EAI Endorsed Transactions
on Industrial Networks and Intelligent Systems
Research Article EAI.EU

A data-driven approach for Network Intrusion Detection and Monitoring based on Kernel Null Space

Truong Thu Huong[1,*], Ta Phuong Bac[1], Quoc Thong Nguyen[2], Huu Du Nguyen[3], Kim Phuc Tran[4]

[1]School of Electronics and Telecommunications, Hanoi University of Science and Technology, 1 Dai Co Viet street, Hanoi, Vietnam.

[2]Division of Artificial Intelligence,Dong A University, Da Nang, Vietnam.

[3]Faculty of Information Technology, Vietnam National University of Agriculture, Hanoi, Vietnam.

[4]GEMTEX Laboratory, Ecole Nationale Sup des Arts et Industries Textiles, BP 30329 59056 Roubaix Cedex 1, France.

Abstract

In this study, we propose a new approach to determine intrusions of network in real-time based on statistical process control technique and kernel null space method. The training samples in a class are mapped to a single point using the Kernel Null Foley-Sammon Transform. The Novelty Score are computed from testing samples in order to determine the threshold for the real-time detection of anomaly. The efficiency of the proposed method is illustrated over the KDD99 data set. The experimental results show that our new method outperforms the OCSVM and the original Kernel Null Space method by 1.53% and 3.86% respectively in terms of accuracy.

Received on 09 June 2019; accepted on 25 July 2019; published on 07 August 2019

Keywords: Network Security Support, Kernel Quantile Estimator, One-class Classification, Kernel Null Space vector machine.

doi:10.4108/eai.13-6-2019.159801

1. Introduction

Security policies are very important in computer systems to prevent the outside attacks. However, it can be said in general that the existing security policies are not strong enough to guarantee this function as more and more new types of attacks appear beyond the capabilities of these security systems. It is therefore necessary to build a monitoring system for the computer systems to early detect novelties. The early detection of abnormality can help the computer systems to reduce the damage and protect the crucial information. Among the available methods, Intrusion detection system (IDS) is a powerful tool and it attracts the attention of researchers [3]. The IDS has been used in a great number of applications such as network intrusion, fraud detection and security systems.

Currently, there are two families of mechanisms in IDS: signature-based IDS and anomaly-based IDS. In this paper, we focus on developing an anomaly-based IDS solution, in which the designed IDS system is trained based on knowledge of normal traffic only. Such a system does not need to be trained with attack data traces to later detect if incoming traffic is anomaly or normal. This characteristic is good for the attack detection aspect since attack manners may vary over time. Due to continuous variation of attacks, the system might not be trained with a new attack pattern before, and as the result, may not be effective any more.

Among the anomaly-based IDS solution family, Novelty Detection is a research direction attracting researchers who have been working in the the attack detection field. Novelty detection is the identification of unknown data that an IDS system is not aware of during training. Its goal is to identify abnormal behaviors which are not consistent with the normal state of a

*Corresponding author. Email: huong.truongthu@hust.edu.vn

system. A model is built from normal data to detect unknown abnormality by novelty detection algorithms such as OCSVM [9, 17] and Kernel Null Space [1, 2, 7]. There is also an approach in intrusion detection using Statistical Process Control [10].

Our proposed solution aims at improving the performance of the Kernel Null Space method [2] in terms of accuracy. To be more specific, we propose using a Control-Chart based method called Kernel Quantile Estimator to determine the detection threshold dynamically driven by each specific training data set instead of using a fixed threshold as described in the existing Kernel Null Space solutions [1, 2, 7]. The Control Chart Based on a Kernel Estimator of the Quantile Function was also developed in [8]. In addition, we also optimize the kernel parameter of the kernel function to improve the performance of novelty detection.

The rest of the paper is organized as follows: Section 2 elaborates the related work. Our proposed Enhanced Kernel Null Space solution - EKNS for Novelty Detection is provided in Section 3, followed by the performance evaluation in Section 4. Finally, conclusion is given in Section 5.

2. Related work

Recently, a variety of defense mechanisms have been proposed to combat transport-level DDoS flooding (distributed denial of service) in the state of the art.

Ingress/Egress filtering mechanisms [4] is a source-based solution that detects and filter packets with spoofed IP based on the valid IP address range internal to the network. And this solution is a signature-based approach as well. However, the spoofed packets can not be detected if their addresses are still in the valid internal IP address range. Another source-based and signature-based scheme D-WARD [6] monitors inbound and outbound traffic of a source network and comparing the network traffic with predefined normal flow models. This solution can be bypassed by attackers who can control traffic in a normal range. In addition, another signature-based and source-based approach (MULTOPS) [5] makes use of a significant difference between the traffic rates going out and coming from a host to define if the network is either the source or the destination of an attack. However the assumption that incoming and outgoing traffic rates are proportional is not always the case.

