A Neural Network Based Intrusion Detection System For Wireless Sensor Networks

OKAN CAN
Turkish Air Force Academy
Computer Engineering Department
Istanbul
Turkey
ocan@hho.edu.tr

CANSIN TURGUNER
Turkish Air Force Academy
Computer Engineering Department
Istanbul
Turkey
cturguner@hho.edu.tr

OZGUR KORAY SAHINGOZ
Turkish Air Force Academy
Computer Engineering Department
Istanbul
Turkey
sahingoz@hho.edu.tr

Abstract: Wireless Sensor Networks (WSNs) are used in many application areas including smart homes, military security applications, tracking applications and monitoring applications (oceans, animals, habitats). The number of these usage areas are increasing by day by. Currently, cyber attacks are a new battlefield and threat WSNs certainly. Protecting WSNs against attacks is an important topic and WSN have some big security deficiencies. In this paper, it is aimed that designing an intrusion detection system (IDS) developed by neural network approach for WSN and is shown that WSNs have some differences from wired and non-constrained networks so IDSs proposed for WSN have different features. The proposed system tested by KDD99 data set and it is examined that the system is successful

Key–Words: Wireless Sensor Networks (WSN), Intrusion Detection Systems (IDS), Neural Networks.

1 Introduction

Wireless Sensor Networks (WSNs) are a large scale network having thousands of tiny devices, sensing and collecting data from physical environment. These tiny and small sized devices have low processing and storage capacity and low cost. It is easy to design and create a WSN because of it's cheap price and so it is used for many different fields by various applications. These areas can be lined up as science (exploring oceans, animals, habitats, wildlife and space e.g.), health-care (Monitoring in Mass-Casualty Disasters) [Paper 1], military (Boomerang Sniper Identifying System, Nuclear Bilogic Chemical Attack Identifying), tracking and monitoring applications (monitoring highway traffic, fire alarm systems and home automation systems), agriculture, transportation and many others [1].

Anymore, it is more important that protecting, processing, hiding and moving the data because of incredible progressing of information technologies. Spy-wares, mobile threats, growing up of number of attack and attack types have increased importance of cyber security. Because of these increasing, a lot of approach is used to serve a healthy system security. As a information system, WSNs need a protection mechanism to resist against cyber attacks. This counteractive mechanism can be defined as two lined approach. The first line is prevention based approach (encryption, authorization, authentication) and the second line is detection based approach (Intrusion detection). If any attacker passes the first line and then the second line tries to find whether there is any intrusion or not. Intrusion detection system is a software or hardware that is an alarm component of any network warning the system administrator against unwanted and unauthorized movements. WSNs have some differences and constraints so their IDS approaches are different from wired and non-energy constraint networks [2][3].

There are a lot of IDS design approaches for WSNs. Such as neural network based, data mining based, mobile agent based, rule based, game theory based, statistical based, genetic algorithm based. In this working, artificial neural network based approach is selected to design an IDS to serve a smart system having learn ability. To train and test the neural network KDD' 99 Cup data-set is selected and all processes is achieved by *"Matlab nftool"*. Test results and graphics got from Matlab are shown and all process steps is explained detailed.

The plan of the paper is organized as: Section 2 is about intrusion detection for wireless sensor networks. In this section it is explained that what intrusion detection is, how wireless sensor networks works, what is differences of WSNs from wired networks, what type attacks can be for WSNs and how features must have an IDS for WSNs. Section 3 is about neural networks.Section 4 is about proposed

system and test results. In this section, flowchart, used data set and pre-processing are explained.Section 5 is about conclusion and future work.

2 Secure Communication And Intrusion Detection For Wireless Sensor Networks

Wireless Sensor Networks are used in many important areas like unmanned aeronautical vehicle communications and military zones security [4]. Physical or transmission security of these areas are provided based on WSN. Although many researches consider energy management of the WSN systems mainly, it is crucial that WSNs have vulnerabilities by means of secure communication [5]. To obtain secure communication of WSN, there are many methods that have weaknesses. Some of them are not useful considering energy consumption and the rest of them are opened against attacks [6]. Implementing IDS structure in WSN like in traditional networks, is the main point for secure communication. The constraints for IDS implementation in WSN are;

- Have to cover considerable security attacks.
- Applications that run in nodes have to consume low energy.
- Whole system must have fault tolerance against losses of main nodes by attacks[7].
- Proposed applications have to use low memory of the devices.
- Have to support mobility and scalability. The WSN system can be in distributed form dependant on environment.

