Abstracting Packet Header Information for Intrusion Detection in High-Speed Networks

*1Lawrence B. Adewole, 2Catherine R. Adeyeye, 3Adebayo O. Adetunmbi, 1Bosede A. Ayogu, 1Olaiyi Folorunsho
1Department of Computer Science, Federal University of Oye-Ekiti, Nigeria
2Department of Computer Science, Ekiti State University, Ado-Ekiti, Nigeria
3Department of Computer Science, Federal University of Technology, Akure, Nigeria

Received: 01-JUL-2020; Reviewed: 04-JUL-2020; Accepted: 06-JUL-2020
http://dx.doi.org/10.46792/fuoyejet.v5i2.541

Abstract- Increase in network traffic coupled with increasing adoption of end-to-end encryption of network packets are two major factors threatening the security of information technology systems. In the other hand, applications of the next technological wave, as being predicted, will become a reality even if not available for the general populace. Inspection of packet payloads while flow-based intrusion detection uses information found in the packet header. Remote access to resources has significantly increased and the next technological wave, as being predicted, will witness more traditional activities being automated, which in turns significantly increases network traffic.

Signature-based approach detects intrusion by applying a predefined set of rules in identifying common attack patterns. It however lacks the ability to detect zero-day attacks (Wang, et al., 2016; Ayogu, et al., 2019). Anomaly-based intrusion detection, on the other hand, applies a statistical approach to intrusion detection (Khraisat, et al., 2019). In another development, research in scaling up network speed has produced a network speed increase from the realm of gigabit per second to Terabits per second (Corcoran, et al., 2020). With a speed of 44.2 Terabits per second, which is equivalent to a data transfer of 5,525 Gigabyte per second, the impact of unauthorized access to vital information within a second might be unimaginable when the adoption of such technology becomes a reality even if not available for the general populace. Inspection of packet payload at such high speed will either be performed at random or slows down the network communication and as well become computationally expensive in terms of hardware resources and timeliness when performed on all packets.

Keywords- Data Abstraction, Data Mining, Flow-based, Intrusion detection, Network Security

1 INTRODUCTION

Attack and defence have become a never-ending game in the domain of information security (Clarke, 2008). Unlike physical systems where valuable resources can be protected using various mechanisms such as: perimeter fencing, sophisticated monitoring system, physical security guards, multi-layered access control system among others; security of information system is much more complicated. While organizations are busy developing protection mechanisms, attackers are also inventing new approaches to circumventing such mechanism for various reasons ranging from personal gain to show of skill. Organizations are increasingly adopting online solutions for most operations especially with the current pandemic. Restrictions imposed to curtail the spread of the Corona virus (COVID 19) has led to an increased traffic as most organizations are now leveraging on ICT facilities to conduct their businesses.

Remote access to resources has significantly increased and the next technological wave, as being predicted, will witness more traditional activities being automated, which in turns significantly increases network traffic.

Combating one of the major challenges of information system, which is security, there is a need for an intrusion detection mechanism to complement preventive measures that have been put in place. Intrusion detection system alerts the network administrator in case of a detected attack on the network; so that further action can be taken to neutralize the effect of such attack. Network intrusion detection has been an active area of research over the past two decades with different approaches proposed. Intrusion can be monitored either at the network level (network-based intrusion detection) or host level (host-based intrusion detection) using a Signature or Anomaly based approach.

*Corresponding Author
require the detailed packet content for classification. Features such as source IP address, destination IP address, source port, destination port, and protocol are common features associated with flow-based intrusion detection system. Modelling flow information has a high tendency of over-fitting to the training data especially when source and destination addresses are used. This paper presents an abstraction model for feature transformation using domain knowledge and target class statistics encoding; and evaluates the performance of such abstraction on UNSW-NB15 intrusion detection dataset.

The rest of the paper is organized into sections; the next section focuses on review of related literature. Section 3 presents proposed methodology while experimental setup and results are presented in Section 4. Conclusion and future direction are presented in Section 5.

2 LITERATURE REVIEW

2.1 FLOW-BASED INTRUSION DETECTION SYSTEM

Flow-based intrusion detection (FID) approach is gradually gaining popularity among researchers due to the scalability issue associated with packet-based intrusion detection in high speed network (Alaidaros et al., 2011). FID relies on the use of flow information in a network, harvested through an observation point within the network (Umer, et al., 2017). This entails the collection of packets belonging to the same flow within a given timeframe. Each frame is assigned a unique timestamp based on its arrival time. Aggregated packets are then exported and stored in a database for further processing.

