Journal of Information Assurance and Security.
ISSN 1554-1010 Volume 6 (2011) pp. 469–477

Misfeasor Classification and Detection Models Using Machine Learning Techniques

Nesrine Sameh[1], Neamat El Gayar [1, 2], Nashwa Abdelbaki[1]

[1] Nile University, Center for Informatics Science,
Extension of 26th of July Corridor, Giza, Egypt
nesrine.said@nileu.edu.eg
nabdelbaki@nileuniversity.edu.eg

[2] Cairo University, Faculty of Computers and Information,
12613 Giza, Egypt
elgayar.neamat@gmail.com

***Abstract*: Misfeasors (or insiders) are considered among the most difficult intruders to detect due to their knowledge and authorization within the organization. Machine learning techniques have been widely used for intrusion detection but only little work has addressed the use of machine learning for detecting and classifying different types of insiders. The aim of this study is to exploit different recognition models for misfeasors detection by adding the Mac address as a feature in classification. Three different recognition models (a Rule Based Model, a Hierarchical Classification Model and a Composite Feature Model) are proposed. The models differ mainly in the amount of prior knowledge required for the problem and hence how training data is used to construct the models. The Rule Based Model uses explicit domain classification rules given by expert to detect insiders. The Hierarchical Classification Model uses some domain specific knowledge to manufacture the training data in order to construct the hierarchy in the recognition model. The Composite Feature Model on the other hand attempts to discover classification rules directly from the training data without any prior knowledge. All three proposed classification models are tested on two benchmark data sets and are evaluated using different performance measures. Results for the different models are presented and compared for several classification techniques. Experiments reveal that using machine learning at different levels in the proposed models yield a good approximation for the classification rules for the problem of misfeasor detection**

***Keywords*:** Misfeasors, Masqueraders, Mac address, Machine Learning, Classification, Intrusion Detection

I. Introduction

Intrusion detection is the process of passively identifying and detecting attempts of intrusions [5]. Packet Filtering Firewalls apply a set of rules to each incoming and outgoing IP packet to actively prevent the occurrences of intrusions [13]. Intrusion Detection Systems (IDS), which are usually used in network security, can be evaded by two types of intruders: misfeasors and masqueraders. Misfeasors (also known as insiders) are the most difficult to detect due to their knowledge and legitimate authorizations within an organization. Masqueraders (known as outsiders) disguise as legitimate employees and threat the security of the organization. There are two detection approaches used by IDS, anomaly based detection and signature based detection. In the anomaly based detection approach the system's normal patterns are learned and an alert is raised when deviations are detected. In contrast, signature based IDS learn the system's normal patterns as well as known patterns of attacks and raise an alert when attack patterns are detected. Moreover, there are two types of IDS, host based IDS which monitor the characteristics and events of a single host and network based IDS which monitor network traffic [15].

Recently, various research works have applied the state of the art machine learning techniques for intrusion detection. Examples of such are [17], [3], [11], [14] and [12]. Machine learning enables systems to analyze complex suspicious patterns and have better detection insight of unknown attacks [7]. However; only few attempts have been made using machine learning to detect misfeasors and particularly for network-based intrusion detection. [6] use an approach based on k nearest neighbor outlier detection for detecting anomalies. This approach focuses on misfeasors' detection at the operating system level (host-based) by tracking system calls using three features namely, 'ngrams of system call names' , 'frequency counts of system call names' and 'parameter and return code information' . The misfeasors' attacks detected are 'privilege escalation', ' change file extension', 'removable media', encipher/ decipher' browse malware', 'unusual search' and 'export via email'. In [8] supervised algorithms such as Naïve Bayes Classifier, Decision Tree Classifier and Support Vector Machines are applied for anomaly detection. The scope of the research focuses on detecting misfeasors in database systems by profiling users' database access patterns and observing exactly what they access to detect their anomalies. The approach is host based. Examples of misfeasors' attacks detected are 'masquerade', 'SQL injection' and 'Data harvesting'. [18] use a Support Vector Machine classifier for anomaly detection. Misfeasors are detected by building users' profiles and assigning numerical values to elements within

*Table 1.*Common features for detecting attacks

Feature	Feature Name	Description
X1	Hours	The starting hour of the session
X2	Minutes	The starting minutes of the session
X3	Duration	The length (in seconds) of the connection
X4	Service	The service name accessed by the connection
X5	Source port	The source port number of the session
X6	Destination port	The destination port number of the session
X7	Source IP address	the source IP address of the session
X8	Destination address	The destination IP address of the session

these profiles representing the frequency of users performing certain system calls and actions within applications. After the profile is established deviations are flagged and considered suspicious. Again, the approach is host based.