First IDS approach was implemented by Doroty Denning in 1980 to detect unauthorized access to computer systems. Since 1980, a lot of studies have been achieved and various detection approaches have appeared. There are two important topics for computer security paradigm. These can be lined as preventing approach and detecting approach. In a used security plan for any computer system preventing approach (encryption, authorization, authentication) is the firs security line and detecting approach (intrusion detection) is the second line. This second security line aims that serving deterrence for probable attacks, detecting attacks early, detecting infraction appearing in the system, serving continuity for system security rules, gathering evidence about attacks [8].

IDS can be described as software or hardware reporting internal or external attacks. IDSs work by comparing current network packages with signatures of known attacks and defining network movements as normal or anomaly. IDSs are divided into two group as **Network Based Intrusion Detection (NIDS)** and **Host Based Intrusion Detection (HIDS)** according to monitoring approach to system. NIDS monitors all network and tries to find intrusions happening on this network but HIDS monitors only single host. Also,according to used detection techniques, IDSs are divided into two groups as **Anomaly Detection** and **Misuse Detection**. Anomaly detection tries to find system anomalies and misuse detection tries to compare packages with signatures of known attacks. It is decided that which technique and approach is chosen according to system requirements and properties.

Different approaches implemented by researchers and created from global definition of IDS. In [9], classifying is made as intrusion type, intruder type, detection techniques, source of the collected data, analyzing location of the collected data, usage frequency and this classifying is the most comprehensive in the literature. . In a network, intruder type is grouped into two categories. These categories are internal intruder (selfish or malicious node) and external intruder (An outside attacker trying to reach the system). In WSN, according to intrusion type, intrusion can be by stealing the data, by creating false data and so altering the system, by denying to access the system, by influencing the energy efficient. For detection methodologies it has been described above as misuse and anomaly detection but additionally some papers point out hybrid or specification based detection [10].

2.1 Anomaly Detection Approaches in WSN

In [11], wireless sensor networks anomalies grouped as follows:

- **Network Anomalies:** This anomaly type occurs with connection problems. General behavior of a WSN is specific and sudden changes on this behavior is portent (anomaly) for it.
- **Data Anomalies :** Size of data packets shows whether there is any anomaly or not. Irregular data sets cause this type anomalies.
- **Node Anomalies :** This anomaly types mean hardware or software problems on WSN sensor.
- **Other Anomalies :** This type anomalies are described es not fitting problems to Network, Data and Node anomalies.

2.2 Misuse Detection Approaches in WSN

Misuse detection technique is not suitable for wireles sensor network because of their limited resources such

as processor, storage, memory, energy. Of course misuse detection is so successful to catch known attacks but storing attack signatures in WSN may not be suitable.

2.3 Hybrid Detection Approaches in WSN

Some detection approaches and techniques can not be fit in anomaly and misuse detection or some researchers mix anomaly and misuse detection and then Hybrid Detection Approach occurs. Classification of Hybrid Approach is made by [12] as Decentralized Approach, -Pre-defined Watchdog Approach, Hybrid System Approach.

3 Artificial Neural Networks

Neural Networks are used for solving problems that cannot be formulated as an algorithm. At this point a question occurs. "How do people learn solving problems?". People make learning by their brains and computers have some processor and storage devices too. With Neural Networks, it is aimed that creating a structure simulating human brain's learning ability by these processor and storage devices. Neural networks can be modelled as both software and hardware. Even, first neural network approaches are created as hardware but because of hardware isn't flexible and can't be changed dynamically, software approaches have came to prominence. Learning is a comprehensive process. A learning system makes learning by adaption to environmental changes. Basically, a neural network provides learning by changing own components [13]. These changes will be explained shortly in this section and can be listed as this:

- Creating new connections
- Deleting the current connections
- Changing connection weights
- Using one or more neuron functions.

3.1 Components of Artificial Neural Networks

A neural network consists of **neurons**, and **directed, weighted connections** between these neurons. Connections transfer the data between neurons. Neurons have five components and these are listed as follows:

- **Inputs from other neurons :** Outputs of previous neuron connecting by directed and weighted connections.
- **Propagation Function :** Propagation function and "network input" are described as a component that receiving the outputs of previous neuron and transforming them to the network input according to the connecting weights. Network inputs are the result of propagation function.

