CICIDS2017 dataset: performance improvements and validation as a robust intrusion detection system testbed

Akram Boukhamla*

Faculty of New Technologies of Information and Communication,
Kasdi Merbah University,
Ouargla BP.511,30000, Algeria
Email: akr_lmd@hotmail.com
*Corresponding author

Javier Coronel Gaviro

Signal Processing Applications Group of Signals, Systems
and Radiocommunications,
Department ETSI de Telecomunicación,
Universidad Politécnica de Madrid,
Avda. Complutense, 30. 28040 Madrid, Spain
Email: javier.coronel@alumnos.upm.es

Abstract: Nowadays, network security represents a huge challenge on the fight against new sophisticated attacks. Many intrusion detection systems (IDS) have been developed and improved to prevent not allowed access from malicious intruders. Developing and evaluating accurate IDS involve the use of varied datasets that collect most relevant features and real data from up-to-date types of attacks to real hardware and software scenarios. This paper describes and optimizes a new dataset available called CICIDS2017 (CICIDS2017, 2017). Using principal component analysis (PCA) for the optimization process of the CICIDS2017 dataset, the dimensionality of the features and records have been reduced without losing specificity and sensitivity, thus, reducing the overall size and leading to faster IDS. Finally, the optimized CICIDS2017 dataset is evaluated using three well known classifiers (KNN, C4.5 and naïve Bayes). The results obtained show that the optimized dataset maintain the same specificity and sensitivity of the non optimized version.

Keywords: intrusion detection system; IDS; network security; network attacks; CICIDS2017; principal component analysis; PCA; machine learning.

Reference to this paper should be made as follows: Boukhamla, A. and Gaviro, J.C. (xxxx) 'CICIDS2017 dataset: performance improvements and validation as a robust intrusion detection system testbed', *Int. J. Information and Computer Security*, Vol. X, No. Y, pp.xxx–xxx.

Biographical notes: Akram Boukhamla is an Associate Professor at the Department of Computer Science, University of Kasdi Merbah Ouargla/Algeria. He worked as reviewer for many academic journals. His current research focuses on cyber security, big data security analysis, internet traffic analysis and the detection of malware and attacks.

Javier Coronel Gaviro is Computer Science Engineer, Medical Physics MSc, PhD candidate and CEO of Expo Impex, S.L. with 20+ year of working experience in computer forensics. His current research focuses on medical imaging and computer forensics.

1 Introduction

Today, internet and computer networks are exposed to an increasing number of security threats where intrusions are a well-known type of them. These attacks have been growing in number and sophistication over the last few years. Malware, botnets, spam, phishing, and denial of service attacks have become continuous threats for today's networks and hosts (Feily et al., 2009).

Due to this, intrusion detection system (IDS) have been developed and have attracted the attention of many researchers in order to propose more efficient solutions against these intrusion's threats. Existing IDS methods and techniques are commonly categorised as either anomaly-based (Jyothsna et al., 2011), signature-based (Holm, 2014) or a combination of both.

Signature-based intrusion detection systems (SIDS) analyse incoming information, compare it using defined rules of attack signatures and then decide whether it is a positive or a negative alert. This type of systems have the advantage to be much more accurate at identifying an intrusion attempt, resulting in a low false positive rate (type I error) as long as attack patterns are clearly defined in advance (Kumar, 2012); but, as SIDS only can detect intrusions if the attack's signature matches a pattern that is in the database, they cannot detect new attacks (also called zero-day-attacks) since the attack signature does not exist yet in the system database.

The second category of IDSs is the anomaly-based intrusion detection systems (AIDS), also called behavior based detection systems that are based on analysing the network behavior (Jyothsna et al., 2011). AIDS are continuously monitoring network traffic in order to find abnormal behavior. They also can be trained to detect new treats or even implement self-learning algorithms. Anomaly detection algorithms have the advantage that they can detect new types of intrusions as deviations from normal usage (Lazarevic et al., 2003). AIDS' sensitivity must be fine-tuned to get high specificity and avoid a high rate of type I errors (false positives) that could burden the system (Yang et al., 2006). If sensitivity is set too low, the rate of type II errors (false negatives) could compromise the network.