In Computer Security, the IP address is a logical address used to identify systems on a LAN and on the internet in which it operates at layer 3 in the OSI model. On the other hand the Mac address is a physical hardware address used only to deliver frames on a LAN and is not routed through the internet as it operates at layer 2 in the OSI model. The IP and the Mac addresses are assigned to each host in order to identify the host uniquely [4]. However, both can be spoofed. Nevertheless, the Mac address is considered reliable in identifying a host, because in organizations registered Mac addresses are known by administrators and since they are not routed through the internet, they are hard to be evaded by misfeasors (i.e. insiders pretending to be outsiders) or predicted by masqueraders (i.e. outsiders pretending to be insiders). Therefore, the Mac address is reliable in detecting misfeasors disguising themselves and spoofing their IP addresses [2] particularly with the deployment of the IDS itself at the boundary of the internal network of the organization on the internal interface of a firewall [16].

This paper proposes a network - signature based approach for detecting the misfeasor activity by adding the feature of the Mac address. In particular, three different recognition models (a Rule Based Model, a Hierarchical Classification Model and a Composite Feature Model) are proposed to detect the misfeasors as well as the masqueraders. The models differ mainly in the amount of prior knowledge required of the problem and hence how training data is used to construct the models. Experiments are conducted using the DARPA 1998 and the DARPA 1999 datasets [9] [10]. Different performance measures are used for comparison. Moreover different classifiers are exploited (Bayes Naïve, J48, Simple Cart, Attribute Selected Classifier, BF Tree and NB Tree) to investigate how the choice of the component classifiers affects the overall recognition models.

The rest of this paper is organized as follows: Section 2 discusses the features and attacks issued by misfeasors. In section 3, three proposed models for classifying and detecting misfeasors using machine learning are described. The dataset and the experimental setup are described in Section 4. In section 5 results are presented and discussed. Finally the paper is concluded in section 6.

II. Misfeasors' Features and Attacks

In this section, we discuss common features used in detecting attacks and the effect of adding the proposed Mac address feature in detecting more types of attacks mostly issued by misfeasors.

A. Common Features for Intrusion Detection and Associated Attacks

Generally, information contained in a network packet routed through the internet is used to detect attacks. For example, 'Hours', 'Minutes', 'Duration', 'Service Name', 'Source Port', 'Destination Port', 'Source IP Address' and 'Destination IP Address' as described in Table 1are common features that are useful for distinguishing various attacks. Following types of attacks are typically detected when using features described in Table 1[9] [10]:

- **C1**, (Guess), is guessing numerous passwords to log into a target computer remotely.
- **C2**, (Port-Scan), is determining which services on a target computer are active.
- **C3,** (Phf), a suspicious UNIX command line on a web server.
- **C4,** (Rlogin), is remotely logging to a target computer without a password.
- **C5,** (Rsh), is executing a command on a target computer without a password.
- **C6,** (Rcp), is remotely copying a file to/from a target computer without a password.
- **C7,** (Dos), is denying access of legitimate users to computer system resources.
- **C8,** (U2r), is a user to root attack in which an attacker gains root access to a computer.
- **C9,** (R2l), is an attack where an attacker remotely gains unauthorized local access to a computer.
- **C10,** (Probe), is scanning a network of computers to gather information or find vulnerabilities.
- **C11**, (*Data)*, is non authorized action done on a computer including deleting, copying or altering data.
- **C12**, (Data-U2r), is gaining root access to a computer and performing non authorized action to data.
- **C13,** (Data-R2l), is gaining local access to a computer and performing non authorized action to data.

