Clusters explainability and analysis

Introduction

In this section of the report, we will focus on the analysis and interpretation of the clusters identified in the previous section. The main objectives include the characterization of the clusters in terms of the distribution of features and activity patterns, and the formulation of considerations related to the analysis of darknet traffic.

The primary goal is to examine the detected clusters and identify new patterns. This analysis will pursue four main tasks:

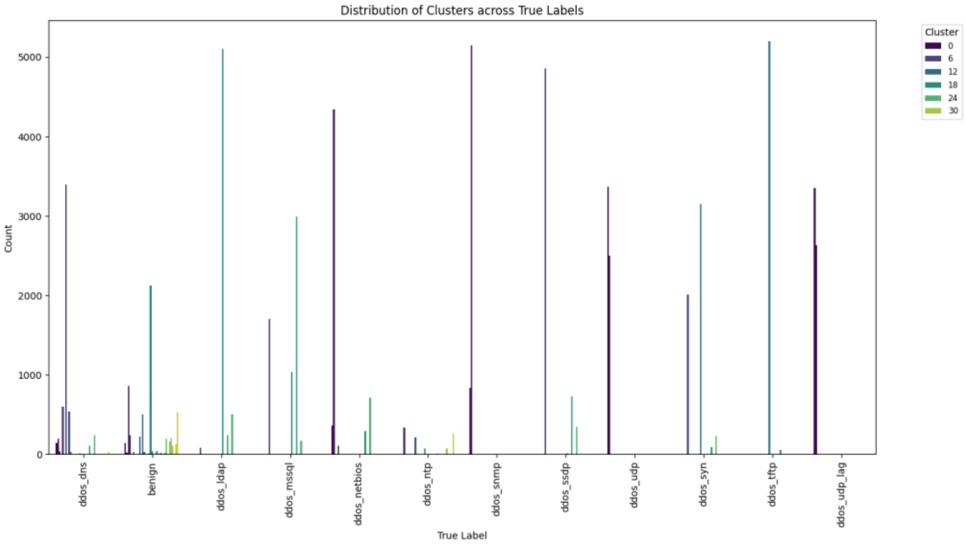
1. Reflection on clusters with respect to GT labels: We will evaluate whether the clusters reflect the ground truth (GT) labels by calculating the empirical cumulative distribution function (ECDF) of the number of clusters assigned to each class. We will verify the existence of pure clusters, in which all elements belong to a single class, and analyze whether there is benign traffic with characteristics similar to malicious traffic.
2. Identification of the most important features in the obtained clusters: We will use methods that provide feature importance or explainability techniques to identify the most relevant features within the clusters, helping us understand which features most influence cluster formation.
3. Identification of sub-attacks: We will try to identify any sub-attacks within the clusters by analyzing the features that contribute to this identification. Additionally, we will explore the possibility of identifying new groups or similar clusters and understand the characteristics that form them.
4. Similarity between attacks: We will analyze which attacks are most similar to each other and according to which features, to better understand the nature of the attacks and identify common patterns in malicious traffic.

The ultimate goal is to provide an in-depth understanding of the detected clusters, highlighting the dynamics of darknet traffic and contributing to the improvement of threat detection techniques. For the cluster analysis, we will compare the K-means and DBSCAN methods. K-means assumes spherical clusters of similar size and requires the number of clusters to be specified a priori. In contrast, DBSCAN identifies clusters based on point density, not requiring the number of clusters a priori and better handling arbitrary shapes and noise. DBSCAN is particularly useful for analyzing darknet traffic, characterized by significant variations in data density and the presence of noise, offering a more comprehensive and robust view of the patterns in the data.

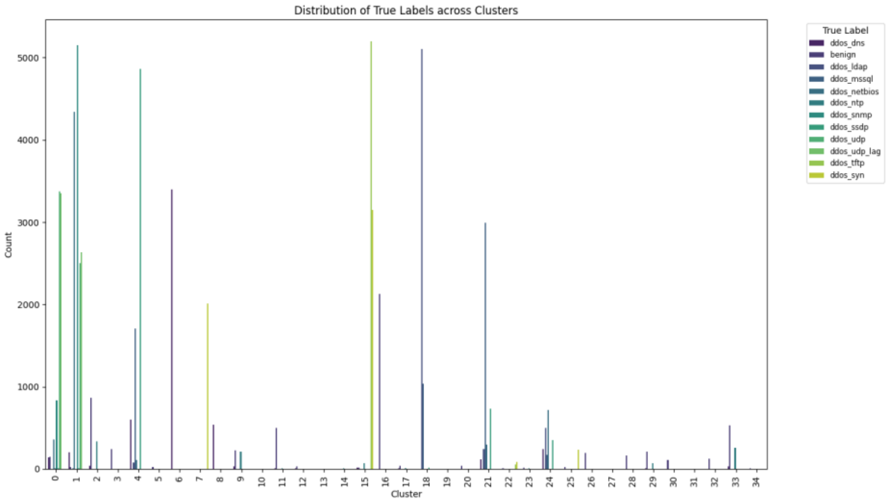
K-means

Distribution of clusters in relation to the ground truth

To begin with, we visualized the distribution of clusters in relation to the ground truth (GT) labels to gain an initial understanding of how the identified clusters reflect the labels.