  Definition 1 *$I = \{i_1, i_2, i_3 ... i_n\}$ a neuron set. Such that $\forall \in z\{1, ..., n\} : \exists w_{i_z,j}$. Here, input data is net_j and it is calculated by f_{prop}.*

  $net_j = f_{prop}(O_{i_1}, ..., O_{i_n}, w_{i_1,j}, ...w_{i_n,j})$

  For this study **weighted sum** is used and according to literature it is the most popular:

  $net_J = \Sigma(O_i.w_{i,j})$
- **Activation Function :** Activation function change the state of neuron. Neurons decide to what it will do by their activation value determined by activation function. Activation state represents the condition of neuron's activity. Activation function calculates a value and if this value is above of the threshold value of neuron and then the activation state is modified.

  Definition 2 *J indicates a neuron and activation function is described as follows:*

  $a_j(t) = f_{act}(net_j(t), a_j(t-1)\Theta_j$

  In this definition Θ is unique threshold value of the neuron. Changing the neuron state depends on this threshold value and it can be change by the time. In the literature activation function is also named as **transfer function**. Activation functions can be lined up as follows:

  - *"Heaviside function"*
  - *"Fermi function or logistic function"* holds the value in the range of $(0, 1)$ so value is always positive. According to literature this function is the most popular because of value is mapped as positive. In this study this function is accepted.

    $$\frac{1}{1 + e^{-x}} \tag{1}$$
  - *"Hyperbolic tangent function"* holds the value in the range of $(-1, 1)$.
- **Output Function :** By the output function, activation is committed again and after that values are directed to another neuron.

  Definition 3 *J indicates a neuron and output function is described as follows:*

  $f_{out}(a_j) = o_j$

  Output function calculate it's value from activation state.
- **Outputs to other neurons:** Outputs are final values of neuron sent to next neuron.

ISBN: 978-1-61804-307-8

3.2 Learning Process of Artificial Neural Networks

The first step of learning is activation. It is important that inputs of a neuron must be sufficient to activate the neuron. If value of activation is above the threshold value of neuron, it is sufficient and the neuron is active $(y = 1)$. Otherwise the neuron is not active $(y = 0)$. This approach is the basic of learning process. Basically, there are two learning approaches for artificial neural networks. These approaches are "supervised" and "unsupervised" learning methodologies [14]. In literature some sources indicates that there are three approaches and they are supervised, unsupervised, reinforcement learning. In this study supervised approach is used.

For learning process, weights of connections are very important and these weights are signs of learning achievement. Connection weights adopts themselves according to inputs that exceeding threshold value of neuron. For example; a neuron have five input and these inputs are as follows:

$$i_1 = 6, i_2 = 10, i_3 = 14, i_4 = 4, i_5 = 16$$

Initial weights of neuron are as follows:

$$w_1 = 0.3, w_2 = 0.3, w_3 = 0.3, w_4 = 0.3, w_5 = 0.3$$

For the next iteration inputs are as follows:

$$i_1 = 7, i_2 = 17, i_3 = 5, i_4 = 8, i_5 = 13$$

If both of the inputs exceed threshold value after activation function, weights of neuron adapts itself to new situation and neuron success learning process. Basically, this example describes learning.

4 Proposed System And Test Results

Until this section, it is described that secure communication and intrusion detection for wireless sensor networks and artificial neural networks. In this study, artificial neural network is chosen as a solution to implement an intrusion detection system for wireless sensor networks because of it some features. These features are flexibility, having fault tolerance and developing ability.

In this study, supervised learning is accepted as learning technique and logistic function is accepted as activation function. In order to achieve supervised learning, KDD'99 Cup data set is used. Proposed process steps, used approaches and got test results are described in this section.

4.1 Data Set

KDD'99 Cup data set is adopted for this study because it is widely used intrusion detection data set and so facility of comparison this study' s results with different studies is achieved. KDD'99 Cup data set is created by extracting some features (ip number, port number, initial date) from DARPA 98 and it has about 4.900.000 data vector. KDD' 99 Cup data set includes 80 % attack and 20% normal data. Each connection vector has 41 features and is labeled either normal or attack. These 41 features and attack label can be shown in Table 1 [15][16].