It is important to note at this point that the features described in Table 1 fail to identify the users' exact hardware location and consequently cannot be used to detect misfeasors and are vulnerable to IP spoofing attacks.

B. Enhancing Features Using the Mac Address

We consider authorized users (insiders) spoofing their IP addresses without yet launching noticeable attacks as stealthy misfeasors. We propose adding the feature of the MAC address to detect IP spoofing attacks. In this work, we use the DARPA1998 [9] and the DARPA 1999 [10] datasets, in which IP addresses are listed within the dataset representing inside and outside hosts. We assume a list of Mac addresses and add them to the datasets in which each Mac address is correlated with an IP address, and then we blend different combinations of inside and outside IP and Mac addresses. In addition to the attacks listed above, using the Mac address

results in identifying more attacks. The following are the resulting attacks after adding the Mac address using the DARPA 1998 dataset [9]:

- **C14,** (Masquerader Guess), a masquerader guessing numerous passwords to log into a target.
- **C15,** (Misfeasor Guess), a misfeasor guessing numerous passwords to log into a target
- **C16,** (Masquerader Port-scan), masquerader determining which services on a target machine are active.
- **C17,** (Misfeasor Port-scan), a misfeasor determining which services on a target machine are active
- **C18,** (Masquerader Phf), a masquerader running a suspicious Unix command line on a web server.
- **C19,** (Misfeasor Phf), a misfeasor running a suspicious Unix command line on a web server.
- **C20,** (Masquerader Rlogin), a masquerader logging in to a target without a password.
- **C21,** (Misfeasor Rlogin), a misfeasor logging in to a target without a password.
- **C22,** (Masquerader Rsh), masquerader executing a command on the target machine without a password.
- **C23,** (Misfeasor Rsh), a misfeasor executing a command on the target machine without a password.
- **C24,** (Masquerader Rcp), a masquerader remotely copying a file to or from a target without a password.
- **C25,** (Misfeasor Rcp), a misfeasor remotely copying a file to or from a target without a password.
- In addition, we present the resulting attacks after adding the feature of the Mac address to the DARPA 1999 dataset (MIT Lincoln Labs 1999)as follows:
- **C26,** (Masquerader Dos), a masquerader issuing a denial of service attack .
- **C27,** (Misfeasor Dos), a misfeasor issuing a denial of service attack .
- **C28,** (Masquerader U2r), a masquerader who gains root access to a computer.
- **C29,** (Misefasor U2r), a misfeasor who gains root access to a computer.
- **C30,** (Masquerader R2l), a masquerader who remotely gains unauthorized local access to a computer.
- **C31,** (Misfeasor R2l), a misfeasor who remotely gains unauthorized local access to a computer.
- **C32,** (Masquerader Probe), masquerader scanning a network of computers looking for vulnerabilities.
- **C33,** (Misfeasor Probe), a misfeasor who scans a network of computers looking for vulnerabilities.
- **C34,** (Masquerader Data), a masquerader who performs non authorized action to data on a computer .
- **C35,** (Misfeasor Data), a misfeasor who performs non authorized action to data on a computer.
- **C36,** (Masquerader Data-U2r), masquerader gaining root access and performing unauthorized action
- **C37,** (Misfeasor Data-U2r), misfeasor gaining root access and performing unauthorized action to data
- **C38,** (Masquerader Data-R2l), masquerader gaining local access and performing unauthorized action.
- **C39,** (Misfeasor Data-R2l), misfeasor gaining local access and performing unauthorized action to data.
- **C40,** (Masquerader), unauthorized individual (outsider) spoofing the IP address.
- **C41,** (Misfeasor), authorized individual (insider) spoofing the IP address pretending to be from outside.

III. Misfeasors' Attacks

We propose three recognition models for classification and detection of misfeasors. The suggested models use the Mac address as an augmented feature to the common features usually used to identify other attacks. In the following sections we describe the proposed models in details.

A. The Rule Based Model (Model A)

In the rule based model (we will refer to it as Model A) classification proceeds into two separate stages. This is denoted in Figure 1by the two classifiers D_1^A and D_2^A.