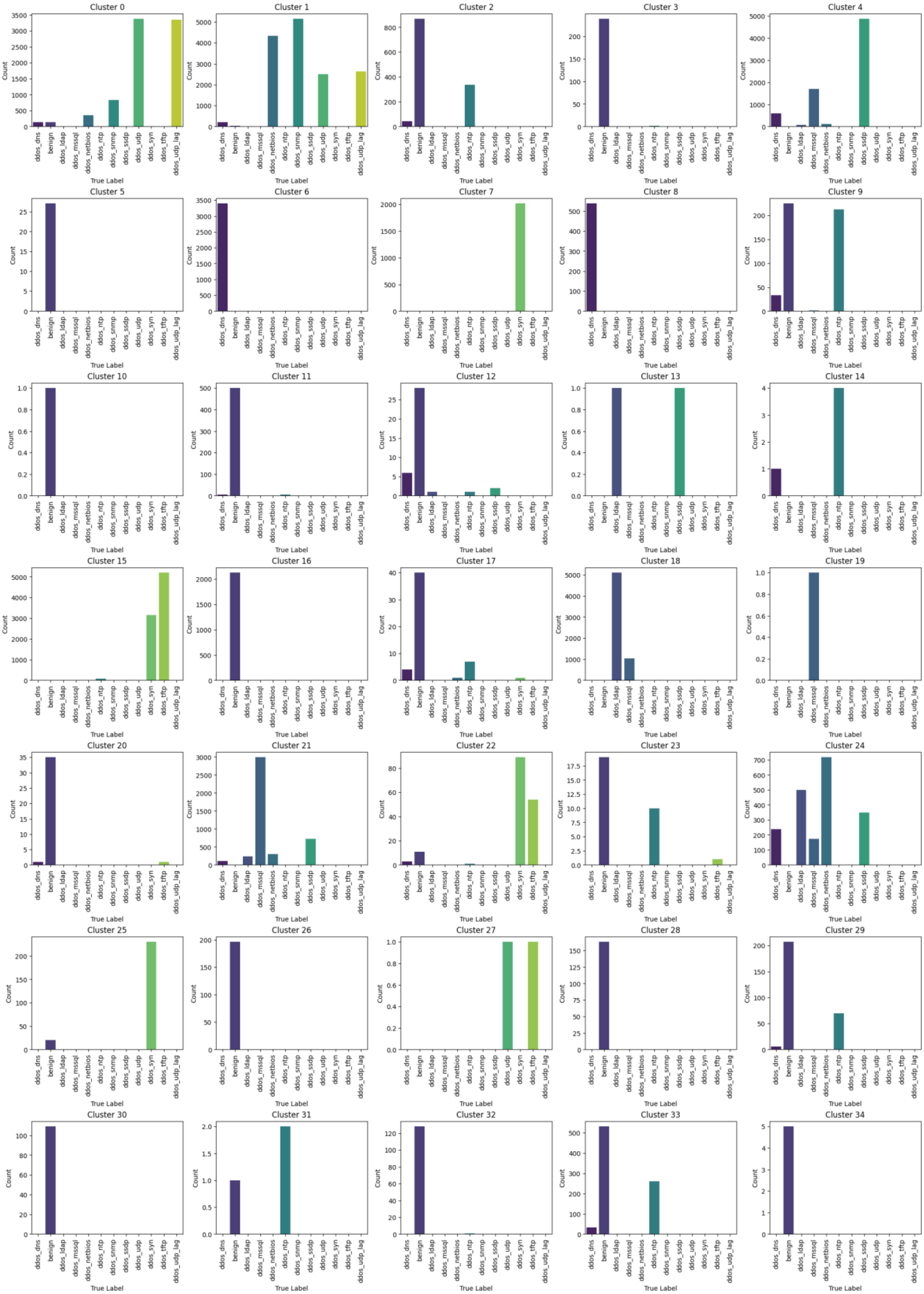


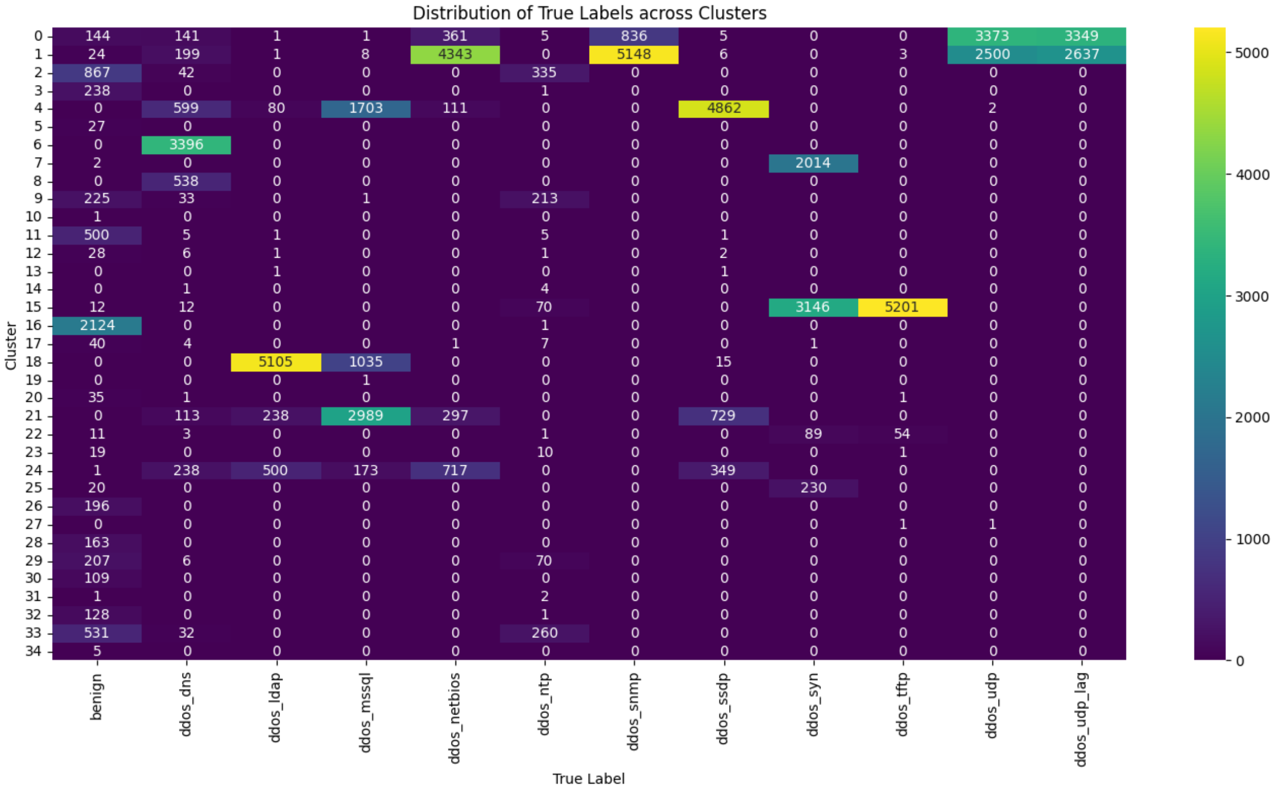
The first figure shows the distribution of various clusters in relation to the ground truth labels. This visualization facilitates the identification of labels composed of a mixture of different clusters, as is evident for the benign and ddos_dns labels represented by different clusters.