2.2 FLOW-BASED DATASET AND FEATURES

Basic features used in flow-based intrusion detection include the Source IP address, Destination IP address, Source Port, Destination Port, Protocol, flags, which are information embedded in the packet header. These features are then aggregated to extract more information about traffic flow within a network. Some other features include: total number of packets in the flow, total duration of flow, timestamp marking beginning of flow (Pontes, et al., 2020), average flow size, average packet size, number of flows to the same destination IP, number of flows to the same destination port, number of flows to different destination ports, land (i.e. source IP = destination IP and source port= destination port), SYN - SYN/ACK, number of flows from the same source IP, number of flow from different source IPs, number of flows from different source ports (Abuadlla, et al., 2014). The most frequently appeared transport protocol with same destination IP, Summation, average, and deviation of flow size with the same destination IP (Kim, et al., 2004). Each statistically extracted feature coupled with basic features are used to detect different types of attack such as Denial of Service/ Distributed Denial of Service (DoS/DDoS), SYN flood, Transport Control Protocol (TCP) flooding, IP (Internet Protocol) Spoofing, Port Scan, among others.

Sperotto et al, (2009) presented the first public available labelled dataset for flow-based intrusion detection. Before this, researchers have either captured traffic flow on private networks or generated flow traffic from raw DARPA dataset. Another flow-based dataset is the Tezpur University Intrusion dataset, which is a combination of both flow-based and packet-based features (Bhuyan et al., 2015). More recent flow-based datasets are CIDDS-001 (Ring et al., 2017), CIDDS17 (Sharafaldin, et al., 2018), Punjabi University Flow Dataset (Sharma, et al., 2018) and CIDCDoS19. Details of these datasets can be found in various survey research on flow-based intrusion detection system (Gogoi, et al., 2012; Umer, et al., 2017; Ring, et al., 2019).

2.3 RELATED WORK

Research into network intrusion detection gained popularity among researchers with the release of the publicly accessible DARPA dataset in 1998 (Lippmann et al., 2000), and its variant: KDD 99 (Hainess et al., 2001) with hundreds of publications on different approaches aimed at improving accuracy, enhancing detection rate, reducing model complexity, among others. The predominant approach utilizes packet-based features which require deep inspection of packet payload in real-time. However, the adoption of a flow-based intrusion detection approach over the packet-based approach became imperative with increasing network speed to avoid the high overhead cost associated with packet inspection (Sperotto et al., 2009). Flow-based intrusion detection relies on information embedded in the packet header and their analysis for intrusion detection. The flow can also be aggregated to detect changes in packet flow over a defined period (Abuadilla et al., 2017). This approach has been proved to sufficiently detect novel attacks as well as indirect attacks on network facilities. Several researchers have proposed intrusion detection systems using flow-based information.

Kim et al. (2004) proposed a flow-based intrusion detection system for a robust and efficient intrusion detection system in high-speed network. IP flows are aggregated and deviations in the flow pattern are monitored as an indicator for possible intrusion. Features such as number of flows with a common destination, number of distinct source IP addresses with common destination IP addresses, number of source and destination ports having the same destination port, among others were extracted from the network. Additional statistics were also gathered by aggregating all flow from the same host, irrespective of the IP address used. A custom detection algorithm was proposed and implemented in C language. Their work excelled in the detection of intrusion that impacts network performance such as DDoS attacks.

Lakhina et al (2004) presented a flow-based intrusion detection system by extracting meta-data on IP flows from a specific source to a specific destination. Metadata extracted include the number of bytes (B), number of packets (P) and number of IP-level flows (F). Further information was extracted by aggregating data with correlating timestamp over B, P, and F, thereby yielding new categories BP, BF, FP, and BPF. Sub-space sampling was applied to reduce the computational complexity. Various indicators were used to identify different categories of anomalies in the network such as Alpha,
Flash Crowd, Worm, Scan, DoS, DDoS, Point-to-Multipoint, Outage, and Ingress-Shift. For example, a spike in B, P, and BP between the same source-destination pair within a time frame of 10 minutes classifies as Alpha anomaly. However, the goal of the research is tailored towards the validation of the sub-space sampling technique and its representativeness of the network.

