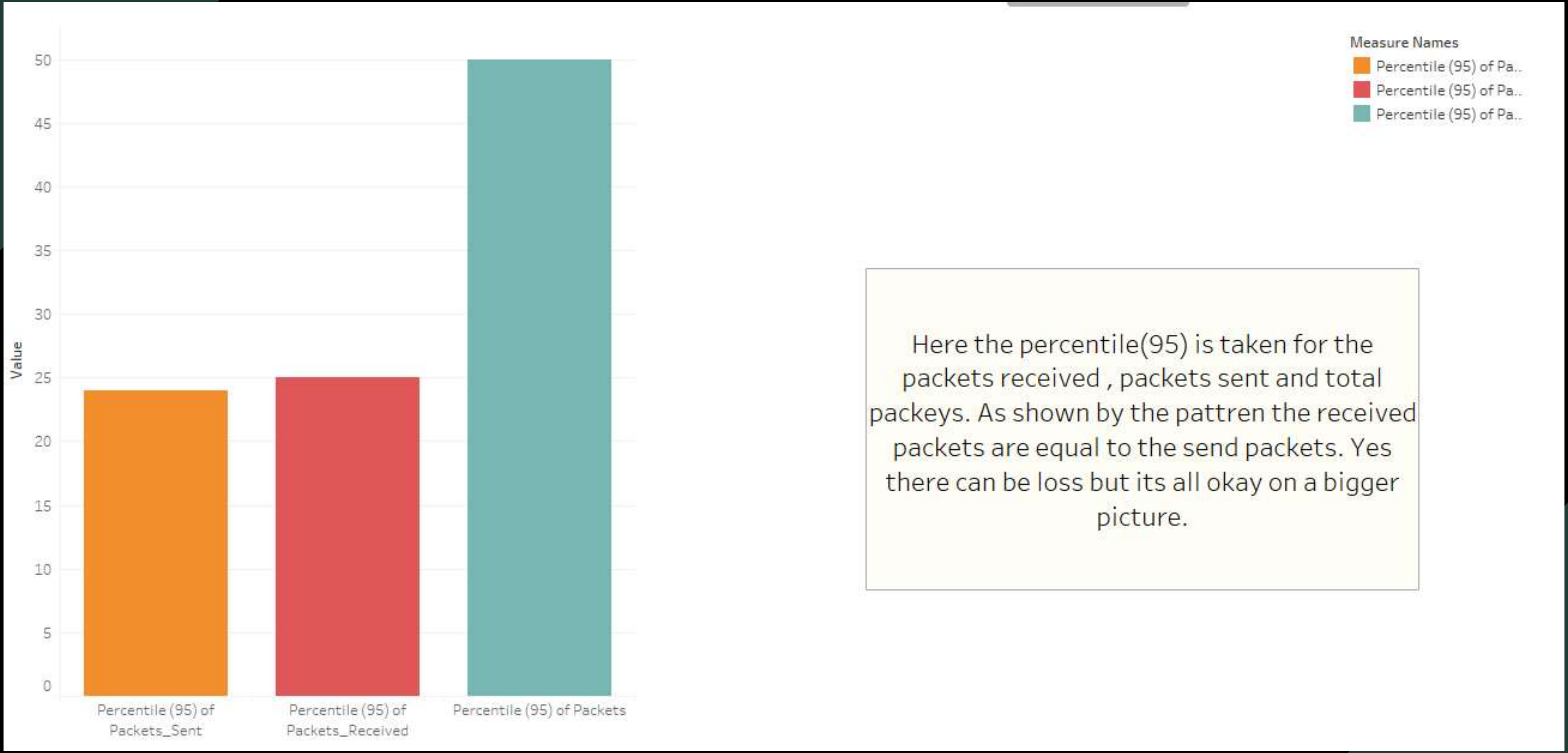
We receives most packets in 0 elapsed time this due to either elapsed time is not stored properly, or the packets travel in milli second and second are not sufficiently to be recording time. Both bar chart distribution and clustered pie- chart shows it.