The second figure shows the inverse visualization, i.e., how the ground truth labels are distributed with respect to the clusters. This visualization facilitates the identification of clusters that primarily contain samples of a specific label. For example, it is evident that cluster 18 mainly contains samples belonging to the ddos_ldap label, or that cluster 6 only contains ddos_dns labels.

Given the large number of clusters and labels, this visualization does not allow for capturing all the nuances of interest. Therefore, the distribution of all labels within each individual cluster and the corresponding contingency matrix are presented below to achieve a much clearer understanding.


Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10
Cluster 11
Cluster 12
Cluster 13
Cluster 14
Cluster 15
Cluster 16
Cluster 17
Cluster 18
Cluster 19
Cluster 20
Cluster 21
Cluster 22
Cluster 23
Cluster 24
Cluster 25
Cluster 26
Cluster 27
Cluster 28
Cluster 29
Cluster 30
Cluster 31
Cluster 32
Cluster 33
Cluster 34
Count
True Label
ddos_dns
benign
ddos_ldap
ddos_mssql
ddos_netbios
ddos_ntp
ddos_snmp
ddos_ssdp
ddos_udp
ddos_syn
ddos_tftp
ddos_udp_lag



Distribution of True Labels across Clusters

With reference to the two graphical representations, a number of considerations can be made.

Noise (less than 10 samples per cluster):

Cluster	10	13	14	19	27	31	34

Important note: in all visualizations the noise clusters will still be present, but they will be ignored.

Pure (or nearly pure) clusters:

Cluster	3	5	6	7	8	11	12
Label	benign	benign	ddos_dns	ddos_syn	ddos_dns	benign	benign

Cluster	16	17	20	25	26	28	30	32
Label	benign	benign	benign	ddos_syn	benign	benign	benign	benign

As was reasonably expected, the analysis revealed the presence of numerous clusters representing benign traffic. This phenomenon can be explained by considering that generic traffic is inherently diverse, characterized by a wide range of attributes that reflect the different ways in which users navigate and interact on the network. This diversity results in a multiplicity of traffic patterns, each of which can be distinctly categorized within a specific cluster.

Looking at the figure, it can be seen that some specific clusters, such as 2, 9, 23 and 33, exclusively include traffic classified as benign and ddos_ntp type attacks, the latter known to be an underrepresented class in the data. To understand the reason for this distinct aggregation, it is essential to conduct a deeper analysis of the key characteristics that define these clusters.

Investigation of the most relevant characteristics may reveal distinctive patterns or common attributes that justify the coexistence of benign traffic and ddos_ntp attacks within the same clusters.

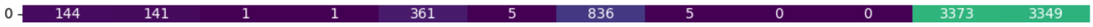
At this point in the discussion, although it is possible to start making preliminary remarks regarding underrepresented attacks and similarities between the different clusters, it is considered more appropriate to defer detailed analysis of these aspects to later sections of the paper.

Cluster mapping

From the initial analysis, it emerged that the clusters more or less tend to reflect the assigned labels. However, to conduct a more meticulous investigation, we adopted a specific approach. Using a technique known as cluster mapping, we aimed to overcome the challenge presented by the coexistence of two or more labels within single clusters, as observed, for example, in cluster 0. The goal was to avoid automatically selecting the most frequent label without considering other significantly represented labels.

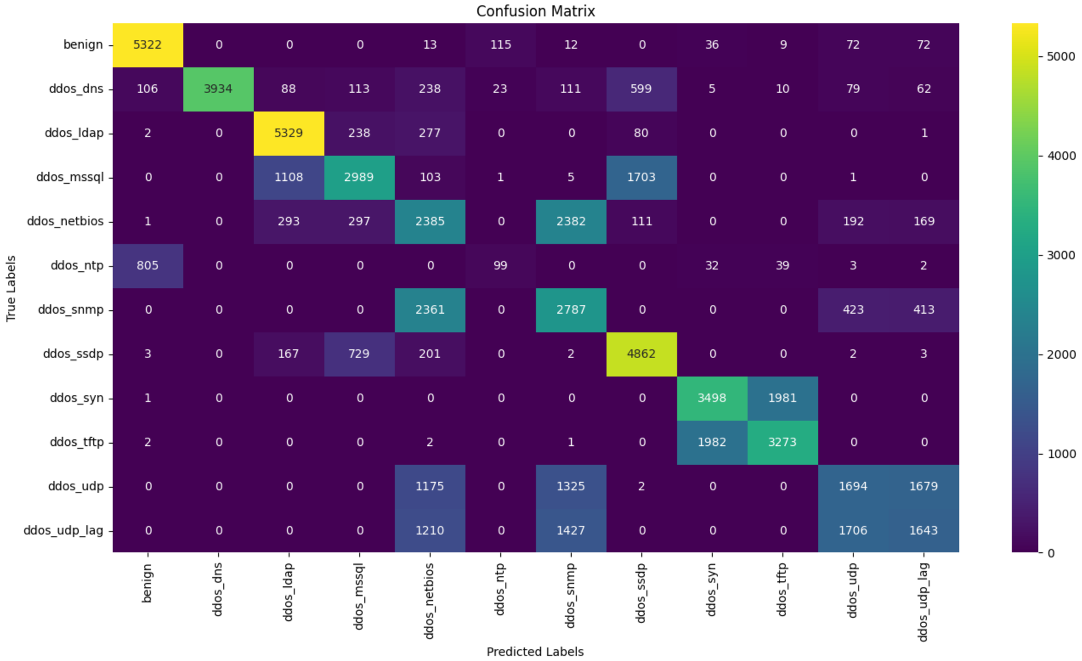
For this, we developed a script that analyzes each cluster by identifying the frequencies of the various labels present. The algorithm selects the label with the highest frequency and, concurrently, identifies and includes all those labels whose frequency is at least 60% of the highest recorded value.