Sperotto (2009) presented a flow-based intrusion detection dataset and performed intrusion detection on the dataset by detecting deviations in network based on a flow time-series, pre-trained on normal network profile built using Hidden Markov model (Sperotto, 2010). Winter et al., (2011) developed a flow-based intrusion detection model using outlier detection technique. A classification model was trained on malicious traffic using One-class SVM using Sperotto dataset. Tran et al., (2012) proposed an improved block-based neural network approach to intrusion detection on net-flow data, achieving a sustained accuracy of 99.92 with a false alarm rate of 5.14%.

Abuadlla et al (2014) presented a flow-based intrusion detection framework using a two-stage Neural Network. The first stage detects the abnormality in network traffic while the second stage classifies the anomaly detected into known or unknown attack. Dataset used was created from the DARPA raw dataset. The dataset contains 145,408 flow data featuring 96,852 attack traffic and 48,556 normal flows. The first stage of their proposed framework uses 7 features to classify traffic as either normal or anomaly while the second phase uses 12 features. The best detection rate of 94.2% was recorded at the anomaly detection phase while the classification phase attained best detection rate of 99.42%.

Umer et al., (2018) proposed a two-stage flow-based malicious activity detection for next generation network. One-class support vector machine algorithm was applied at the first stage for malicious flow detection while the second stage clusters the attacks into categories using a self-organizing map (SOM). Three flow datasets were used for evaluation with each having 9 attributes. The strength of the proposed approach lies in its ability to achieve high detection rate despite the use of unsupervised learning approaches. Despite the advantages of flow-based intrusion detection system over packet-based approach in high speed network, the flow-based model requires regular update due to constant changes in network pattern across different domains (Viegas et al., 2018).

In contrast with existing literature, this paper leverages on basic packet-header features for classification rather than utilizing statistical information on flows. The proposed model utilizes both source and destination IP address as these features play a vital role in packet identification (Shao, 2019) and the significance of these features has not been investigated in anomaly-based intrusion detection (Moustafa & Slay, 2015; Pontes et al., 2020). It also proposes feature transformation to avoid model over-fitting to underlying IP addresses as is the case with signature-based intrusion detection (Kruegel and Toth, 2003). The abstraction presents a reduced and anonymized record for further pre-processing.

3 METHODOLOGY
3.1 PROPOSED FRAMEWORK
This section presents the abstraction model proposed in this research for the flow-based features in intrusion detection. Prior understanding of the problem domain has proved to be an asset in data modelling and preparation for machine learning. The basic flow of operation for the detection process is as delineated in Figure 1.

In this research, the UNSW-NB15 full dataset was used. The dataset consists of 49 features and 2,540,044 instances. The dataset has two labels in which one has binary value (attack or normal), while the second label classifies record as normal or specifies the attack type. The dataset features 9 different attack types. This research used the label with binary classification values since packet header do not hold sufficient information which can be leveraged on to distinguish between different attack types.

Also, features used in this research are limited to the set of features categorized as flow features by the dataset authors. This includes Srcip, Sport, Dstip, Dsport, and Proto, which guarantees the possibility of packet-by-packet inspection without necessarily aggregating the packets. Table 1 presents the definition of this selected feature set.

| S/N | Name   | Type   | Description                  |
|-----|--------|--------|------------------------------|
| 1   | Srcip  | nominal| Source IP address            |
| 2   | Sport  | integer| Source port number           |
| 3   | Dstip  | nominal| Destination IP address      |
| 4   | Dsport | integer| Destination port number     |
| 5   | Proto  | nominal| Protocol type (TCP, UDP,...) |
| 49  | Label  | binary | Class label                  |

3.2 PACKET HEADER FEATURE ABSTRACTION
Two levels of abstraction are applied to the UNSW-NB15 dataset. The first level of abstraction is the feature transformation of IP addresses, Ports, and Protocol using domain knowledge. Each IP address that appeared in the
source or destination IP addresses of the packet header is replaced with the corresponding IP address class as detailed in Table 2 while the source and destination ports were grouped based on the system port classification (Schneider, 1996) as detailed in Table 3.

| Class | 1st Octet Decimal Range | Encoding |
|-------|------------------------|----------|
| A     | 1 – 126*               | 0        |
| B     | 128 – 191              | 1        |
| C     | 192 – 223              | 2        |