Sum of Packets received per second(sum)



Sum of Pakets received per second(sum)

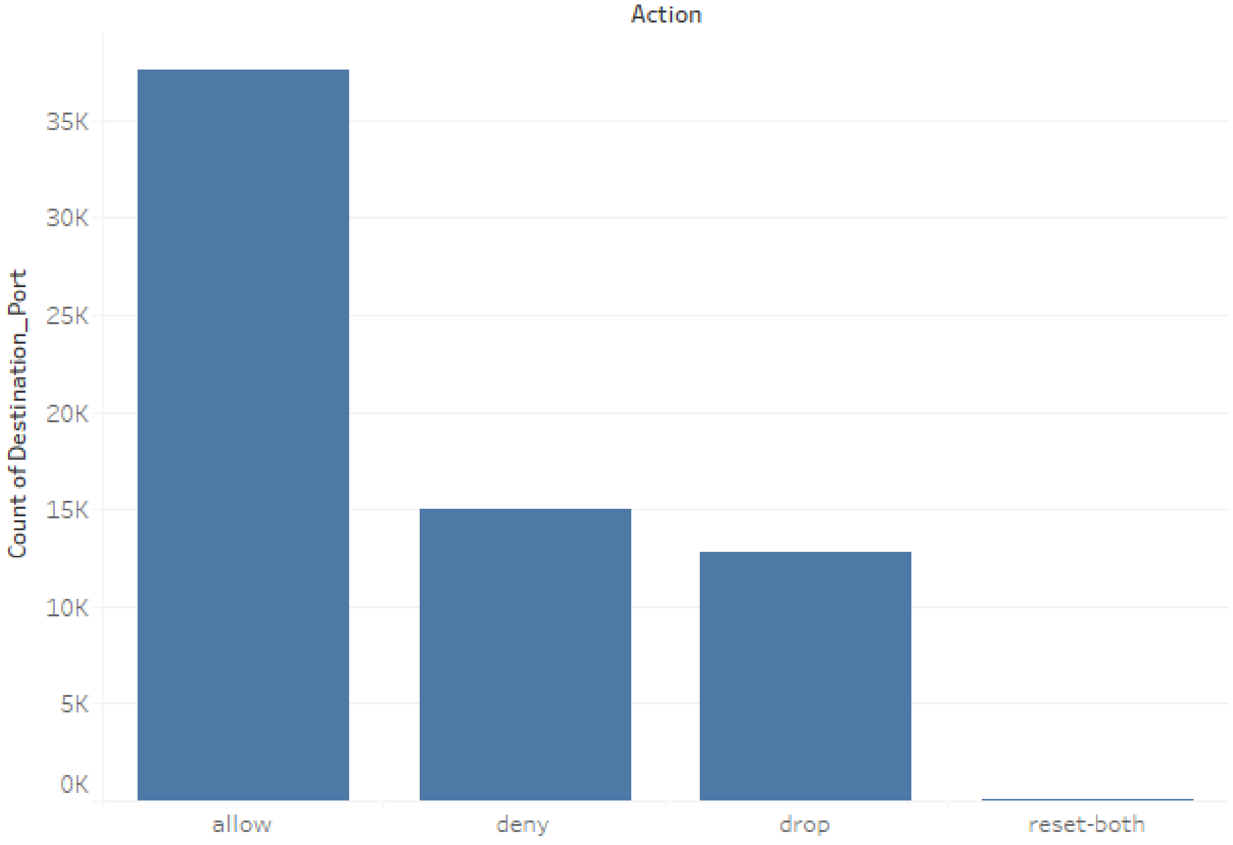




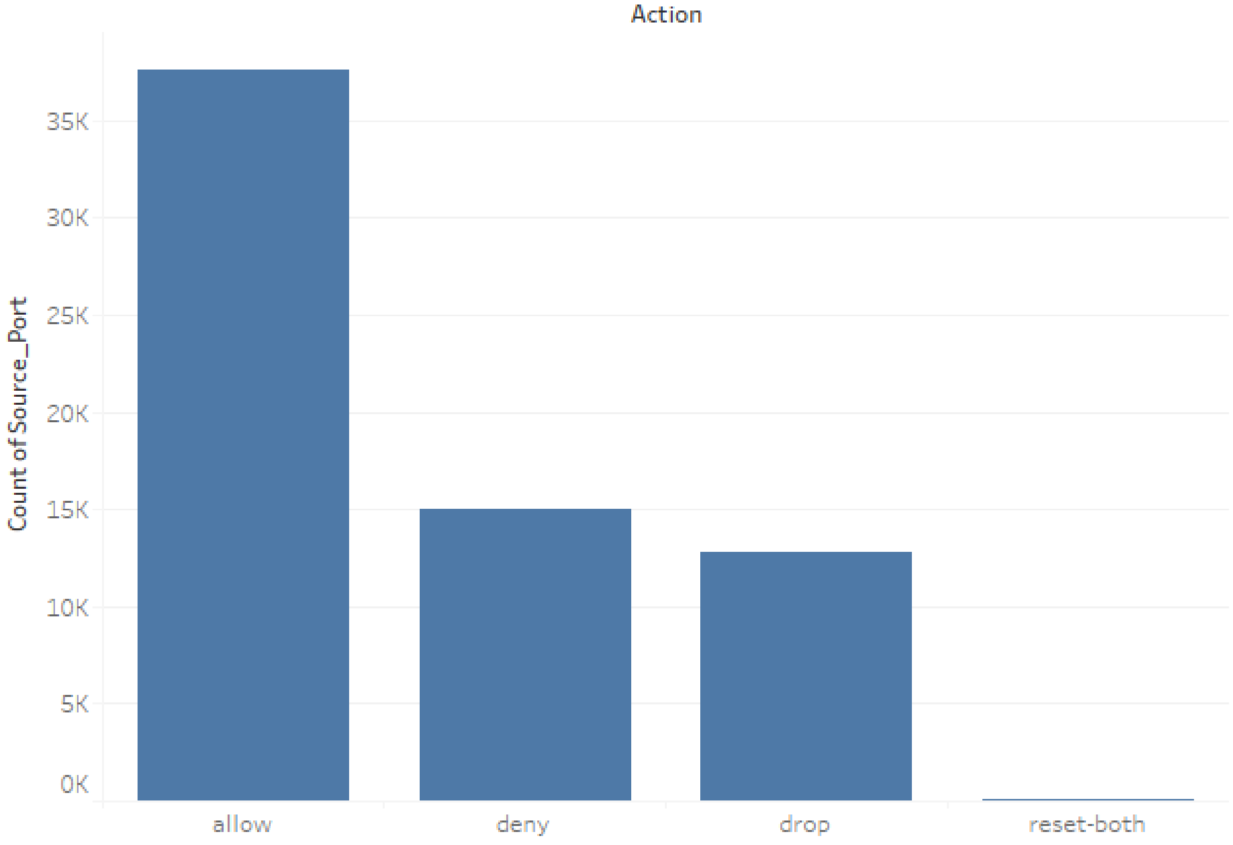
Here the percentile(95) is taken for the packets received , packets sent and total packeys. As shown by the pattren the received packets are equal to the send packets. Yes there can be loss but its all okay on a bigger picture.

This shows the action according to the rules of the firewall. The maximum time form both the source port and the destination port the action is allows. This show most packets and bytes are allowed to go and received.