Applying this method to cluster 0, the algorithm selected labels with frequencies of 3373 and 3349, excluding those with frequencies below 60% of 3373.



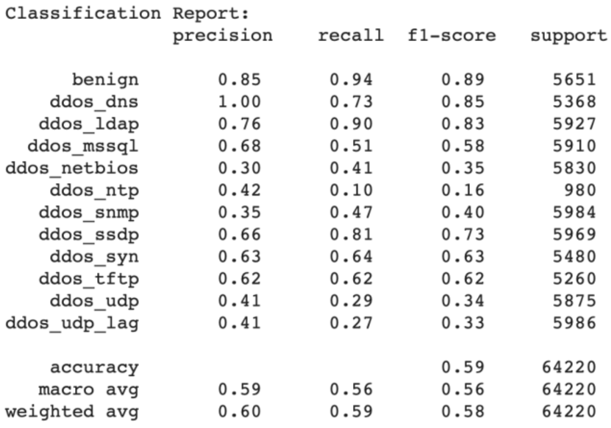
0	144	141	1	1	361	5	836	5	0	0	3373	3349

After defining the functioning of the algorithm, we move on to evaluating the results of this mapping. We use the confusion matrix as an evaluation tool. The confusion matrix allows us to visualize the effectiveness with which the algorithm has categorized traffic in the clusters relative to the original labels, thus highlighting the accuracy and any discrepancies in classification. This analysis will provide us with a clear picture of the accuracy of our mapping strategy and help us identify areas that could benefit from further improvements or a different approach.



Confusion Matrix

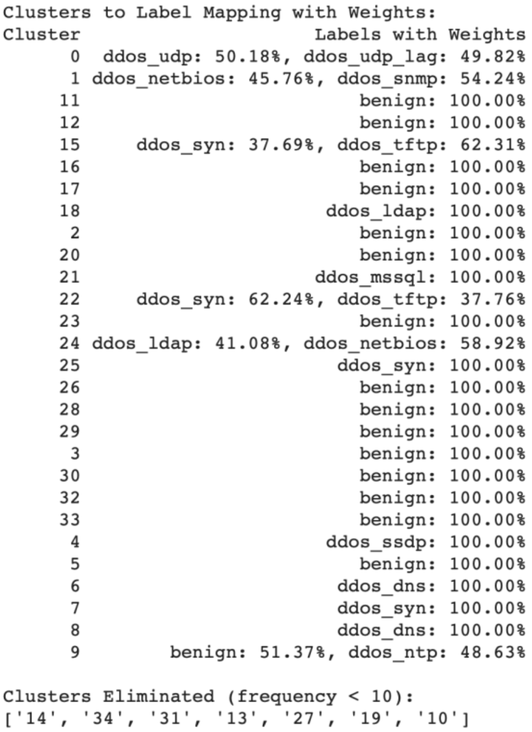
True Labels \ Predicted Labels	benign	ddos_dns	ddos_ldap	ddos_mssql	ddos_netbios	ddos_ntp	ddos_snmp	ddos_ssdp	ddos_syn	ddos_tftp	ddos_udp	ddos_udp_lag
benign	5322	0	0	0	13	115	12	0	36	9	72	72
ddos_dns	106	3934	88	113	238	23	111	599	5	10	79	62
ddos_ldap	2	0	5329	238	277	0	0	80	0	0	0	1
ddos_mssql	0	0	1108	2989	103	1	5	1703	0	0	1	0
ddos_netbios	1	0	293	297	2385	0	2382	111	0	0	192	169
ddos_ntp	805	0	0	0	0	99	0	0	32	39	3	2
ddos_snmp	0	0	0	0	2361	0	2787	0	0	0	423	413
ddos_ssdp	3	0	167	729	201	0	2	4862	0	0	2	3
ddos_syn	1	0	0	0	0	0	0	0	3498	1981	0	0
ddos_tftp	2	0	0	0	2	0	1	0	1982	3273	0	0
ddos_udp	0	0	0	0	1175	0	1325	2	0	0	1694	1679
ddos_udp_lag	0	0	0	0	1210	0	1427	0	0	0	1706	1643



```
Classification Report:
               precision    recall  f1-score   support

      benign       0.85      0.94      0.89      5651
    ddos_dns       1.00      0.73      0.85      5368
   ddos_ldap       0.76      0.90      0.83      5927
  ddos_mssql       0.68      0.51      0.58      5910
ddos_netbios       0.30      0.41      0.35      5830
    ddos_ntp       0.42      0.10      0.16       980
   ddos_snmp       0.35      0.47      0.40      5984
   ddos_ssdp       0.66      0.81      0.73      5969
    ddos_syn       0.63      0.64      0.63      5480
   ddos_tftp       0.62      0.62      0.62      5260
    ddos_udp       0.41      0.29      0.34      5875
ddos_udp_lag       0.41      0.27      0.33      5986

    accuracy                           0.59     64220
   macro avg       0.59      0.56      0.56     64220
weighted avg       0.60      0.59      0.58     64220
```



```
Clusters to Label Mapping with Weights:
Cluster                        Labels with Weights
      0  ddos_udp: 50.18%, ddos_udp_lag: 49.82%
      1 ddos_netbios: 45.76%, ddos_snmp: 54.24%
     11                          benign: 100.00%
     12                          benign: 100.00%
     15    ddos_syn: 37.69%, ddos_tftp: 62.31%
     16                          benign: 100.00%
     17                          benign: 100.00%
     18                       ddos_ldap: 100.00%
      2                          benign: 100.00%
     20                          benign: 100.00%
     21                      ddos_mssql: 100.00%
     22    ddos_syn: 62.24%, ddos_tftp: 37.76%
     23                          benign: 100.00%
     24 ddos_ldap: 41.08%, ddos_netbios: 58.92%
     25                        ddos_syn: 100.00%
     26                          benign: 100.00%
     28                          benign: 100.00%
     29                          benign: 100.00%
      3                          benign: 100.00%
     30                          benign: 100.00%
     32                          benign: 100.00%
     33                          benign: 100.00%
      4                       ddos_ssdp: 100.00%
      5                          benign: 100.00%
      6                        ddos_dns: 100.00%
      7                        ddos_syn: 100.00%
      8                        ddos_dns: 100.00%
      9       benign: 51.37%, ddos_ntp: 48.63%

Clusters Eliminated (frequency < 10):
['14', '34', '31', '13', '27', '19', '10']
```