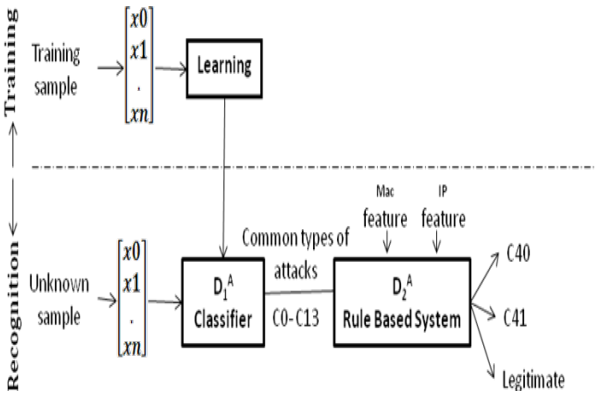

Figure 1. The Rule Based System

The first stage is concerned with identifying common type of attacks, as described in section II.A, and in distinguishing them from normal patterns while the second stage is a rule based system that applies simple rules related to the Mac and IP addresses to issue a signal whether this attack is a misfeasor, a masquerade or a legitimate user.

D_1^A is trained using a training data T_1^A which consists of a set of labeled patterns with features [x0, x1: xn], where x0 is the augmented feature of the Mac address and x1:xn represents the features as described in Table 1. The labels denote the type of attacks as described in section II.A. If the label of the normal patterns is C0 then the label *(labeli*) of any pattern *ti* in the training data can take values [C0:C13].

$D_2^A$is a rule based classifier that identifies the type of attacker using the Mac and IP addresses. D_2^A does not require training as it models explicit rules that map an input attack to one of the labels [C40, C41, Legitimate] depending on the following rule:

If the source IP address of a user is from outside the organization (different LAN) and the Mac address is a registered one, the pattern is labeled as 'Misfeasor' and if the source IP address of a user is from inside the organization (same LAN) and the Mac address is not a registered one, the pattern is labeled as 'Masquerader', otherwise the pattern is labeled as 'legitimate'.

Hence, D_1^A acts as an IDS classifying normal patterns and different types of attacks and D_2^A acts as a packet filtering firewall relying on specific rules to determine the occurrences

of IP spoofing attacks and thus identifying the type of intruder at hand. Therefore, the two classification stages when combined identify misfeasors as well as the type of their attacks.

Model A is built on the hypothesis that rules for D_2^A are known before hand or are at least easy to deduce and simple to implement. D_1^A on the other hand can be learned from data and can be implemented using any machine learning technique.

B. The Hierarchical Classification Model (Model B)

The second proposed model (Model B) is a hierarchical classification model where the classification task is decomposed into hierarchies. As shown in Figure2, the first classification level (classifier D_1^B) classifies a collection of patterns into two subsequent classification levels with which further classifiers (D_2^B and D_3^B) are concerned. Classifier D_1^B classifies each pattern in the training data T_1^B described by features [x0, x1: xn], where again x0 is the proposed added Mac address feature, to belong to one of the labels ∈ [Attack, Normal]. The Attack label represents patterns of attacks regardless of the type of attack. The Normal label represents patterns of normal behaviors and includes the two types of attacks [C40, C41] because we consider these types of attacks unnoticeable at this stage.

In the next level the 'Attack' category is further classified by classifier D_2^B. The training data T_2^B is composed of a collection of attack patterns described by features [x0: xn]. Each attack pattern is classified into the different types of attacks [C1: C39] excluding the misfeasor/masquerader [C40, C41].

On the other hand the 'Normal' category is further classified by classifier D_3^B. The training data of this classifier T_3^B is composed of a set of normal patterns and patterns of the attacks [C40, C41]. Each pattern is classified to represent a misfeasor, a masquerader or a legitimate user.

Model B needs some prior knowledge to manufacture the training data and decompose the problem into a hierarchy for classification. However, classifiers D_1^B, D_2^B and D_3^B are trained from examples of data collected and do not rely on explicit expert domain rules.