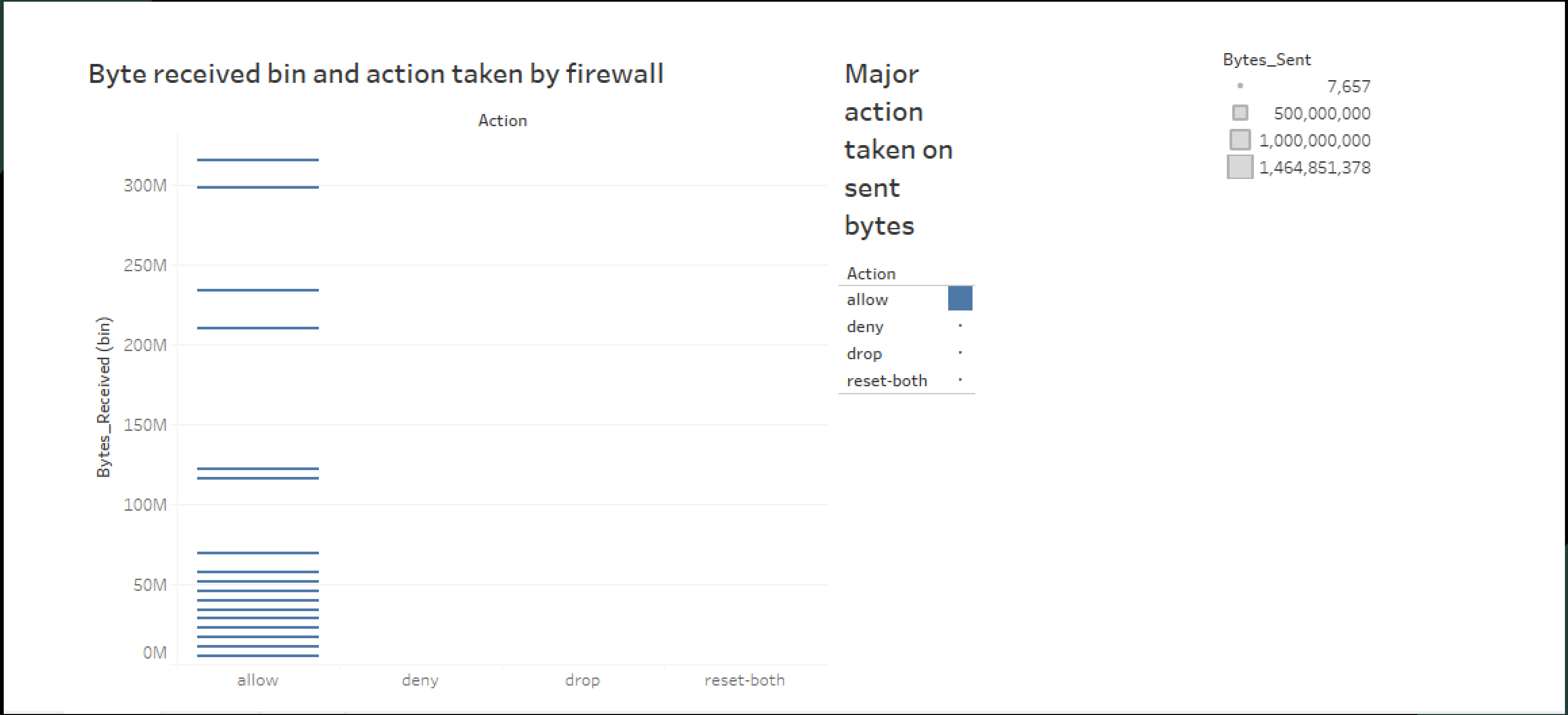
Destination port and Action



Source port and Action



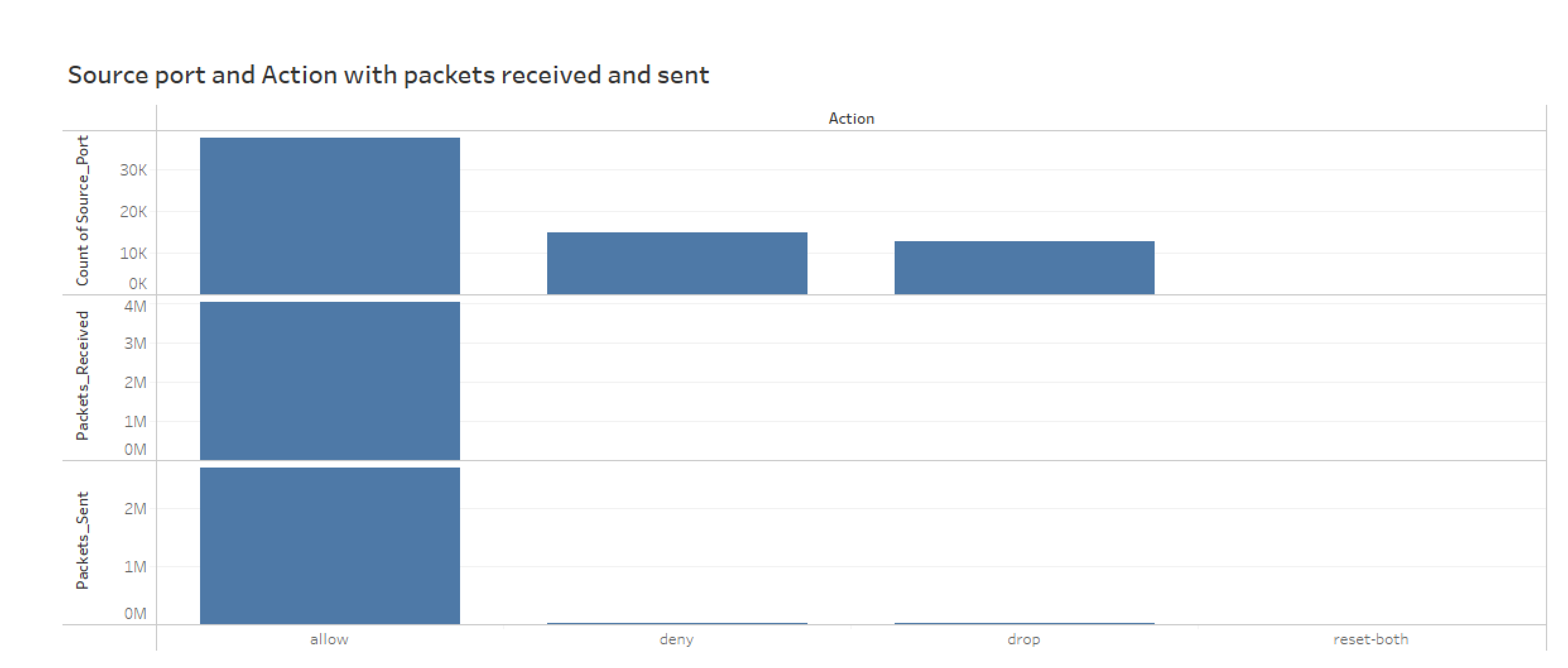
This also shows that the most bits are allowed to travel