From the data provided, it is evident that the clusters have been mapped with varying accuracy to different labels.

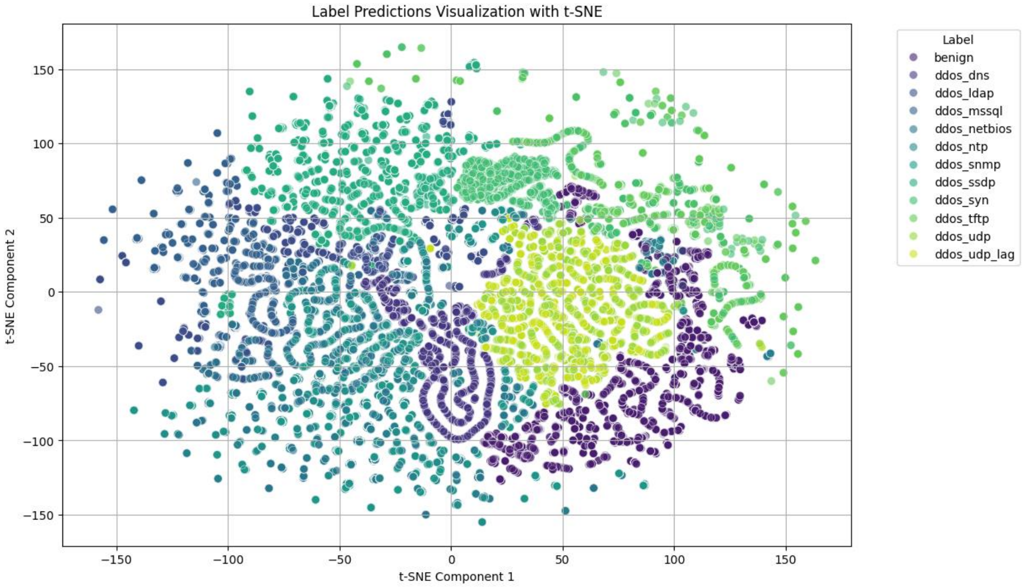
For example, cluster 0 shows an almost equal distribution between the labels ddos_udp and ddos_udp_lag, suggesting a similarity in the characteristic traits between these categories of traffic. On the other hand, clusters like 11 and 17 are clearly identified as benign traffic, with a 100% match. However, some clusters, like 9, present an almost even balance between contrasting labels such as benign and ddos_ntp, indicating the possibility of confusion or overlap in the traffic characteristics, or even a strong resemblance.

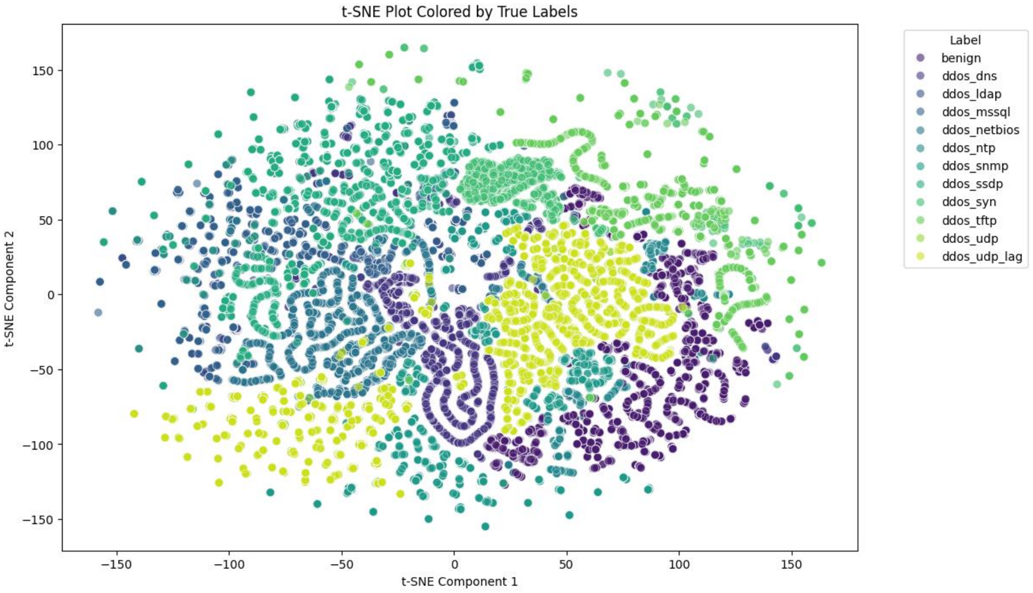
The 60% accuracy achieved through the k-means clustering algorithm represents an acceptable result considering its inherent limitations, especially in complex contexts like network traffic analysis. K-means tends to perform better with homogeneous and simple data.