General speaking, source-based solutions are not totally effective against DDoS flooding attacks since, attack sources can be distributed; and it is not easy to differentiate legitimate from attack traffic near the sources, since traffic volume of the traffic may not be big enough.

Besides, the network-based mechanisms are deployed to detect an attack and stop it at intermediate networks. Some of the main schemes to handle DDoS attacks can be listed as follows: AVANT-GUARD [12] builds a module at the gateway switch to mitigate saturation attacks by checking the TCP 3 hand-shake process. The authors in [18] proposed another way to handle the TCP 3-hand shaking to detect attack flooding as well. But the two solutions are designed specifically for TCP SYN flood.

An anomaly-based scheme can be found in [11] that uses Fuzzy Interference System to detect anomalies based on traffic pattern. This solution can detect if an attack happens no matter what type of attack is it. However, this type of solution requires training data sets with both labels: Anomaly and Normal data sets, so in case the attack data set is not available, it is difficult to realize this scheme in reality, when attacks can vary in many different new ways.

In this paper, we propose a network-based and novelty-based mechanism that is based on traffic data set to detect novelty in traffic. Our proposed solution does not require advanced knowledge in attack pattern (i.e. available attack data sets) but only normal traffic behavior in order to detect anomalies for incoming traffic.

Generally, the novelty detection issues can be divided into two types based on the number of known classes during the training phase: one-class and multi-classes. Since our work focuses on one-class classification, we will review the state of the art for the family of one-class novelty detection. To the best of our knowledge, Kernel Null Space has the highest performance in novelty detection and there are only three studies dealing with one-class classification in novelty detection using this method [1, 2, 7]. The authors [2] proposed Kernel Null Space for novelty detection but they made the experiment with a fixed threshold and a fixed kernel parameter of the kernel function. Paul et al [1] also improved the performance of the original method. However, they only concentrated on decreasing the timing operating of the algorithm, the accuracy remains unchanged. Following this trend, Liu Juncheng et al [7] improved the solution proposed in [2] by decreasing the complexity of the kernel null space method without taking the accuracy into account.

From another approach, the OCSVM method, which detects novelty by finding the boundary of training data with maximum margin, is often used to solve the one-class novelty detection problem, for example, in [17, 19]. The OCSVM method has received more extensive attention since it can easily handle nonlinear data with kernel trick and also achieve a high level of detection accuracy [17].

In order to improve the accuracy of the Kernel Null Space method [2] in the favor of anomaly detection.

We propose a solution combining Kernel null space and Control chart to automatically define an efficient detection threshold stemming from each training data trace.

Moreover, we also use the optimizing parameter method proposed in [17] to increase the accuracy for the algorithm. Our proposed solution is proved to outperform the Kernel Null Space methods in [1, 2, 7] and OCSVM in [17, 19] in terms of Accuracy.

3. Intrusion Detection Scheme using the control-chart-based Kernel Null Space Method.

3.1. Intrusion detection system architecture

Endpoint security is a key part of an organizational response to cyber threats. As can been seen in Figure 1, beside perimeter firewalls and Identity Access Management tools can restrict and control access to the organizations network, people built an IDS/IPS system to detect and stop an intruder who has managed to breach our security. IPS and IDS systems look for intrusions and symptoms within traffic. IPS/IDS systems would monitor for unusual behavior, abnormal traffic, malicious coding and anything that would look like an intrusion by a hacker being attempted.

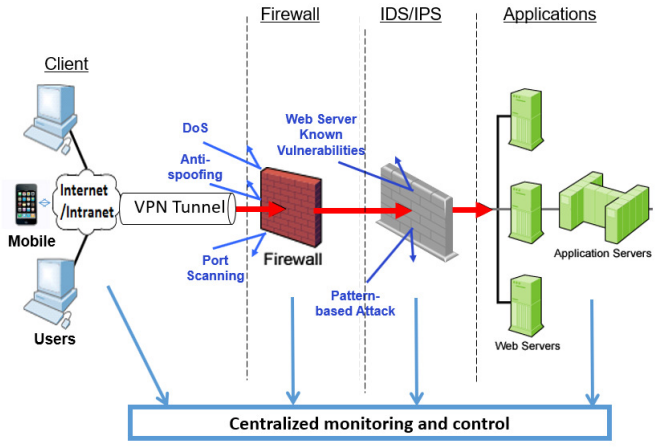

Figure 1. Networks Intrusion detection system

Our designed architecture of an IPS/IDS system that monitors and analyzes traffic in real time can be described in Figure 2. Incoming Internet traffic getting through Network devices is analyzed and extracted to different attributes which are indicators for anomaly detection. Data attributes in different formats and scales are then normalized, and finally trained with a specific training algorithm. Basically, to detect attack threats, an IPS/IDS system can use both a knowledge database of signatures that is a database of attacks happened in the past and online anomaly-based detection to detect if attacks have happened. Policy enforcer is the final step of an IPS/IDS system where we can set different policies onto the network devices to prevent or mitigate attacks from damaging our system.