Developing an accurate AIDS depends on having a rich updated and well-formed dataset that contains various criteria and features. In the literature there are numerous datasets available on the internet, but only a few fulfil all expectations. KDD-99 (Thomas et al., 2008), CAIDA (Udhayan and Hamsapriya. 2011), ISCX2012 (Shiravi et al., 2012), Kyoto (Song et al., 2006), are the most well-known datasets that are conceived to resemble the real world data traffic, however, the continuous evolution of network's threats and attack types, forms a challenge to existing datasets, since these datasets should cover most of the current trends of attacks. Therefore, many of the existing datasets suffer from being outdated after a few years of their appearance, as the case of KDDCUP 99.

The current work gives a description of a new IDS dataset CICIDS2017 that includes most of the up-to-date attacks along with labelled flows, covering more than 80 features which are created to overcome some issues of existing datasets. Also, by applying preprocessing techniques (dataset cleaning, removing related network flow features, and removing features with low variance) and principal component analysis (PCA), we considerably reduce the dimensionality of the dataset since the number of features decreases approximately 75%, of the total features number. Finally, to evaluate the performance of the optimised dataset, three well-known machine learning algorithms have been applied (KNN, C4.5 and naïve Bayes).

This paper is believed to be the first to describe and optimise CICIDS2017 dataset since its appearance.

The rest of the paper is organised as follows. Section 2 presents an overview of recent network types of attacks. Section 3 gives a detailed description of CICIDS2017 dataset. In Section 4 it is shown the proposed CICIDS2017 dataset optimisation and the techniques applied. In Section 5, the optimised dataset is evaluated and the results are discussed. Finally, conclusions and future works are presented in Section 6.

2 Overview of the most recent network attack types

2.1 DDoS

Distributed denial of service DDoS) is an attack type which tries to get a target unavailable by overwhelming it with network traffic from multiple sources. Another description of DDoS, given by WWW Security FAQ (Stein and Stewart, 2003), states that is an attack designed to render a computer or network incapable of providing normal services. Specht and Lee (2004) have proposed taxonomies to characterise the scope of DDoS attacks, they have considered two main classes of DDoS: bandwidth depletion and resource depletion attacks. The bandwidth depletion is performed to deny legitimate traffic from achieving target (victim) by flooding victim network with unwanted traffic. While resource depletion attacks serve to tie up the resources of victim's system, making the victim unable to respond to clients' requests

2.2 Port scan attacks

Port scan attack is the first step in remote-to-local (R2L) attacks (Kim and Lee, 2008). This type of attack attempts to know which ports are open by scanning a range of hosts' ports. The process of scanning itself is typically harmless, but could yield to a network attack, since the attacker could find potential vulnerabilities inside the victim host and therefore gain access to its services. Port scan attacks can be classified into open scan, half-open scan and stealth scan. Each of the cited categories aims to identify the open ports of a target host.

There are two scenarios that can occur while port scanning, either port is open: the attacker reveals that by 3-way TCP/IP handshake process termination; or port is closed: where the registered network traffic is as follows: client → SYN (synchronise), server → RST/ACK (reset/acknowledge), client → RST (reset).

2.3 Botnet

The term Botnet is composed from two words ‘robot’ and ‘network’. It is a software piece that can infect to computers, servers, smartphones, and potentially to any internet-connected device, in order to get user’s information or serve as a remote control for an attacker who can start other kinds of distributed attacks (i.e. DDoS).