	Feature		Feature
1	duration	22	is_guest_login
2	protocol_type	23	count
3	service	24	srv_count
4	flag	25	serror_rate
5	src_bytes	26	srv _serror _rate
6	dst_bytes	27	rerror _rate
7	land	28	srv _rerror _rate
8	wrong_fragment	29	same _srv _rate
9	urgent	30	diff _srv _rate
10	hot	31	srv _diff _host _rate
11	num _failed _logins	32	dst _host _count
12	logged _in	33	dst _host _srv _count
13	num _compromised	34	dst _host _same _srv
14	root_shell	35	dst_host _diff _srv
15	su_attemted	36	dst _host _same _src_port
16	num _root	37	dst _host _srv _diff_host
17	num _file _creations	38	dst _host _serror
18	num _shells	39	dst _host _srv _serror
19	num_access _file	40	dst _host _rerror
20	num _outbound _cmds	41	dst _host _srv _rerror
21	is _host _login	42	attack _type

Table 1: KDD'99 Cup Data Set Features

KDD 99 Cup data set has four group main attack types. These groups can be shown as follows:

- **Denial of Service DoS:** This attack type aims that making busy a computer system or a network so system can not answer real requests.
- **Remote to Local Attack (R2L):** R2L occurs when any attacker has user ability in a network, nevertheless he/she is not a user.
- **Probing Attack:**This attack type is used to gain information from any computer system so attacker aims that find vulnerabilities of the system.
- **User to Root Attack (U2R):** U2R occurs any attacker (internal attacker or external attacker who gets user account by phishing, sniffing password e.g.) who has normal user account get admin authorization.

ISBN: 978-1-61804-307-8

In this study 10 percent of KDD 99 Cup data set **"kddcup.data_10_percent.gz"** is used and it has 494.000 data connection vector.Attack types, number of samples and attack groups of used data set are shown in Table 2.

Attack	Example	Category	Normalization
Smurf	280790	DOS	1
Neptune	107201	DOS	2
Back	2203	DOS	3
Teardrop	979	DOS	4
Pod	264	DOS	5
Land	21	DOS	6
Normal	97277	NORMAL	7
Satan	1589	PROBE	8
IpSweep	1247	PROBE	9
PortSweep	1040	PROBE	10
Nmap	231	PROBE	11
WarezClient	1020	R2L	12
Guess_Password	53	R2L	13
WarezMaster	20	R2L	14
Imap	12	R2L	15
FTP _Write	8	R2L	16
Multihop	7	R2L	17
Phf	4	R2L	18
Spy	2	R2L	19
Buffer _Overflow	30	U2R	20
RootKit	10	U2R	21
LoadModule	9	U2R	22
Perl	2	U2R	23

Table 2: "kddcup.data_10_percent.gz" Attack Types And Normalization Values

4.2 Feature Selection And Pre-processing

Feature selection has an important role to achieve a successful system. In KDD, some features does not have any effective role to classify the outputs even some of them cause additional errors. Feature selection is a big research area and there are different approaches to achieve it. According to literature, feature selection problem solving approaches are grouped as filter, wrapper and hybrid (combination of filter and wrapper) [17][16].

Some feature selection researches are examined and Shirazi's is selected In order to adopt to this study, so 41 features of KDD is decreased to 22 type. It is selected due to it' s test results are successful and well.

Selected Features																					
1	2	3	4	5	6	10	12	14	17	22	23	24	27	29	30	32	33	35	36	37	41

Table 3: Number Of Selected Features

In [17], suitable initial features of each data vectors are selected by Memetic Algorithm (MA). It is described that MA helps system by escaping from local minimum and coming closer to global optimum.The list of selected features can be shown in Table.3

It is described above that KDD has 41 features and three of them is string. In order to use these three string features and to achieve normalization, they should be convert to numeric state. Service feature is normalized as in Table 4, protocol feature is normalized as in Table 5, flag feature is normalized as in Table 6 and labels are normalized as in Table 2.