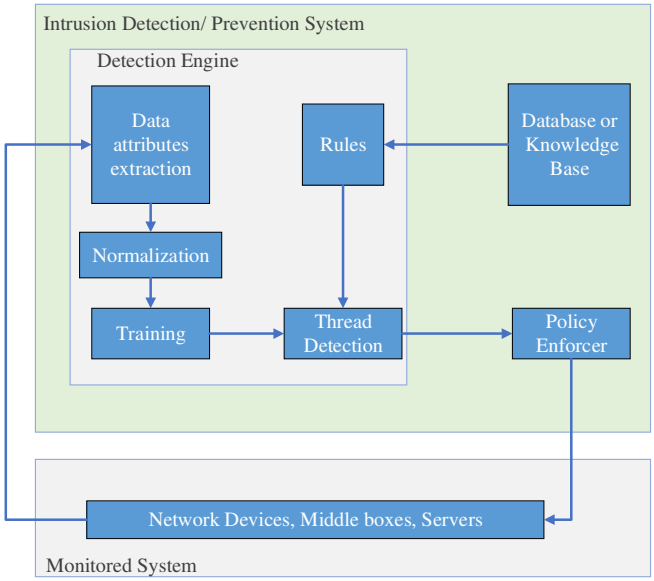

Figure 2. General architecture

In this paper, we try to improve the accuracy in intrusion detection of the IDS system. More specifically, first of all we show how Internet traffic can be pre-processed and normalized to get to the data we need for the training phase. We then propose a so-called Enhanced Kernel Null Space method - EKNS - at the training phase of the IPS/IDS system. EKNS is proved to improve the accuracy of detecting novelty samples. The scheme is elaborated as follows:

- Pre-process and normalize the attributes of the data set.
- Design an Enhanced Kernel Null Space method to analyze data inputs.

In thi method, the threshold is computed by Kernel Quantile Estimator [14] for a given probability q.

3.2. Pre-processing and normalizing data attributes.

In order to do the comparison with different intrusion detection methods, in the experiment, we use the NSL-KDD data set [15] which is commonly used for classification problem. Each sample in this NSL-KDD corresponds to a real connection in the simulated military network, containing 41 attributes with Normal and Attack-type labels. In the data set, there are 39 types of attacks divided in 4 groups:

- DoS - Denial of services, e.g. syn flood.

Table 1. Symbolic-typed Attributes

Attribute	Symbolic	Numeric value
Protocol_type	UDP	1
	TCP	2
	ICMP	3
Flag	OTH	1
	REJ	2
	RTSO	3
	RTSOSO	4
	RSTR	5
	S0	6
	S1	7
	S2	8
	S3	9
	SF	10
	SH	11
Service	65 values	from 1 to 65

- R2L: Unauthorized access from a remote machine, e.g. guessing password.
- Probing: surveillance and other probing, e.g. port scanning.
- U2R: unauthorized access to local super user (root) privileges, e.g. buffer overflow.

To make the data set simpler, reducing the redundancy without losing the information, we pre-process the data set as follows:

- **Conversion from the Symbolic type to the Numeric type**: there are 3 attributes in the Symbolic manner such as: Protocol, Service, Flag which are needed to be converted to the Numeric type to be compatible with the inputs of the algorithm. The symbolic values are labeled as in Table 1.
- **Normalization**:Normalization of data in the NSL-KDD data set is necessary since there are many big values in comparison with much smaller values in the set. We apply the Min-max normalization method to turn all values to the range [0,1] as follows:

$$\hat{v}_i = \frac{v_i - min(v_i)}{max(v_i) - min(v_i)}, \text{ for i } = 1, 2, \ldots, 41 \qquad (1)$$

where:
v_i: value of one attribute before normalization.
$\hat{v}_i$: value of one attribute after normalization.
$i = 1, ..., 41$: 41 attributes
After normalization, each data sample becomes a 41-attribute vector x_i which is an input for the detection process later. A 41-attribute vector before and after the normalization process can be illustrated in Figure 3 and Figure 4, respectively.

```
0 tcp ftp_data SF 491 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0 0 0 0 1 0
0 150 25 0.17 0.03 0.17 0 0 0 0.05 0 normal
```

Figure 3. An example of an original vector x_i.

```
0 1 18 10 491 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0 0 0 0 1 0 0 150
25 0.17 0.03 0.17 0 0 0 0.05 0 normal
```

Figure 4. An example of a normalized vector x_i.