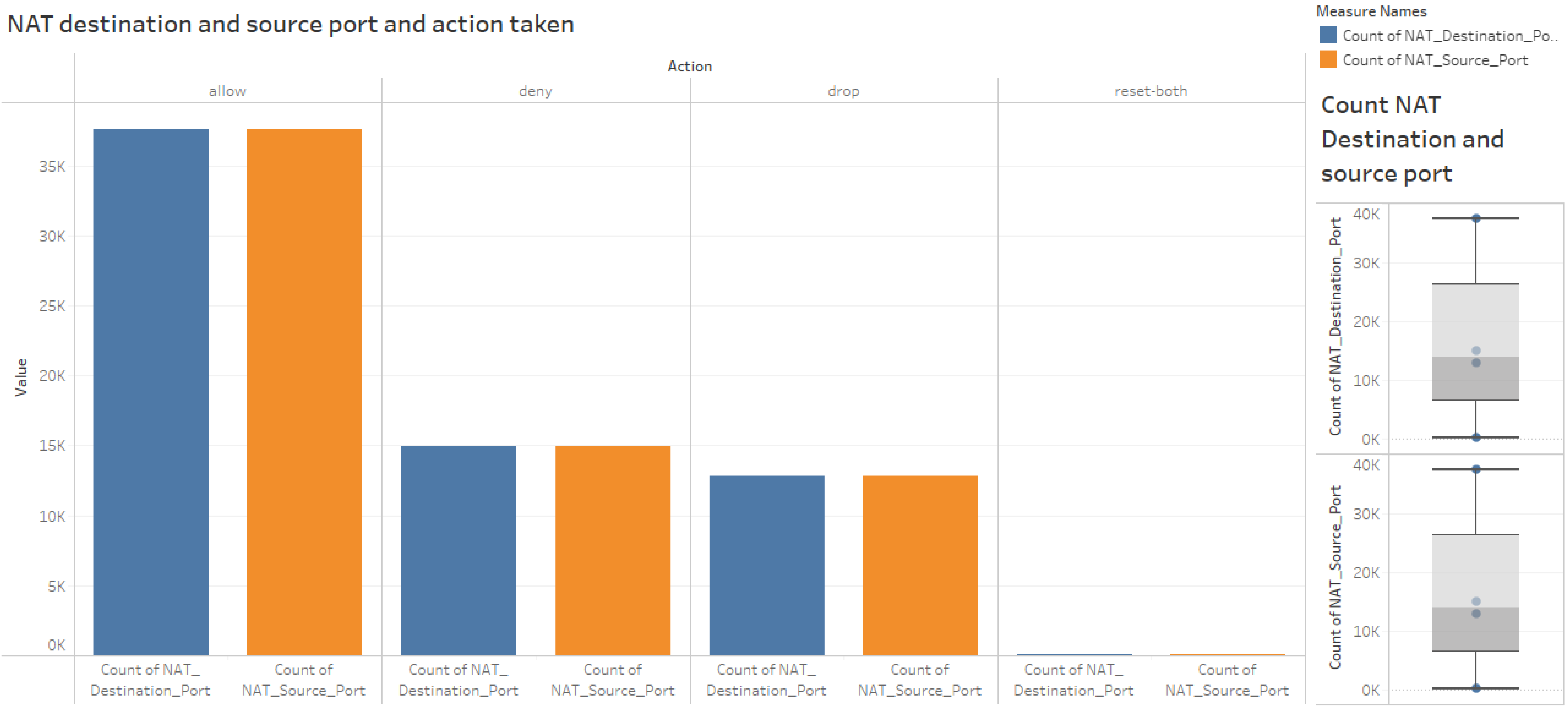
Byte received bin and action taken by firewall