Table 2. IP address classification

| Port Number Range | Category | Encoding |
|------------------|----------|----------|
| 0-1023           | System   | 0        |
| 1024 – 49151     | Reserved | 1        |
| >49151           | Dynamic  | 2        |

Table 3. Port Classification

The last feature, which is the protocol, has 139 distinct values. A reduction in the value space was achieved by partitioning the universal set using basic set theory as shown in Figure 2. The dataset was divided into two subsets P\textsubscript{Normal} and P\textsubscript{Attack} based on the target class. We defined three subsets A, N, and C where A is the set of protocols in P\textsubscript{Attack} but not in P\textsubscript{Normal}, N is the set of protocols in P\textsubscript{Normal} but not in P\textsubscript{Attack} and C is the set of protocol common to both P\textsubscript{Normal} and P\textsubscript{Attack}. Finally, integer encoding was applied for the transformation of the nominal classes (Pargent, 2019).

The second level of transformation is the exemplar extraction (Wang, et al., 2016). Exemplar extraction abstract massive data to build lightweight training model by clustering instances using clustering algorithms such as k-means, k-centre, or affinity propagation. This paper adopts the affinity propagation approach to data point clustering. Given a dataset with N instances defined by D = \{I_1, I_2, ..., I_N\}, we define a distance measure d(I_i, I_j) between instance I_i and I_j. Data points are clustered based a fitness function E such that:

\[
E(c) = \sum d(I_i, I_j) \tag{1}
\]

where d(I_i, I_j) is the Euclidean distance between the instances which has been set to 0.

3.3 Training and Testing Set Extraction

After feature extraction phase, duplicate instances were removed from the dataset to avoid biasness in the training phase. This led to the reduction of the dataset from 2,540,044 to 1,980,675 distinct instances. This dataset was used for testing. The dataset is transformed as detailed in Section 3.2. This reduces the dataset to 217 instances which was used for model training. Table 4 presents the detail of the datasets.

| Dataset   | Transformation | No of features | No of instances | Number of Attack | Number of Normal |
|-----------|----------------|----------------|-----------------|-----------------|-----------------|
| UNSW-NB15 | Feature Selection, Duplicate Instance Removal, Discretization | 49              | 2,540,044       | 321,283         | 2,218,761       |
| Testing   | Duplicate Instance Removal | 6               | 1,980,675       | 68,284          | 1,912,391       |
| Training  | Duplicate Instance Removal | 6               | 217             | 83              | 134             |

Table 4. Dataset Description

3.4 Experimental Setup

Four existing classification algorithms are selected to build classification models for intrusion detection using packet header information. These include: Classification and Regression Trees (Peng, et al., 2018), Support Vector Machine (Jha & Ragha, 2013), Naïve Bayes (Panda & Patra, 2007), and k-Nearest Neighbour (Liao & Vemuri, 2002).

All experiments were carried out on a Core i5 PC with 2.40 GHz processor speed and 6GB of RAM. Python programming language was used for the implementation of each phase of the classification process. Libraries such as pandas, sklearn and numpy were utilized.

3.5 Model Evaluation

To evaluate the potency of basic flow-based information at classifying network traffic, three (3) standard machine learning evaluation metrics were used. These include the accuracy, sensitivity and specificity as depicted in equations 2, 3 and 4.

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}
\]

\[
\text{Sensitivity} = \frac{TP}{TP + FN} \tag{3}
\]

\[
\text{Specificity} = \frac{TN}{TN + FP} \tag{4}
\]

4 Result and Discussion

Feature abstraction phase of the research produced a high level abstraction of the underlying data. The number of distinct IP addresses in original UNSW-NB15 dataset was reduced from 45 to 2 which include 0 for class A IP addresses and 1 for class B IP addresses. The ports (source and destination) whose values are continuous and can assume any integer value between 0 and 65535 was reduced to three (3) distinct values: 0, 1, and 2 based on Table 3. The total number of distinct protocols was also reduced from 139 down to 5 based on target-based
partitioning. Verification results of the level of representativeness of the underlying data, by the abstracted training dataset, are presented in Table 5.

| Algorithm | CART | GNB | KNN | SVM |
|-----------|------|-----|-----|-----|
| TP        | 68284| 68284| 63303| 54936|
| TN        | 1873843| 1873651| 1876492| 1881320|
| FP        | 38548| 38740| 35899| 31071|
| FN        | 0    | 0   | 4981 | 13348|
| Accuracy  | 98.05| 98.04| 97.94| 97.76|
| Sensitivity| 1.0  | 1.0  | 0.93 | 0.80|
| Specificity| 0.97 | 0.98 | 0.98 | 0.98|
| ROC       | 0.99 | 0.99 | 0.98 | 0.99|

Results obtained from the model evaluation show that the abstraction proposed produced an acceptable accuracy on each model with minimum accuracy of 97.76 on SVM. Also, high sensitivity and specificity are the indication that the model classifies

Figure 3 shows the Receiver Operating Characteristics (ROC) curve for the algorithms.