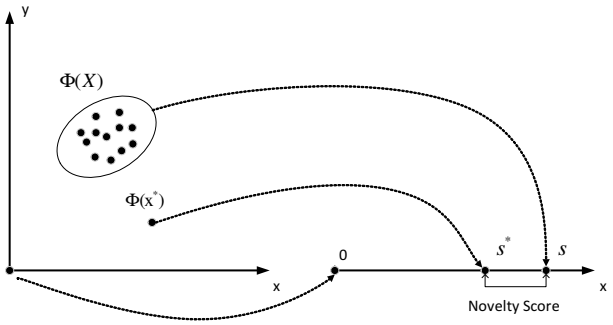

Figure 5. The samples are separated from the origin in the kernel feature space with a mapping Φ, then mapped on a point s, and the novelty score of a testing sample x^* is the distance of its projection s^* to s.

3.3. Control-chart based Kernel Null Space

Before describing EKNS, we briefly re-call the One-Class Classification using Kernel Null Space proposed in [2]. Let us consider a dataset of N training samples $\{x_1, x_2, \ldots, x_N\}$, with each $x_i \in R^D$, and D is the number of observed features. In the one-class setting, all the training samples belong to a single target class. The input features $X = [x_1, x_2, ..., x_N]$ are separated from the origin in the high-dimensional kernel feature space similar to one-class SVM [13]. As described in [2], a single null projection direction is computed to map all samples on a single target value s. A test sample x^* is projected on the null projection direction to obtain the value s^*. Figure 5 illustrates the one-class approach with kernel null space. The novelty score of x^* is the distance between s and s^*:

$$NoveltyScore(x^*) = \mid s - s^* \mid . \qquad (2)$$

A large novelty score indicates more likely novelty. In [2] and [1], a hard decision threshold $\theta_{threshold}$ is used to determine whether the test sample x^* belongs to the target class or not. Determining the threshold plays a very important role to the performance of the novelty detection process. To the best of our knowledge, this threshold has been selected heuristically up till

now. Therefore, in this study, we propose an intrusion detection scheme based on an enhanced version of this Kernel Null Space method.

The procedure of the EKNS is illustrated in Figure 3 with two phases: the training phase and the detection phase.

In the training phase: training data samples $\{x_1, x_2, \ldots, x_N\}$, which have been already pre-processed, will be mapped on a point s in the Null Space F. The intrusion detection system uses another data set called the validation set that comprises other normal data samples $\{y_1, y_2 \ldots, y_M\}$. Each sample y_i of the validation set is mapped on a point $\hat{s_i}$ in the feature null space, for which $NoveltyScore(y_i)$ is calculated. After mapping all samples of the validation set and calculating Novelty scores for all of them, a set $\{NoveltyScore(y_i)\}$ is formed. Based on this set of novelty scores, we use the Kernel Quantile Estimator to derive the threshold $\theta_{threshold}$, which will be described in Section 3.3.

During the detection phase in real time, when a test data sample x^* comes, the system maps it on a point s^* and then calculate its $NoveltyScore(x^*)$. Then by comparing the $NoveltyScore(x^*)$ with $\theta_{threshold}$ found in the training phase, x^* can be classified as Normal or Anomaly.

In the following subsections, we will elaborate how we achieve an optimal kernel parameter on the given training data set and how to calculate threshold $\theta_{threshold}$ by Kernel Quantile Estimator.

Determination of Kernel and Kernel parameter. In this paper, we select the Gaussian kernel (or Radial Basic Function (RBF)) for Kernel Null Space which is commonly used.

$$k(x, y) = \exp\left(\frac{-\| x - y \|^2}{2\sigma^2}\right) \tag{3}$$

where: σ stands for the kernel parameter in [0,1].