Major action taken on sent bytes

This shows the action on the source port. And the comparison on the action on the packets being send and received. As before most followed action is allowed and deny and drop the reset both option is very least in count. This simply shows that many source port are sending multiple packets. May be even on the same time.



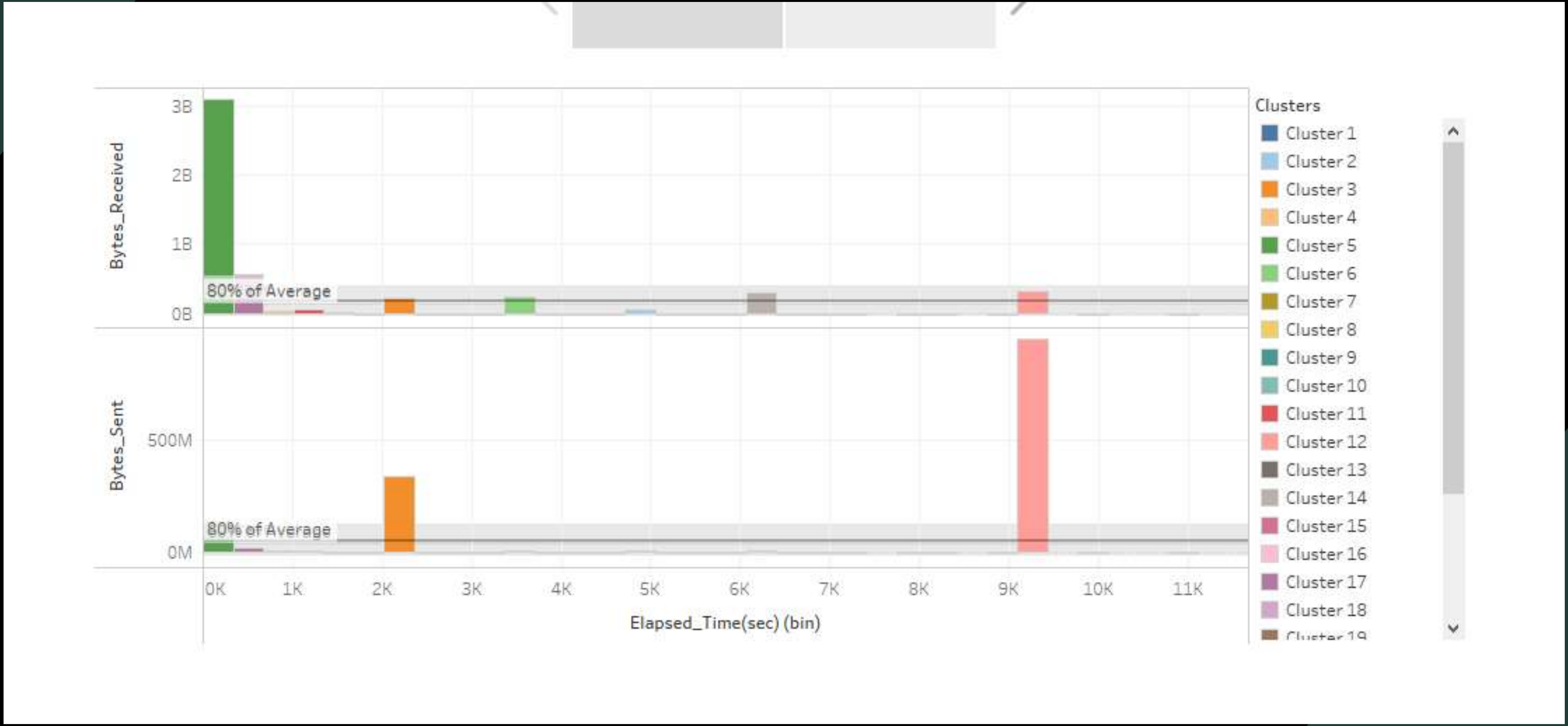
This slide compare the NAT destination source and destination. We see the same result as before the allowed action still remained most prominent. May be the firewall need to stop more going packest to control traffic and security. The box-plot also shows the allow as an outlier.