2.4 Web attacks

Most well-known web attacks are brute force, XSS and SQL injection cited below:

2.4.1 Brute force

According to IBM (2017), brute force is an attack that uses a repetitive method of trial and error to guess username, password, credit card number, or cryptographic key of victim. An attacker could launch a brute force attack by trying to guess the user ID and password for a valid user account on the web application. If the brute force attempt is successful, the attacker might be able to access confidential information.

2.4.2 Cross-site scripting

Cross-site scripting attacks (sometimes referred to as XSS) are attacks on websites that dynamically display user content without checking and encoding information entered by users. Cross-site scripting attacks thus consist in forcing a website to display HTML code or scripts entered by users. The code thus included (the term ‘injected’ is usually used) in a vulnerable website is said to be ‘malicious’.

2.4.3 SQL injection

This type of attacks has, specially, web-servers as target, since it is a common language used in database servers (SQL). To get access to the information contained in the database server, attackers insert customised queries to obtain critical information such as personal information, passwords, or credit card numbers for further malicious purposes. Also, such as the case for web-servers that contains a content management system (CMS), it is possible to insert arbitrary code as a database register and execute it, thanks to any vulnerability of the CMS. Usually, most of the SQL injection attacks infect vulnerable server, it may be possible for an attacker to go to a website's search box and type in code that would force the site's SQL server to dump all of its stored usernames and passwords for the site.

2.5 Heartbleed

Heartbleed vulnerability was intentionally introduced by a developer in 2011 and it has been officially revealed by OpenSSL in 2014. This vulnerability allowed attackers to remotely dump protected memory – including data passed over secure channels and private cryptographic keys – from both clients and servers (Durumeric et al., 2014).

3 Cicids2017 dataset

The CICIDS2017 dataset has been created to overcome the existing gaps in the current datasets. A common issue is the outdated data that most of the datasets suffer from because intrusion attack types are evolving continuously and become more sophisticated. Other problems found in some datasets are the lack of features and metadata, some of them do not contain enough variety of known attacks.

In Durumeric et al. (2014), relevant criteria for building an accurate dataset are described, such as: Complete Network configuration, complete traffic, labelled dataset, complete interaction, complete capture, available protocols, attack diversity, heterogeneity, feature set and meta-data. An evaluation framework containing cited criteria was built, and a comparison between different IDS datasets was performed. As a conclusion, none of the evaluated datasets fulfil at the same time the eleven criteria. Hence, the idea of building an IDS dataset that covers all the above criteria.

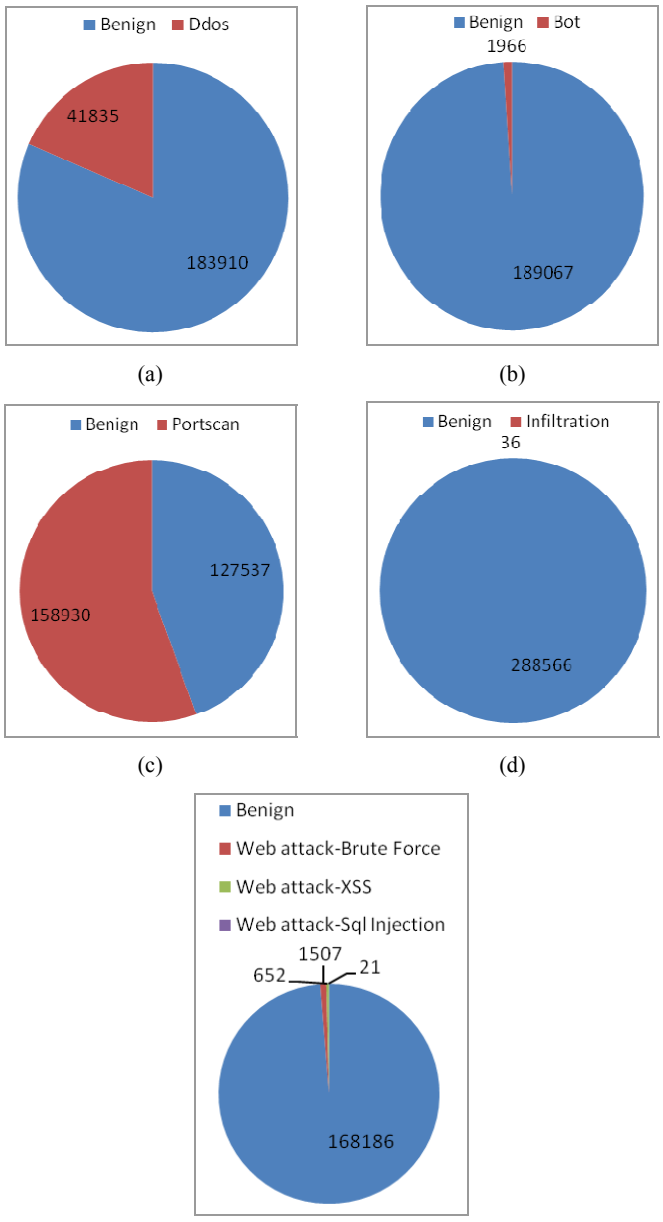
CICIDS2017 dataset meets all criteria established in Gharib et al. (2016), classifying it as the most complete dataset to date as said by (Sharafaldin et al., 2018).