Using the method proposed in [17], the optimal sigma σ^* is estimated from the data set $\{x_1, x_2, \ldots, x_N\}$. The optimal σ^* is the one that maximizes the objective function $J(\sigma)$

$$J(\sigma) = \frac{2}{N}\sum_{i=1}^{n} \exp\left(-\frac{Near(x_i)}{2\sigma^2}\right) - \frac{2}{N}\sum_{i=1}^{n} \exp\left(-\frac{Far(x_i)}{2\sigma^2}\right) \tag{4}$$

Denote the nearest and farthest neighbors distances as:

$$Near(x_i) = \min_{j \neq i} \| x_i - x_j \|^2$$
$$Far(x_i) = \max_{i} \| x_i - x_j \|^2$$

Threshold calculation based on Kernel Quantile Estimator. As mentioned, the threshold for the Novelty Score is the crucial key for the accuracy in anomaly detection. A common method to choose a good threshold that we have observed up till now is checking various discrete threshold values in the increasing order until the test system outputs highest accuracy. But when we have to cope with continuous values, that heuristic check-up hardly finds a good threshold we can not check all continuous values.

The set of the novelty scores is denoted by $\{NS_1, NS_2, \ldots, NS_M\}$ and investigated for the probability density distribution. As observed in Figure 7, the Novelty Score values $\{NS_1, NS_2, \ldots, NS_M\}$ can not be approximated by a normal distribution, i.e. the underlying distribution of the sample is unknown. In this case, non-parametric methods could be used to explore this unknown underlying.

In this paper, we use the **Kernel Quantile Estimator** [14] to estimate $\theta_{threshold}$ over the set of Novelty Score values.

Let $NS_{(1)} \leq NS_{(2)} \leq \ldots \leq NS_{(M)}$ denote the corresponding order statistics of the novelty scores. Suppose that $K(.)$ is a density function symmetric about Zero and that $h \to 0$ as $n \to \infty$, the Kernel Quantile Estimator can be calculated as follows [14]:

$$KQ_p = \sum_{i=1}^{N}\left[\int_{\frac{i-1}{n}}^{\frac{i}{n}} K_h(t-p)dt\right] NS_{(i)} \tag{5}$$

where $h > 0$ is the bandwidth. The bandwidth h controls the smoothness of the estimator for a given sample of size n. $K_h(.) = \frac{1}{h}K(\frac{.}{h})$. And p is the proportion of the quantile.

Here we use the standard Gaussian kernel for the resulting estimate KQ_p which is a smooth unimodal,

$$K(u) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{u^2}{2}) \tag{6}$$

The selection of h is important in kernel density estimation: a large h will lead to an over-smoothed density estimate, while a small h will produce a ragged density with many spikes at the observations. As described in [14], the bandwidth computed as

$$h_{opt} = \left(\frac{pq}{n+1}\right)^{\frac{1}{2}} \tag{7}$$

Where: $q = 1 - p$

For a lot of continuous distributions used in statistics, specific quantiles such as the $p = 0.95$, 0.975, and 0.99 quantiles are tabulated. Therefore, in our experiment, we have investigated 3 cases of q: 0.05, 0.025 and 0.01 respectively. These 3 q values corresponds to 3 threshold value $KQ(p = 1 - q)$ (i.e. $\theta_{threshold}$).