From Table 5, it is evident that the underlying data representation retains information embedded in the original dataset. Accuracy, sensitivity, and sensitivity obtained across the four algorithms are consistent.

5 Conclusion and Future Work

Review of existing approaches to intrusion detection revealed that signature-based approach, though reported to achieve low false alarm rate, and achieves high detection rate, suffers from: its inability to detect new forms of attack not captured in its signature database, as well as the computational complexity involved in rule matching as the signature database grows. Anomaly-based approach on the other hand overcomes the main weakness of signature-based approach by relying on deviation in statistical pattern of network packets using both flow and deep packet inspection. Although, packet-based features have produced a high classification accuracy, the need for a real-time detection in high-speed network coupled with ability to classify encrypted packets are factors favouring flow-based approach.

Flow-based approach however is not without its own limitations which include the need for a regular model update due to volatility in network flow pattern which could make models obsolete. This paper presented an abstraction model for the transformation of packet header features without information loss. The weaknesses of packed-based IDS have been avoided by utilizing flow-based features. Also, the use of basic flow-based features overcame the challenges of regular model update due to volatility in network traffic pattern. While the model is closely related to signature-based approach, the abstraction guarantees the model’s ability to generalize the classification model. The abstracted model attained acceptable results in term of accuracy, sensitivity, and specificity, hence, its suitability for real-time packet classification in high speed network.

The first phase of the abstraction process can be adopted in anonymization of IP addresses for real life network traffic proposed for academic communities. In the future, we intend to evaluate the efficiency of the model across different networks by classifying dataset from another network using classification models developed in this research. Also, we intend to perform extensive evaluation from the algorithm point of view.

References

Abuadlla, Y., Kvascev, G., Gajin, S., & Jovanovic, Z., (2014). Flow-Based Anomaly Intrusion Detection System Using Two Neural Network Stages. Computer Science and Information Systems 11(2):601–622

Abuadlla, Y., Taher, O.B., & Elzentanti, H., (2017). Flow Based Intrusion Detection System Using Multistage Neural Network. Asmari University Journal of Basic and Applied Sciences, 13(7):77-87.

Alaidaros, H., Mahmuddin, M., & Al Mazari, A. (2011). An overview of flow-based and packet-based intrusion detection performance in high speed networks. In Proceedings of the International Arab Conference on Information Technology, pp. 327-331.

Ayogu, B.A., Adetunmbi, A.O., & Ayogu, I.I., (2019). A comparative analysis of decision tree and bayesian model for network intrusion detection system. FuoYE Journal of Engineering and Technology, 4 (2): 71-75.

Bhuyan, M.H., Bhattacharyya, D., & Kalita, J., (2015). Towards generating real-life datasets for network intrusion detection. International Journal of Network Security, 17(6): 675-693.

Clarke, R. (2008). Network attack and defense. In R.J. Anderson (Ed.). Security Engineering: A Guide to Building Dependable Distributed Systems, 2ed, (pp. 633-678).

Corcoran, B., Tan, M., Xu, X. et al., (2020). Ultra-dense optical data transmission over standard fibre with a single chip source. Nature Communication, 11, 2568. https://doi.org/10.1038/s41467-020-16265-x.

Gogoi, P., Bhuyan, M.H., Bhattacharyya, D.K., & Kalita, J.K., (2012). Packet and Flow Based Network Intrusion Dataset. In: Parashar, M., et al., (eds) Contemporary Computing. IC 3 2012. Communications in Computer and Information Science, vol 306. Springer, Berlin, Heidelberg

Jha, J. & Ragh, L. (2013). Intrusion detection system using support vector machine. International Conference & Workshop on Advanced Computing, pp 25-30.

Kim, M., Kang, H., Hong, S., Chung, S., & Hong, J., (2004). A Flow-
based Method for Abnormal Network Traffic Detection. IEEE/IPIP Network Operations and Management Symposium (IEEE Cat. No.04CH37507), Seoul, South Korea, 2004, pp. 599-612 Vol.1, doi: 10.1109/NOMS.2004.1317747.