Table 1 CICIDS2017 feature names

Features name	*Type*
Source Port, Destination Port, Protocol, Flow Duration, Total Fwd Packets, Total Backward Packets,Total Length of Fwd Packets, Total Length of Bwd Packets, Fwd Packet Length Max, Fwd Packet Length Min, Fwd Packet Length Mean, Fwd Packet Length Std,Bwd Packet Length Max, Bwd Packet Length Min, Bwd Packet Length Mean, Bwd Packet Length Std,Flow Bytes/s, Flow Packets/s, Flow IAT Mean, Flow IAT Std, Flow IAT Max, Flow IAT Min,Fwd IAT Total, Fwd IAT Mean, Fwd IAT Std, Fwd IAT Max, Fwd IAT Min,Bwd IAT Total, Bwd IAT Mean, Bwd IAT Std, Bwd IAT Max, Bwd IAT Min,Fwd PSH Flags, Bwd PSH Flags, Fwd URG Flags, Bwd URG Flags, Fwd Header Length1, Bwd Header Length,Fwd Packets/s, Bwd Packets/s, Min Packet Length, Max Packet Length, Packet Length Mean, Packet Length Std, Packet Length Variance,FIN Flag Count, SYN Flag Count, RST Flag Count, PSH Flag Count, ACK Flag Count, URG Flag Count, CWE Flag Count, ECE Flag Count, Down/Up Ratio, Average Packet Size, Avg Fwd Segment Size, Avg Bwd Segment Size, Fwd Header Length,Fwd Avg Bytes/Bulk, Fwd Avg Packets/Bulk, Fwd Avg Bulk Rate, Bwd Avg Bytes/Bulk, Bwd Avg Packets/Bulk,Bwd Avg Bulk Rate,Subflow Fwd Packets, Subflow Fwd Bytes, Subflow Bwd Packets, Subflow Bwd Bytes,Init_Win_bytes_forward, Init_Win_bytes_backward, act_data_pkt_fwd, min_seg_size_forward,Active Mean, Active Std, Active Max, Active Min,Idle Mean, Idle Std, Idle Max, Idle Min	Numerical
Flow ID, Source IP, Destination IP, Timestamp, External IP	Nominal

CICIDS2017 dataset contains the latest attack types such as: DoS, DDoS, brute force SSH, brute force FTP, heartbleed, infiltration and botnet which make it the most up-to-date, compared to other datasets.

Figure 1 CICIDS2017 features distribution, (a) DDoS attacks, (b) botnet attacks (c) port-scan attacks (d) infiltration attacks (e) web attack (see online version for colours)



Unlike other IDS datasets that separate training from testing data, CICIDS2017 gathered all labelled records of each specified type attack into an unique CSV file format. Each

CSV file is composed of a given number of labelled records, and 85 features that describe these records.