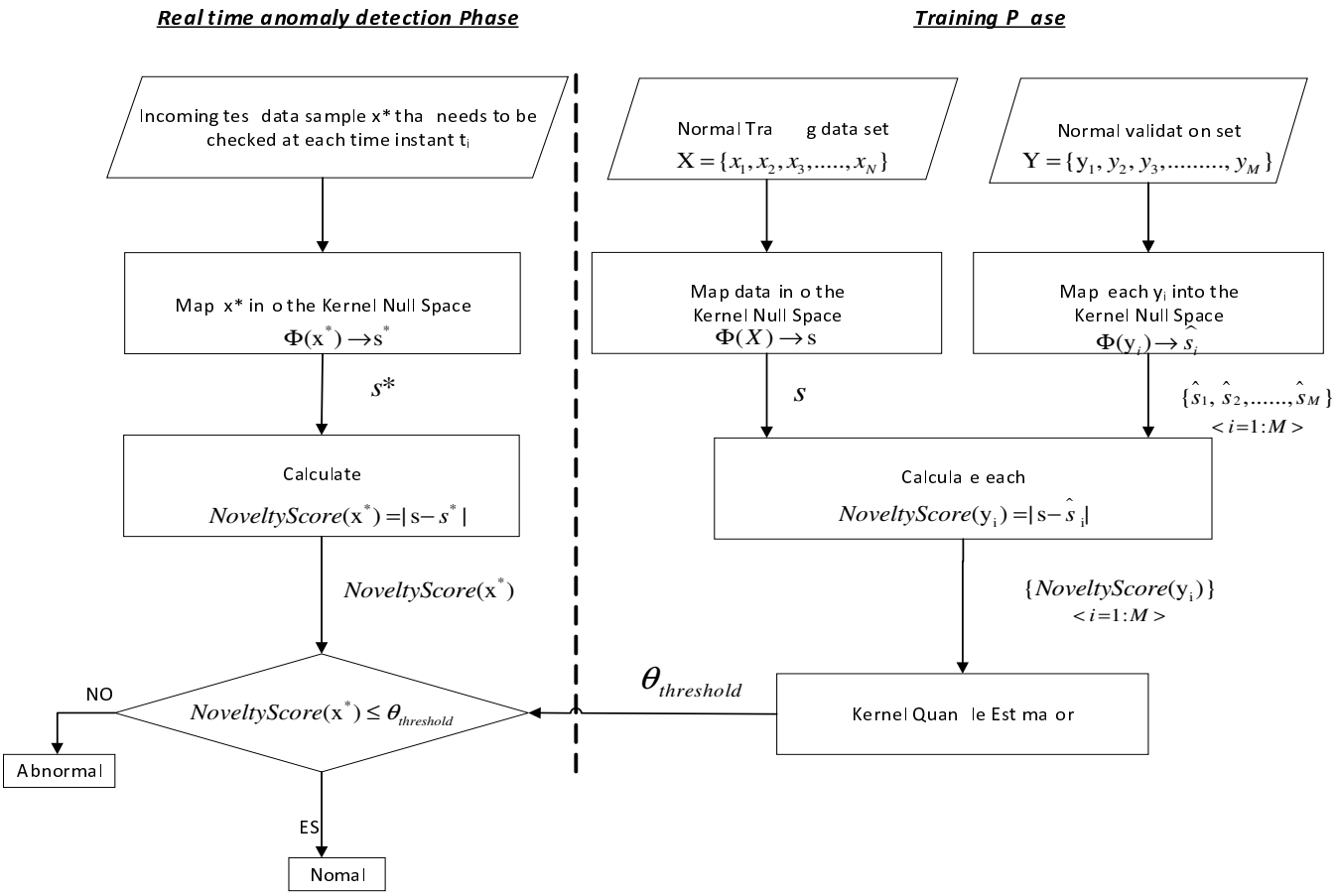

Figure 6. Detection procedure of EKNS

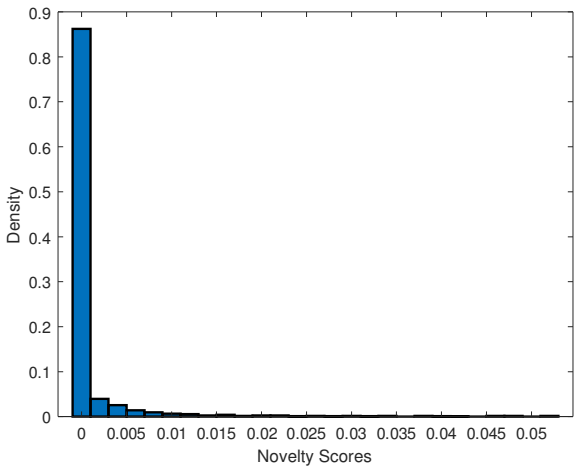

Figure 7. Probability Density Distribution of Novelty Scores

4. Performance Evaluation

Table 2. NSL-KDD Statistics

Data file	Total samples	Normal class	DoS class	Proble class	U2R class	R2L class
KDD+Train_20Percent	25192	13449	9234	2289	11	209
		53.39%	36.65%	9.09%	0.04%	0.93%
KDDTrain+	125973	67343	45927	11656	52	995
		53.46%	36.46%	9.25%	0.04%	0.79%
KDDTest+	22604	9771	7458	2421	200	2754
		43.08%	33.08%	10.74%	0.89%	12.22%

4.1. Data Description

In this experiment, we use the NSL-KDD data set to test the detection accuracy of the proposed solution. The training data set contains 13449 normal samples which are randomly selected from $KDDTrain+\_20Percent$ [15]. This data set takes 20% of $KDDTrain+$ in the NSL-KDD. After training the system with $KDDTrain+\_20Percent$, the system performance is checked by using 6000 normal and abnormal samples of the testing data set $KDDTest+$. Some statistics of the NSL-KDD data set can be illustrated in Table 2. To test performance, we use all 41 attributes/parameters of the data set.