Khraisat, A., Condal, I., Vamplew, P., & Kamruzzaman, J., (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(20):1-22.

Kruegel, C., and Toth, T., (2003). Using Decision Tree to improve signature-based intrusion detection. In Proceedings of the 6th International Workshop on the Recent Advances in Intrusion Detection (RAID), 2820:173-191.

Lakhina, A., Crovella, M., Doit, C.: Characterization of network-wide anomalies in traffic flows. In: Proc. of 4th ACM SIGCOMM Conf. on Internet measurement, IMC 2004 (2004).

Liao, Y., & Vemuri, V.R., (2002). Use of K-Nearest Neighbor classifier for intrusion detection. Computers and Security, 21 (5): 439-448.

Lippmann, R., Fried, D., Graf, I., Haines, et al., (2000): Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In: Proc. of the DARPA Information Engineering. Conf. and Exposition, DISCEX 2000.

Moustafa, N., Slaj, J., (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). Military Communications and Information Systems Conference (MiCIS), 1-6.

Panda, M., & Patra, M. R. (2007). Network intrusion detection using naive bayes. International Journal of Computer Science and Network Security, 7 (12), 258–263.

Pargent, F., (2019). A benchmark experiment on how to encode categorical features in predictive modeling. M.Sc. Thesis, Ludwig-Maximilians –Universitat Munchen, pp12.

Peng, K., Leung, V., Zheng, L., Wang, S., Huang, C., & Lin, T., (2018). Intrusion detection system based on decision tree over big data in fog environment. Wireless Communications and Mobile Computing, pp 1-10.

Pontes, C, Souza, M., Gondim, J., Bishop, M., & Marotta, M., (2020). A new method for flow-based network intrusion detection using the inverse Potts model. https://arxiv.org/pdf/1910.07266.pdf

Ring, M., Wunderlich, S., Grudl, D, Landes, D & Hotho, A., (2017). Flow-based benchmark data sets for intrusion detection. European Conference on Cyber Warfare and Security (ECCWS), ACPI, pp. 361–369.

Schneider, P., (1996). TCP/IP traffic classification based on port numbers. Division of Applied Sciences, Cambridge, MA, 2138 (5): 1-6

Shao, E., (2019) Encoding IP address as a feature for Network Sharafaldin, I., Lashkari, A.H., & Ghorbani, A.A., (2018). Towards generating a new intrusion detection dataset and intrusion traffic characterization. International Conference on Information Systems Security and Privacy, pp 108-116.

Sperotto, A., Sadre, R., van Vliet, F., & Pras, A., (2009) A Labeled Data Set for Flow-Based Intrusion Detection. In G. Nunzi, C. Scoglio, and X. Li (Eds.): IPOM 2009, LNCS 5843, pp. 39–50, 2009. Springer-Verlag Berlin Heidelberg

Sperotto, A., (2010). Flow-Based Intrusion Detection. Ph.D. Thesis, University of Twente.

Tran, QA, Jiang, F, Hu J., (2012) A real-time Netflow-based intrusion detection system with improved BBN and high-frequency field programmable gate arrays. In: Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 201-208.

Umer, M.F., Sher, M. & Bi, Y. (2017). Flow-based intrusion detection: techniques and challenges. Computers & Security (2017), http://dx.doi.org/doi: 10.1016/j.cose.2017.05.009.

Umer, M.F., Sher, M. & Bi, Y. (2018). A two-stage flow-based intrusion detection model for next-generation networks. PLoS ONE 13(1): e0180945. https://doi.org/10.1371/journal.pone.0180945

Viegas, E., Santin, A.O., Bessani, A.N., & Nevas, N.F., (2018). BigFlow: Real-time and reliable anomaly-based intrusion detection for high-speed networks. Future Generation Computer Systems. 93: 473-485.

Wang, W., Liu, J., Pittisils, G., & Zhang, X., (2016) Abstracting massive Data for Lightweight Intrusion Detection in Computer Networks. Information Sciences, vol. 433-434: 417-430.

Winter, P., Hermann, E., & Zollinger, M., (2011). Inductive intrusion detection in flow-based network data using One-class support vector machines. In: New Technologies, Mobility and Security (NTMS), pp. 1-5.