Table 1 illustrates 85 features and their data type (numerical or nominal). We notice that there are only five nominal data type features and the rest are numerical data type.

In this experiment, focus is set on five attack types distributed through five CSV files cited below:

- *Friday-WorkingHours-Afternoon-DDos.pcap_ISCX*

  this dataset contains 225,745 labelled records classified as 183,910 benign records and 41,835 DDoS which represents 18.53% of the entire data

- *Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX*

  in this dataset we can notice that the number of port-scan attack records is higher that benign records with 158,930 records (55.5%) against 127,537 records for benign records

- *Friday-WorkingHours-Morning.pcap_ISCX*

  The total records number of this dataset is 191,033 labelled records with 189,067 benign and 1,966 botnet attacks with percentage of 1% of the whole data

- *Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX*

  this dataset contains only 39 infiltration attack records from total of 288,566 records, representing a low percentage 0.01% of the whole dataset

- *Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX*

  Web attack dataset contains three types of attacks: Brute force 1,507 records (0.88%), XSS 652 records (0.38%) and Sql injection 21 records (0.01%) where 168,186 records were benign.

4 Dataset optimisation

4.1 Pre-processing step

Most of the available datasets contain unwanted elements (missing, redundant or infinite values) that should be removed or transformed. The step of preprocessing is essential to obtain a suitable dataset.

4.1.1 Dataset cleaning

Before start using CICIDS2017 for IDS model evaluation, it has been necessary to clean up the dataset from errors which could occur while flow data are been acquiring. We first removed a redundant attribute 'Fwd_Header_Length' that appeared twice in the list of attributes. Furthermore, redundant records have been also dropped from the whole dataset, 109 in DdoS dataset, 266 in port-scan dataset, 599 in botnet dataset, 550 in web attack dataset and 635 in infiltration attack dataset. All missing values have been replaced by zeros and infinite values have been replaced by the mean of their attribute value.

4.1.2 Removing related network flow features

CICIDS2017 contains features that have been recorded while acquiring data flow, those features are related to a specific network and don't have any impact on model results.

The second step of the dataset pre-processing consists of removing all those meaningless features manually in order to decrease the data dimension. Among useless features we find five nominal ones: Flow ID, source IP, destination IP, timestamp …, etc. By removing them, nominal features processing disappear since some classification models require numerical values rather than nominal ones.

4.1.3 Removing features with low variance

Variance σ^2 is the average of the squared differences from the mean. Here is the formula which calculates the variance of a given feature:

$$\sigma^2 = \frac{\sum_{i=1}^{n} (x_i - \mu)^2}{n} \tag{1}$$

where n is the number of terms in the distribution. μ is the mean, x_i is the given term in the distribution.

Or we can simply define the variance as a measure of how far each value in the dataset is from the mean.

In our experiment, we used variance criteria to remove all features with variance value equal to zero, since removing them increases our model accuracy. Also, those features are irrelevant in our data and can decrease the performance of the model analysis. By applying the variance removing criteria, ten features have been eliminated from datasets which are: Bwd PSH Flags, Fwd URG Flags, Bwd URG Flags, CWE Flag Count, Avg Bytes/Bulk, Fwd Avg Packets/Bulk, Fwd Avg Bulk Rate, Bwd Avg Bytes/Bulk, Bwd Avg Packets/Bulk and Bwd Avg Bulk Rate.

4.2 Principal component analysis

PCA is a technique that extracts the dominant patterns of a matrix in terms of a complementary set of score and loading plots (Abdi and Williams, 2018) to represent it as a set of new orthogonal variables called principal components (Wold et al., 1987).