Table 3. Performance Comparison

σ = 0.5957	Kernel Null Space			OCSVM	Origin Kernel Null Space with fixed threshold=0.05
	q=0.05 $\theta_{threshold}$= 0.0097	q=0.025 $\theta_{threshold}$= 0.0233	q=0.01 $\theta_{threshold}$= 0.0514		
Accuracy	0.9548	0.9598	0.92	0.9445	0.9212
FPR	0.0443	0.018	0.006	0.0433	0.006
Recall	0.954	0.9377	0.846	0.9323	0.8483
AUC	0.9910	0.9910	0.9910	0.9849	0.9910

4.2. Performance analysis

There are some important performance metrics in the novelty (anomaly) detection domain that have been widely used to analyze the performance of a certain detection method. Here, we used confusion matrix for measuring Recall, False positive rate and Accuracy to evaluate detection performance at one value of threshold.

- Accuracy = $\frac{TP+TN}{TP+FP+TN+FN}$
- ReCall-True Positive Rate = $\frac{TP}{TP+FN}$
- FPR - False Positive Rate: $FPR = \frac{FP}{FP+TN}$

Where TP (True Positive) is the number of anomalies correctly diagnosed as anomalies; TN (True negative) is the number of normal events correctly diagnosed as normal; FP (False Positive) is the number of normal events incorrectly diagnosed as anomalies; and FN (False Negative) is the number of anomalies incorrectly diagnosed as normal events.

As mentioned in Section 3.3, we have tested with 3 different q values: 0.01, 0.025 and 0.05. As can be seen in Table 3, with the normalized and pre-processed 41-attribute data set $\{X_1, X_2, \ldots, X_N\}$, the optimal kernel parameter estimated is $\sigma^* = 0.5957$. Subsequently, from the given data set of Novelty scores $\{NS_1, NS_2, \ldots, NS_M\}$, the found threshold is $\theta_{threshold} = 0.0097$, $\theta_{threshold} = 0.0233$, $\theta_{threshold} = 0.0514$ for $q = 0.05$, $q = 0.025$ and $q = 0.001$ respectively.

In the security context, accuracy is more important than recall when you would like to have less False Positives in trade off to have more False Negatives. Therefore, $q = 0.025$ brings best performance in terms of Accuracy, FPR among of the 3 different values q as shown in Table 3.

As another way to evaluate the performance of the detection solution, "ROC-AUC Curve" is often used as a measure of quality of the classification models at various thresholds settings [16]. ROC is a probability curve, it tells how much model is capable of distinguishing classes. This curve depicts relative trade-offs between benefit (TPR) and cost (FPR). To compare classifiers, a common method is to calculate the area under the ROC curve called AUC. AUC stands for "Area under the ROC Curve", represents degree or measure

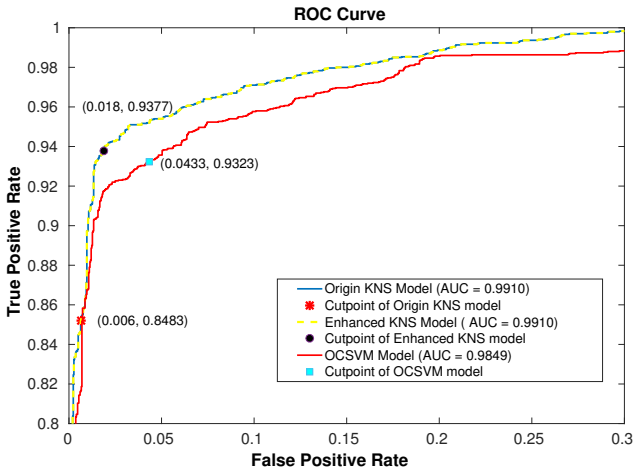

Figure 8. ROC-Curve of the EKNS, original Kernel Null Space, and OCSVM

of separability between two or more different classes. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.

In our test, we compare the performance of the EKNS with the original Kernel Null Space in which the threshold is heuristically selected and fixed at 0.05 [2] and with the One Class Support Vector Machine method (OCSVM) [17].

The ROC curves of three models are shown in Figure 8 with the corresponding cutpoints. The cutpoint of the EKNS model with $q = 0.025$ and $\theta_{threshold} = 0.0233$ has a coordinate of (0.018,0.9377), where 0.018 is the false positive rate, 0.9377 is the true positive rate; of the original Kernel null space is (0.006,0.8483); and the ROC cutpoint of the OCSVM method is (0.0433,0.9323).

We can see, the point at (0.018,0.9377) has the highest accuracy and lowest false positive rate as it produces accuracy of 95,98% and closer to the best point in the ROC Space (0,1). This result represents a balance between true positive rate and false positive rate. The AUC value (Area Under the ROC Curve) of the EKNS method (e.g. 0.991) is higher than OCSVM method (e.g. 0.9849), shows that the ability of classification is better.