The similarities observed between various pairs of attacks, such as ddos_syn and ddos_tftp, and between categories such as benign and ddos_ntp, as well as ddos_udp and ddos_udp_lag, indicate the presence of common characteristics that can induce confusion in the clustering process. These affinities may arise from similar traffic patterns, overlapping features, or a lack of capability of distance-based algorithms, like k-means, to distinguish between subtly different patterns.

Therefore, strong similarities are highlighted between ddos_syn and ddos_tftp, benign and ddos_ntp, ddos_udp and ddos_udp_lag, ddos_netbios and ddos_snmp, ddos_ldap and ddos_netbios.

To obtain a visual representation that contrasts the predicted label mapping with the actual labels, we applied the t-SNE technique to both datasets.





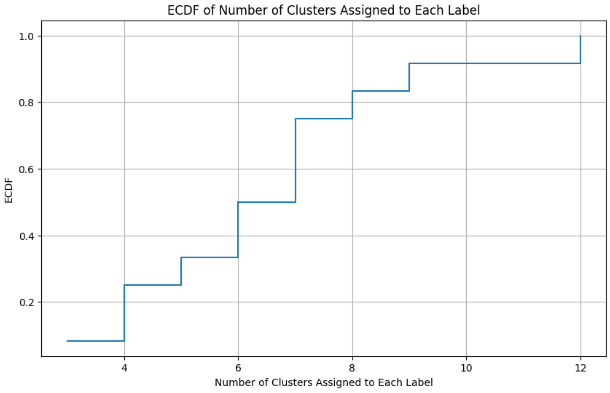
From the visual analysis of the t-SNE graphs, similarities and overlaps between the classifications of various traffic types clearly emerge. In particular, two cases of misclassification attract attention because of their implications:

- Misclassification of ddos_udp_lag: This traffic type is divided into two distinct groupings. The upper portion of ddos_udp_lag is correctly identified but also shows significant overlap with ddos_udp. This suggests that the characteristics of these two traffic types are similar enough to confuse the clustering algorithm. The other clustering of ddos_udp_lag is completely misclassified, being identified as a combination of ddos_netbios and ddos_snmp. This observation indicates a more serious problem of distinguishing between features.
- Misclassification of ddos_ntp: In this case, we note that at several specific coordinates, ddos_ntp traffic is misclassified. At coordinate point (100,50), the traffic is classified as

benign, while at coordinate point (40,10) it is identified as ddos_tftp. These misclassifications could result from subtle variations in the traffic patterns of ddos_ntp that make it similar to benign traffic or other types of DDoS attacks such as ddos_tftp.

These are just some of the evidences we wanted to highlight that confirm the data analyzed earlier.

ECDF of number of cluster assigned to each label



The graph of the empirical cumulative distribution function (ECDF) shows how the number of clusters assigned varies among different true labels (True Labels) in the dataset.
The information we can glean is that the median is between 6 and 8 clusters per label, with the total range ranging from 4 to 12. About 40 percent of the labels have 6 or fewer clusters assigned, which may help to assess the effectiveness of the clustering process.