4.2.1 Covariance matrix

The classic approach to PCA is to perform an eigen-decomposition on the covariance matrix Σ, which is a d×d matrix where each element represents the covariance between two features. The covariance between two features is calculated as follows:

$$\sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^{N} (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k) \tag{2}$$

We can summarise the calculation of the covariance matrix via the following matrix equation:

$$\sum = \frac{1}{(n-1)}\left((X-\bar{X})^T (X-\bar{X})\right) \tag{3}$$

where $\bar{X}$ is the mean vector $\bar{X} = \sum_{k=1}^{n} x_i$.

The mean vector is a d-dimensional vector where each value represents the sample mean of a feature column in the dataset.

Table 2 Final PCA-dataset

Datasets	*Instances nbr*	*Features nbr*
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX	benign: 183,801 ddos: 41,835	21
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX	benign: 127,271 portscan: 158,930	23
Friday-WorkingHours-Morning.pcap_ISCX	benign: 188,468 bot: 1,966	23
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX	benign: 167,636 brute force: 1,507 xss: 652 sql injection: 21	22
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX	benign: 287,931 infiltration: 36	26

Table 3 CICIDS2017 vs. PCA-dataset

	CICIDS2017		*PCA-CICIDS2017*	
	records	*features nbr*	*records*	*features nbr*
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX	225,745	85	225,636	21
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX	286,467	85	286,201	23
Friday-WorkingHours-Morning.pcap_ISCX	191,033	85	190,434	23
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX	288,566	85	287,967	26
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX	170,366	85	169,816	22

Results in Table 2 show the final reduced datasets after applying the cited reduction and transformation techniques. Notice that the number of features has been significantly reduced after applying PCA method with approximately 75% for each dataset. Also, the number of records has decreased for each dataset compared to the original ones. The rest of the experiment considers the data given in Table 2 for the evaluation classifiers. Table 3 contains a comparison between the CICIDS2017 and the optimised PCA-dataset.

4.3 Performance assessment with cross-validation

To avoid biasing in the classification accuracy, class predictions do not consider data used for training. A cross-validation technique is applied, consisting on dividing the original sample into k samples, selecting one of the k samples as a validation set and considering the other $(k - 1)$ samples as the training set. A performance score is calculated. Then the operation is repeated by selecting each of the validation samples from the rest of the samples (the other $k - 1$) that have not yet been used for model validation, obtaining another performance score. The mean of the k mean squared errors is finally calculated to estimate the prediction error. In our experiment we proceed with $k = 10$ folds.

4.4 PCA-dataset evaluation

The accuracy of our PCA-datasets has been evaluated by applying three well known classification algorithms that are: K-nearest neighbour (KNN), naïve Bayes and decision tree (C4.5). Each algorithm has been applied to every dataset studied. Obtained results of each classification model have been extracted to calculate detection rate and false alarm rate. Efficiency results of each model on given datasets are shown in Table 4.

- *Detection rate:* represents attacks detected by the system divided by the number of attacks in the data set.

$$detection\ rate = \frac{TP}{TP + FN}$$

- *False alarm rate:* represents records misclassified as attacks divided by the benign records in dataset.

$$false\ alarm\ rate = \frac{FP}{FP + TN}$$

where *TP* is true positive, *FP* is false positive, *FN* is false negative, and *TN* is true negative.