The obtained results show that; EKNS slightly outperforms the OCSVM and the original Kernel Null Space methods in both terms of Accuracy and AUC while a bit inferior to the Original Kernel Null Space method in terms of FPR.

However, within the security context, accuracy is the more important metric since we desire have less False Positives in trade off to have more False Negatives. In this context, our solution is proved to be slightly better than the competitors.

5. Conclusion and future work

In this research, we have proposed an Intrusion Detection System using the so-called Enhanced Kernel Null Space method - EKNS with data-driven threshold retrieval. The proposed solution with data-driven findings such as $q = 0.025$ and $\sigma = 0.5957$ is proved to outperform the current OCSVM and Original Kernel Null Space methods in terms of detection Accuracy and AUC.

In the future, we would like to address the intrusion detection and the monitoring problem using deep learning, targeting on time series data with uncertainties. We also focus on the detection ability of our proposed approach for large stream data.

Acknowledgements

This paper has been accepted in part to the INISCOM conference 2019 - 5th EAI International Conference on Industrial Networks and Intelligent Systems.

References

[1] Bodesheim, P., Freytag, A., Rodner, E., Denzler, J.: Local novelty detection in multi-class recognition problems. In: Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. pp. 813–820. IEEE (2015)

[2] Bodesheim, P., Freytag, A., Rodner, E., Kemmler, M., Denzler, J.: Kernel null space methods for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3374–3381 (2013)

[3] Borkar, A., Donode, A., Kumari, A.: A survey on intrusion detection system (ids) and internal intrusion detection and protection system (iidps). In: Inventive Computing and Informatics (ICICI), International Conference on. pp. 949–953. IEEE (2017)

[4] Ferguson, P., Senie, D.: Network ingress filtering: Defeating denial of service attacks that employ ip source address spoofing. In: Internet RFC 2827 (2000)

[5] Gil, T.M., Poleto, M.: Multops: a data-structure for bandwidth attack detection. In: 10th Usenix Security Symposium. p. 2238 (2001)

[6] J. Mirkovic, G.P., Reiher, P.: Attacking ddos at the source. In: 10th IEEE International Conference on Network Protocols (2002)

[7] Liu, J., Lian, Z., Wang, Y., Xiao, J.: Incremental kernel null space discriminant analysis for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 792–800 (2017)

[8] Mercado, G.R., Conerly, M.D., Perry, M.B.: Phase i control chart based on a kernel estimator of the quantile function. Quality and reliability engineering international **27**(8), 1131–1144 (2011)

[9] Nguyen, Q.T., Tran, K.P., Castagliola, P., Truong, T.H., Nguyen, M.K., Lardjane, S.: Nested one-class support vector machines for network intrusion detection. In: 2018 IEEE Seventh International Conference on Communications and Electronics (ICCE). pp. 7–12. IEEE (2018)

[10] Park, Y., Baek, S.H., Kim, S.H., Tsui, K.L.: Statistical process control-based intrusion detection and monitoring. Quality and Reliability Engineering International **30**(2), 257–273 (2014)

[11] Phan Van Trung, Truong Thu Huong, e.a.: A multi-criteria-based ddos attack prevention solution using software defined networking. In: IEEE Inter. Conf. on Advan. Tech. for Commun. pp. 308–313 (2015)

[12] S. Shin, V. Yegneswaran, P.P., Gu, G.: Avant-guard: scalable and vigilant switch flow managament in software-defined networks. In: ACM SIGSAC Conf. Com. Commun. pp. 413–424 (2013)

[13] Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural computation **13**(7), 1443–1471 (2001)

[14] Sheather, S.J., Marron, J.S.: Kernel quantile estimators. Journal of the American Statistical Association **85**(410), 410–416 (1990)

[15] Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the kdd cup 99 data set. In: Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on. pp. 1–6. IEEE (2009)

[16] T.Fawcett: An introduction to roc analysis. Pattern Recognition Letter **27**, 861–874 (2006)

[17] Trinh, V.V., Tran, K.P., Huong, T.T.: Data driven hyperparameter optimization of one-class support vector machines for anomaly detection in wireless sensor networks. In: 2017 International Conference on Advanced Technologies for Communications (ATC). pp. 6–10 (Oct 2017). doi:10.1109/ATC.2017.8167642

[18] Van Tuyen Dang, Truong Thu Huong, e.a.: Sdn-based syn proxy - a solution to enhance performance of attack mitigation under tcp syn flood. The Computer Journal **62**(4), 518–534 (2019)

[19] Wang, Y., Wong, J., Miner, A.: Anomaly intrusion detection using one class svm. In: Information assurance workshop. pp. 358–364 (2004)