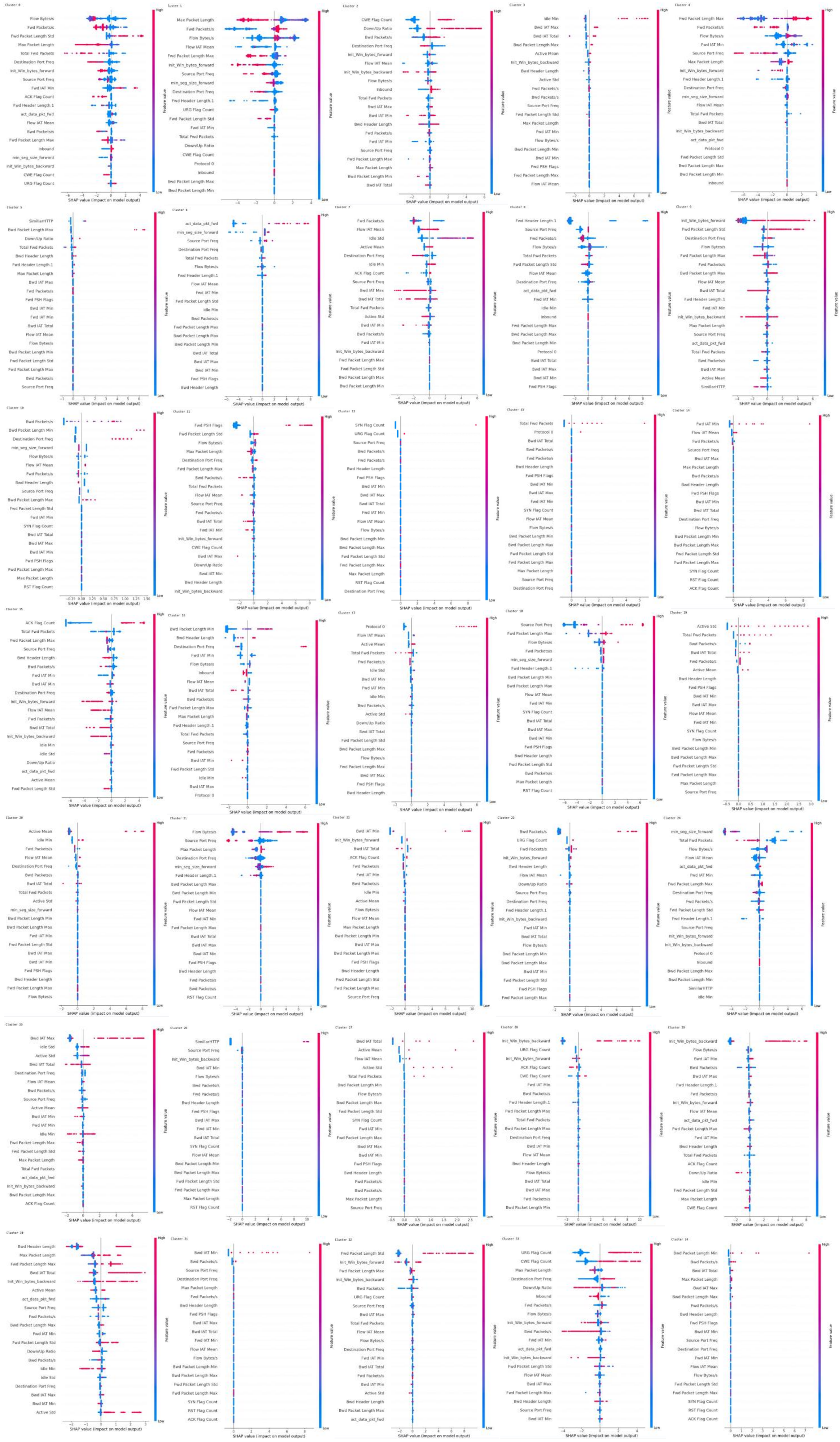
Feature importance

In the context of clustering, assessing the importance of features is critical to understanding how various attributes influence cluster formation. Two effective methods for analyzing this importance are multi-class classification and intra-cluster variable similarity, both of which are useful for refining and interpreting clustering results.

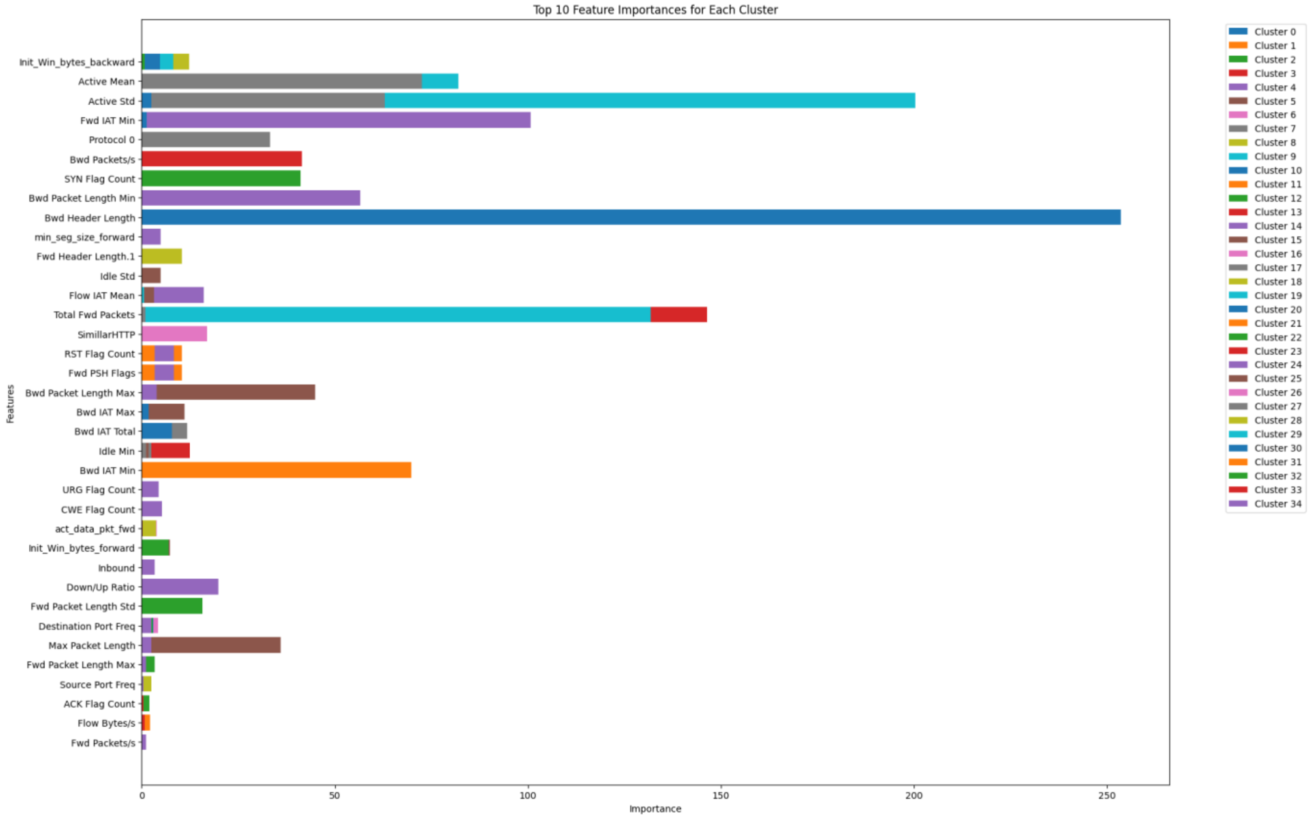
1. Multi-class classification treats objects in a given cluster as belonging to one class, while those in other clusters as members of a second class. Using classification algorithms, such as XGBoost (more faster than other) combined with interpretation techniques such as SHAP it is possible to identify the contribution of each individual feature to the classification decision. This approach not only clarifies which attributes distinguish one cluster from others, but also provides a quantitative basis for assessing the importance of each feature. This process is iterated for each cluster, allowing detailed and specific analysis.
2. Intra-cluster variable similarity method focuses on how much each variable contributes to the internal cohesion of the cluster. By averaging the similarities between each feature and its cluster centroid for each variable, a direct measure of the impact of each feature on cluster compactness is obtained. The use of the k-means WCSS_min technique, which measures the sum of the inner squares of the cluster for each feature, makes it possible to highlight those variables that contribute most to minimizing internal variance, suggesting greater significance in aggregating the data.