Table 4 Detection and false alarm rate of PCA-dataset attacks

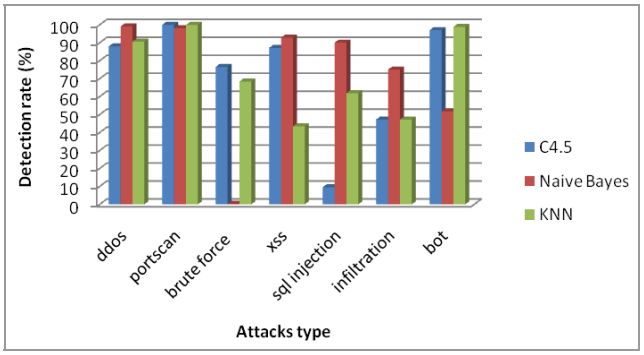
Classifiers / *Attacks type*		*KNN*		*C4.5*		*Naïve Bayes*	
		Detection rate (%)	*False alarm rate (%)*	*Detection rate (%)*	*False alarm rate (%)*	*Detection rate (%)*	*False alarm rate (%)*
DDoS		90.6	1.97	87.9	2.5	99	59
Port-Scan		99.9	8×10^{-5}	99.9	0.02	98	2.9
Botnet		98.8	0.01	97	0.03	51.8	1.27
Web attacks	Brute force	68.4	0.3	76.5	0.1	0.3	0.4
	XSS	43.4	0.2	87.1	0.2	92.8	1.2
	SqlInjection	61.9	0.005	9.52	5×10^{-5}	90.4	4.7
Infiltration		47.2	0.003	47.2	0.002	75	2.2

5 Results and discussion

For this experiment CICIDS2017 has been used because it contains most recent network attacks grouped in CSV files. After preprocessing it, in order to reduce data size, it had been transformed to a PCA-dataset and had been measured each attack type dataset performances.

Table 4 shows detection rate and false alarm rate for each attack type, measured by three classifiers: KNN, C4.5 and naïve Bayes. By interpreting the results obtained, shown in figure 2, we notice a high detection rate for ddos attack with naïve Bayes and KNN classifiers (90.6% and 99% respectively), therefore, naïve Bayes has a high false alarm rate (59%) which classify KNN (with 1.9% of false alarm rate) as an adequate classifier for ddos attack.

Figure 2 Detection rate of PCA-dataset attacks type (see online version for colours)



For port-scan attack, KNN and C4.5 are suitable classifiers since they have the highest detection rate (99.9%) with a low false alarm rate ($5x10^{-5}$% and 0.02% respectively). KNN classifier has given excellent results for botnet attacks since detection rate has reached 98.8% with false alarm rate of 0.01%. Web attacks have been grouped under three types: brute force, with best detection rate given by C4.5 classifier equal to 76.5% and false alarm rate equal to 0.1%, XSS with highest detection rate equal to 92.8% and 87.1% (naïve Bayes and C4.5 respectively) with a slight advantage to C4.5 because it has the lowest false alarm rate (0.2% against 1.2% for naïve Bayes) and the last web attacks sql injection with high detection rate recorded equal to 90.4%, therefore, it has high false alarm rate (4.7%). The last attacks type, infiltration, has 75% detection rate recorded by naïve Bayes with 2.2% false alarm rate.

6 Conclusions and future work

This experiment was based on the new CICIDS2017 dataset which came to overcome some issues of existing intrusion datasets such as anonymity and outdated attack types. First, the dataset was cleaned up from unwanted elements found due to dataset acquiring techniques; also we reduced dataset dimensionality, since it contained a high number of features (85 features), by accurate selecting the features that were more representative, by applying a PCA procedure. The number of features had considerably been reduced, by

approximately 75%, of the total features number. Finally, the transformed PCA-datasets were evaluated by three well-known classifiers in order to measure the true detection and false alarm rates, giving promising results. The transformed data showed also to be a robust representation of the original dataset, getting a high positive detection rate and low false alarm rate as denoted by three well-known classifiers.

From observed results we conclude that classifiers gave all together better detection rate and lower false alarm when attack records number was relatively high comparing with total records number, the case of DDoS and port-scan attacks. Also, it has been found that some CSV files contained few number attack samples compared with the whole records (infiltration, sql injection, brute force …) which affected learning step of classifiers, as a result, poor results were obtained for these models. It was noticed also that some classifiers gave a high detection rate with also a high false alarm rate and vice versa. We are considering to develop a hybrid classifier able to obtain a high detection rate and a low false alarm rate in order to build an accurate model.