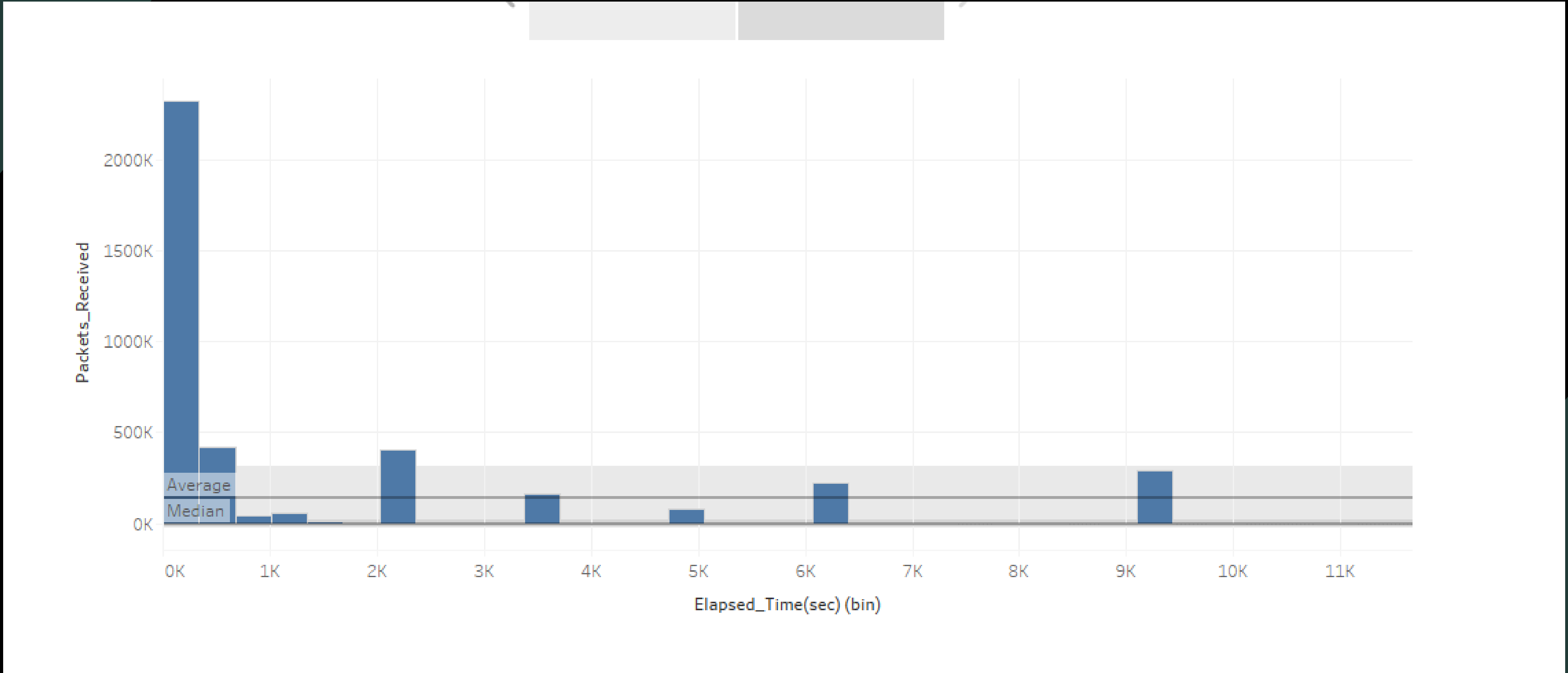
NAT destination and source port and action taken

Count NAT Destination and source port

This shows the clusters formed for the elapsed time – bin and the bytes received and sent. As we see previously the most bytes are sent are in 0 elapsed time. So we make the cluster distribution to analyze further. We still see the cluster one to be most prominent which shows zero.



This also shows the same pattern on zero elapsed time on the packets received. Showing clearly the relation between attributes, Bytes and packets.



END