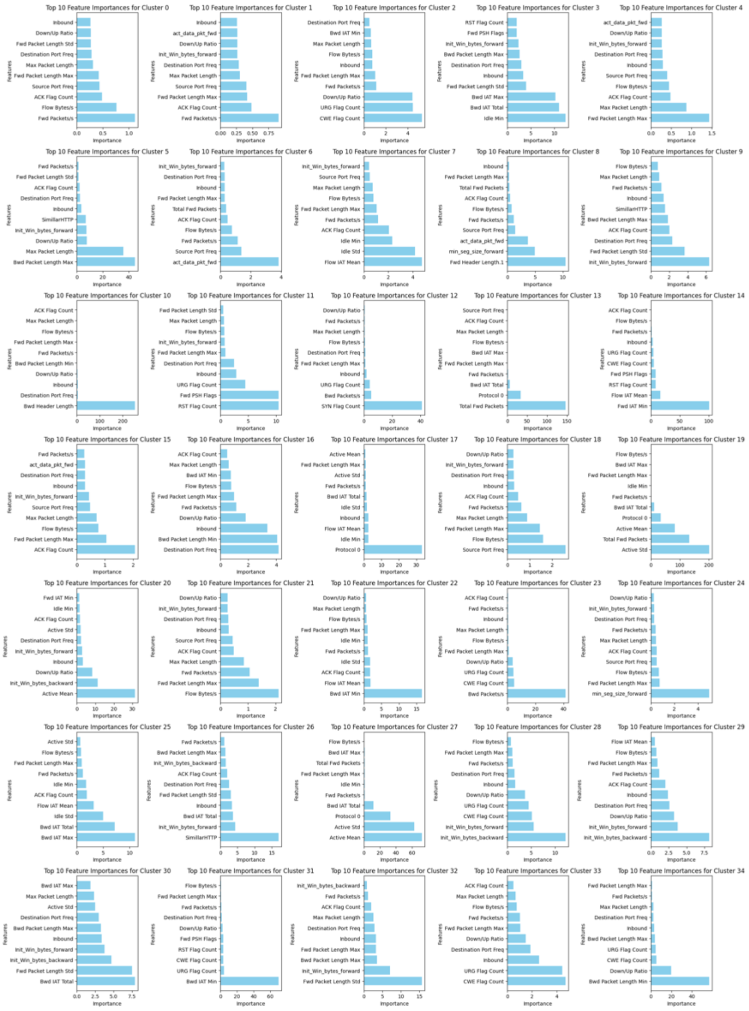
The combined implementation of these methods provides a thorough and balanced understanding of the importance of features in clustering.

Multi-class classification



Intra-cluster variable similarity





Before proceeding with the analysis, it is interesting to note how these two methods return very similar results. In fact, one only has to look at a few clusters to see that the most important features selected by the models are the same. Now we can combine all the previously collected data with the knowledge of the weights that the features have in the individual clusters in order to make the similarities and misclassifications in our k-means clustering algorithm meaningful.

Important note: in these visualizations the noise clusters will be present because they could not be removed; when reading the graphs it is good to ingor them.

Detection of Sub-Attack and Similarities

The next goal is to identify any sub-attacks within the clusters, analyzing the distinguishing characteristics, and explore the formation of new similar groups or clusters based on these characteristics. In addition, a similarity analysis will be conducted among the attacks, identifying common features, in order to better understand the nature of the attacks and detect recurring patterns in malicious traffic.

Noise (less than 10 samples per cluster):

Cluster	10	13	14	19	27	31	34

Pure (or nearly pure) clusters:

Cluster	3	5	6	7	8	11	12
Label	benign	benign	ddos_dns	ddos_syn	ddos_dns	benign	benign

Cluster	16	17	20	25	26	28	30	32
Label	benign	benign	benign	ddos_syn	benign	benign	benign	benign

To be classified:

Cluster	0	1	2	4	9	15	18	21	22	23	24	29	33

- *Cluster 0:* we have seen to be mostly composed of ddos_udp and ddos_udp_lag looking at the most important features we notice that predominating are Fwd Packets/s and Flow Bytes/s. If we look at the graph obtained with t-SNE we notice that the ddos_udp and ddos_udp_lag clusters overlap in a sparse manner, so in this case we can say the kmeans algorithm is not able to capture the subtleties such that these two labels can be distinguished well

- *Cluster 1:* this cluster is very peculiar because it contains so many samples within it and because it identifies as many as 4 labels. Nevertheless, from the feature analysis and also by viewing the t-SNE it is difficult to understand why the algorithm magnifies this cluster

- *Cluster 2,9,23,29,33:* these clusters are all characterized by the predominant presence of two labels, benign and ddos_ntp, we look at the most important features and notice a very interesting recurrence, in fact all clusters have as main features: CWE Flag Count, URG Flag Count, Down/Up Ratio, Fwd Packet Length Max, Init_Win_bytes_forward, Inbound. This is very important information because it makes us realize that there is a strong similarity between the two labels

- *Cluster 4:* this cluster mostly represents ddos_ssdp, but to a small extent also ddos_mssql. The relationship between the two is not seen in any other cluster excluding the possibility of strong similarities between the two

- *Cluster 15, 22:* these two clusters represent the ddos_syn and ddo_tftp labels. We also note here a similarity between the most important features although less pronounced than in benign and ddos_ntp

- *Cluster 18:* this cluster consists mainly of ddos_ldap although other labels appear in significantly smaller and negligible amounts

- *Cluster 21:* this cluster consists mainly of ddos_mssql although other labels appear in significantly smaller and negligible amounts

- *Cluster 24:* this cluster has within it with non-negligible amounts 5 labels, so we can say that it does not carry important information such as to identify strong similarities or under attacks.

Regarding sub-attacks, we can look at the contingency matrix vertically and notice that:

- ddos_syn appears with many samples in two main clusters, so the fact that vare 1 label strongly represented by two clusters makes us think that there are two sub-attacks
- same reasoning applies to ddos_udp and ddos_udp_lag, only that the fact that cluster 1 is so large makes us more insecure