References

Abdi, H. and Williams, L.J. (2018) 'Principal component analysis', *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 2, No. 4, pp.433–459.

CICIDS2017 (2017) *Intrusion Detection Evaluation Dataset* [online] http://www.unb.ca/cic/datasets/ids- 2017.html (accessed 28 December 2017).

Durumeric, Z. et al. (2014) 'The matter of heartbleed', *Proceedings of the 2014 Conference on Internet Measurement Conference*, ACM.

Feily, M., Shahrestani, A. and Ramadass, S. (2009) 'A survey of botnet and botnet detection', *SECURWARE'09. Third International Conference on Emerging Security Information, Systems and Technologies*, IEEE.

Gharib, A. et al. (2016) 'An evaluation framework for intrusion detection dataset', *International Conference on Information Science and Security (ICISS)*, IEEE.

Holm, H. (2014) 'Signature based intrusion detection for zero-day attacks: (not) a closed chapter?', *47th Hawaii International Conference on System Sciences (HICSS)*, IEEE.

IBM (2017) *IBM Security Network Intrusion Prevention System 4.6.0* [online] https://www.ibm.com/support/knowledgecenter/en/SSB2MG_4.6.0/com.ibm.ips.doc/concepts/wap_brute_force.htm (accessed 4 January 2018).

Jyothsna, V., Prasad, V.V.R. and Prasad, K.M. (2011) 'A review of anomaly based intrusion detection systems', *International Journal of Computer Applications*, Vol. 28, No. 7, pp.26–35.

Kim, J. and Lee, J-H. (2008) 'A slow port scan attack detection mechanism based on fuzzy logic and a stepwise policy', *4th International Conference on Intelligent Environments*, IET.

Kumar, V. (2012) 'Signature based intrusion detection system using SNORT', *International Journal of Computer Applications & Information Technology*, Vol. 1, No. 7, pp.35–41.

Lazarevic, A. et al. (2003) 'A comparative study of anomaly detection schemes in network intrusion detection', *Proceedings of the 2003 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics*.

Sharafaldin, I., Lashkari, A.H. and Ghorbani, A.A. (2018) 'Toward generating a new intrusion detection dataset and intrusion traffic characterization', *4th International Conference on Information Systems Security and Privacy (ICISSP)*, Portugal.

Shiravi, A., et al., (2012) 'Toward developing a systematic approach to generate benchmark datasets for intrusion detection', *Computers & Security*, Vol. 31, No. 3, pp.357–374.

Song, J., Takakura, H. and Okabe, Y. (2006) *Description of Kyoto University Benchmark Data* [online] http://www.takakura.com/Kyoto_data/BenchmarkData-Description-v5.pdf (accessed 15 March 2016).

Specht, S.M. and Lee, R.B. (2004) 'Distributed denial of service: taxonomies of attacks, tools, and countermeasures', *ISCA PDCS*.

Stein, L. and Stewart, J. (2003) *The World Wide Web Security FAQ* [online] https://www.w3.org/Security/Faq/wwwsf6.html (accessed 8 January 2018).

Thomas, C., Sharma, V. and Balakrishnan, N. (2008) 'Usefulness of DARPA dataset for intrusion detection system evaluation', *Proc. SPIE 6973, Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security*, 69730G, doi: 10.1117/12.777341 [online] https://doi.org/10.1117/12.777341.

Udhayan, J. and Hamsapriya, T. (2011) 'Statistical segregation method to minimize the false detections during DDoS attacks', *IJ Network Security*, Vol. 13, No. 3, pp.152–160.

Wold, S., Esbensen, K. and Geladi, P. (1987) 'Principal component analysis', *Chemometrics and Intelligent Laboratory Systems*, Vol. 2, Nos. 1–3, pp.37–52.

Yang, D., Usynin, A. and Hines, J.W. (2006) 'Anomaly-based intrusion detection for SCADA systems', *5th Intl. Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies (NPIC&HMIT 05)*.