Service Feature Normalization					
Service	Num.	Service	Num.	Service	Num.
AUTH	1	irc	22	printer	43
BGP	2	imap4	23	private	44
courier	3	iso_tsap	24	red_i	45
csnet_ns	4	klogin	25	remote_job	46
ctf	5	kshell	26	rje	47
daytime	6	ldap	27	shell	48
discard	7	link	28	smtp	49
domain	8	login	29	sql_net	50
domain_u	9	mtp	30	ssh	51
echo	10	name	31	sunrpc	52
eco_i	11	netbios_dgm	32	supdup	53
ecr_i	12	netbios_ns	33	systat	54
efs	13	netbios_ssn	34	tftp_u	55
exec	14	netstat	35	telnet	56
finger	15	nnsp	36	tim_i	57
ftp	16	nntp	37	time	58
ftp_data	17	ntp _u	38	urh _i	59
gopher	18	other	39	urp _i	60
hostnames	19	pm _dump	40	uucp	61
http	20	pop _2	41	whois	62
http_443	21	pop_3	42	vmnet	63

Table 4: Normalization Of Service Feature

Protocol Feature	
TCP	1
UDP	2
ICMP	3

Table 5: Normalization of Prrotocol Feature

Flag Features			
Flag	Num.	Flag	Num
OTH	1	S1	7
REJ	2	S2	8
RSTO	3	S3	9
RSTOS0	4	SF	10
RSTR	5	SH	11
S0	6		

Table 6: Normalization of Flag Feature

ISBN: 978-1-61804-307-8

After all string features are converted to numeric value, all data values of KDD are transferred into fall between (0,1). For normalization, if the formula (2) which is shown below is used, all values will be positive and so logistic function will be use as activation function. Normalization formula can be shown as follow:

$$V_N = 0.8 \times [\frac{V_r - V_{min}}{V_{max} - Vmin}] + 0.1 \quad (2)$$

- V_N means data which is normalized.
- V_{max} means data which has smallest value.
- V_{min} means data which has biggest value.

4.3 Proposed Flowchart

In this study, "kddcup.data_10_percent.gz" is used to train and test artificial neural network. Neural network training and testing processes achieved by MATLAB nntool. Network is created with this properties:

- **Number of Input and Output Layers:**22 / 1.
- **Network Type:**Feed Forward Backprop.
- **Training Funtion:** TRAINLM
- **Adaption Learning Function:** LEARNGDM
- **Performance Funtion:** MSE.
- **Number Of Layers:** 2.
- **Number Of Neuron:** 20.
- **Transfer Function:** LOGSIG.
- **Epochs:** 100.
- **Min_Grad:** 1e-010.

Study can be divided into as three groups. These groups are feature selection and pre-processing, artificial neural network training, artificial neural network testing. All groups have some steps particularly.

- **Group 1 Feature Selection And Pre-processing**
  - Step 1: Get "kddcup.data_10_percent.gz".
  - Step 2: Divide KDD as %85 training, %15 testing
  - Step 3: Achieve feature selection
  - Step 4: Achieve normalization by formula (2).
- **Group 2 Artificial Neural Network Training**
  - Step 4: Create network by MATLAB nntool.
  - Step 5: Train network by %85 training subset.
- **Group 3 Artificial Neural Network Testing**
  - Step 6: Test network by %15 testing subset.

4.4 Test Results

Network is tested by 67500 data connection vector and performance of network comes true as 0.8488. Formula which determining success rate is shown as follow.

$$DedectionRate = \frac{Numberofdetected Attacks}{Numberofattacks} \times 100\%$$

$$FalsePositive = \frac{Misclassifiedconnections}{Numberofnormalconnection} \times 100\%$$

$$Accuracy = \frac{Correctclassifiedconnections}{Numberofconnections} \times 100\%$$

Test results are shown in Table 7 detailed. It cen be seen clearly that attacks having fewer training samples has low success rate. Generally, network has 75% success rate. This rate can be change by feature selection, number of layers, number of neourons e.g.

Attacks	Sample	Det. Rt	Success	False P.
Smurf	12553	100%	100%	0%
Neptune	1250	99.76%	91.36%	0.24%
Back	100	100%	98%	0%
Teardrop	100	100%	100%	0%
Pod	50	100%	100%	0 %
Land	10	80%	70%	20%
Normal	52821	92.41%	92.41%	7.39%
Satan	100	97%	97%	3%
IpSweep	100	91%	100%	9%
PortSweep	100	97%	97%	3%
Nmap	100	99%	88%	1%
WarezClient	100	89%	91%	11%
Guess_Passwrd	10	90%	90%	10%
WarezMaster	10	90%	80%	10%
Imap	10	80%	70%	20%
FTP _Write	4	75%	0%	25%
Multihop	4	75%	0%	25%
Phf	2	100%	100%	0%
Spy	2	100%	0%	0%
Buffer _Overflow	10	50%	50%	50%
RootKit	4	25%	25 %	75%
LoadModule	4	25%	25%	75%
Perl	2	50%	50%	50%
Total	67500	91.64%	89.17%	8.36%

Table 7: Test Results

5 Conclusion And Future Work

In this study it is aimed that proposing an artificial neural network based intrusion detection system for wireless sensor networks. Test results are success-

ISBN: 978-1-61804-307-8

ful but only Buffer _Overflow, RootKit, LoadModule, Perl attacks have low rates because they have fever training data connection vector.As future work in order to get more success detection rate for other attack types, new neural networks will be implemented by different properties. To obtain flexible structure and easy development for IDS to secure communication in WSN, artificial neural network java classes of the Joone will be used. For minor measurements, Joone upgraded Oracle Sun Spot development kit will let this study more effective using different weights of neurons from neural network. For wide measurements, SecWiseNet Simulator is functional according to its free java programming and ready cyber attacks features. Both SunSpot Development Kit and SecWiseNet Simulation will show the proposed system performance considering energy consumption, security skill of communication, fault tolerance and transmission time.

References:

[1] K. K. Khedo, R. Perseedoss, A. Mungur *et al.*, "A wireless sensor network air pollution monitoring system," *arXiv preprint arXiv:1005.1737*, 2010.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] A. Abduvaliyev, A.-S. K. Pathan, J. Zhou, R. Roman, and W.-C. Wong, "On the vital areas of intrusion detection systems in wireless sensor networks." *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1223–1237, 2013.

[4] O. K. Sahingoz, "Multi-level dynamic key management for scalable wireless sensor networks with uav," in *Ubiquitous Information Technologies and Applications*, ser. Lecture Notes in Electrical Engineering, vol. 214. Springer Netherlands, 2013, pp. 11–19.

[5] C. Turguner and O. K. Sahingoz, "Secure communication in wireless sensornetworks," in *International Conference on Modelling, Simulation and Applied Optimization (ICMSAO), 2015*. IEEE, 2015.

[6] L. B. Jivanadham, A. M. Islam, N. Mansoor, and S. Baharun, "A secured dynamic cluster-based wireless sensor network," in *Computational Intelligence, Communication Systems and Networks (CICSyN), 2012 Fourth International Conference on*. IEEE, 2012, pp. 223–228.

[7] C. Turguner, "Secure fault tolerance mechanism of wireless ad-hoc networks with mobile agents," in *Signal Processing and Communications Applications Conference (SIU), 2014 22nd*. IEEE, 2014, pp. 1620–1623.

[8] O. Can, "Mobile agent based intrusion detection system," in *Signal Processing and Communications Applications Conference (SIU), 2014 22nd*. IEEE, 2014, pp. 1363–1366.

[9] H. Moosavi and F. Bui, "A game-theoretic framework for robust optimal intrusion detection in wireless sensor networks," 2014.

[10] O. Can and O. Sahingoz, "A survey of intrusion detection systems in wireless sensor networks," in *6th International Conference On Modeling, Simulation And Applied Optimization*, May 2015, p. Accepted.

[11] E. Karapistoli and A. A. Economides, "Anomaly detection and localization in uwb wireless sensor networks," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, Sept 2013, pp. 2326–2330.

[12] S. Roy, S. Nag, I. K. Maitra, and S. K. Bandyopadhyay, "International journal of advanced research in computer science and software engineering," *International Journal*, vol. 3, no. 6, 2013.

[13] D. Kriesel, "A brief introduction to neural networks," *Retrieved August*, vol. 15, p. 2011, 2007.

[14] S.-J. Horng, M.-Y. Su, Y.-H. Chen, T.-W. Kao, R.-J. Chen, J.-L. Lai, and C. D. Perkasa, "A novel intrusion detection system based on hierarchical clustering and support vector machines," *Expert systems with Applications*, vol. 38, no. 1, pp. 306–313, 2011.

[15] M. Tavallaee, E. Bagheri, W. Lu, and A.-A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications 2009*, 2009.

[16] T. Eldos, M. K. Siddiqui, and A. Kanan, "On the kdd'99 dataset: Statistical analysis for feature selection," *Journal of Data Mining and Knowledge Discovery, ISSN*, pp. 2229–6662, 2012.

[17] H. Shirazi, A. Namadchian, and A. khalili Tehrani, “A combined anomaly base intrusion detection using memetic algorithm and bayesian networks,” *differences*, vol. 16, p. 17, 2012.