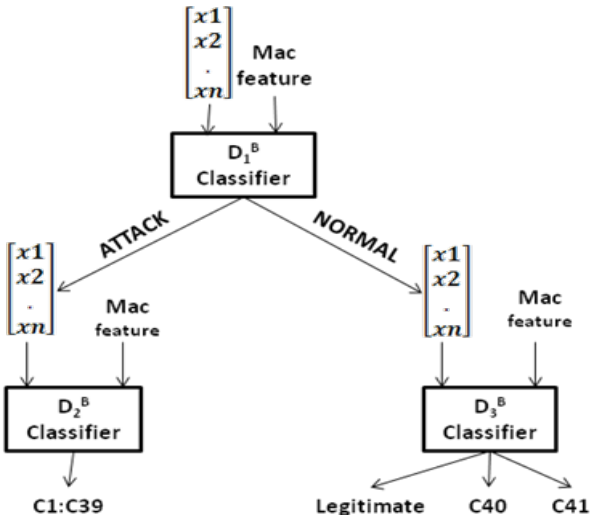

Figure2. The Hierarchical Classification Model

C. The Composite Feature Model (Model C)

The third proposed model (Model C) is composed of a single classification process indicated by classifier D^C as shown in Figure 3. The training data for D^C is composed of a collection of patterns described by features [x0, x1:xn]. Each pattern is labeled to belong to classes [C0:C41]; as described in section II.A where C0 represents the normal patterns.

Model C uses minimal amount of prior information of the problem in contrast to Model A and Model B described previously. More particularly the Composite Feature Model attempts to discover classification rules directly from the training data without any prior knowledge and using all features at hand in one step. We consider this model to be the most generic model presented in this work so it can be easily applied to other data sets.

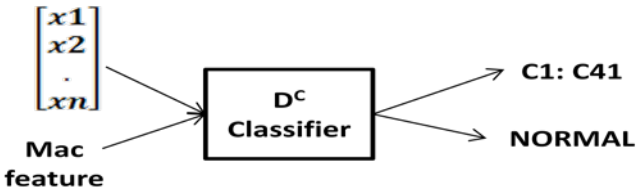

Figure 3.The Composite Feature Model

IV. DATA AND EXPERIMENTS

We use the 1998 DARPA evaluation program sample dataset [9] and the 1999 DARPA evaluation program dataset [10] due to their relevance to our scenario. The data is collected using TCP-dump sniffer. Each line corresponds to an individual TCP/IP connection between two workstations, one at the inside interface of a router and the other at the outside. Data processing is done on the datasets such that irrelevant features including '*time in seconds*' and '*date*' are removed. Table 2 and Table 3 show a better insight of the structure of the DARPA 1998[9] and the DARPA1999 [10] datasets after adding the MAC address as a feature in classification. It is important to note that the types of attacks increases when adding the feature of the MAC address and thus the patterns of attacks significantly increases from 72 to 484 and from 1399 to 4798 patterns of training data for the DARPA1998 [9] (Table 2)and DARPA1999 [10] (Table 3) datasets respectively. We test our proposed recognition models (Models A, B and C) using a wide variety of classifiers available in WEKA (Waikato Environment for Knowledge Analysis) data mining toolbox [19]. Here we present the results of the best classifiers which are Bayes Naïve, J48, Simple Cart, Attribute Selected Classifier, BF Tree and NB Tree when used to implement D_1^A, D_1^B, D_2^B, D_3^B and D^C. Results show the performance for 10 fold cross validation on the same dataset.

Note that accuracy alone is not indicative in our application. High accuracy does not necessary mean all intrusions are detected .Therefore, considering *false negatives* (FN) which indicate the failure of detecting classes such as misfeasors and *false positives* (FP) where false alarms rates are calculated can be also useful together with accuracy as a measure of performance. We use the weighted average False negative rate (WFN) and the weighted average False positive rate (WFP) as proposed in [19].

Table 2. Structure of DARPA1998 dataset after adding MAC address

Item	Number of occurrences in DARPA1998 dataset before adding the MAC address as a feature	Number of occurrences in DARPA1998 dataset after adding the MAC address as a feature
Features	10	11
Patterns	412	824
Normal patterns	340	340
Attacks patterns	72	484
Types of attacks	6	20
Guess	7	7
Port-Scan	56	56
Phf	2	2
Rlogin	1	1
RSh	4	4
RCp	2	2
Masquerader Guess		4
Misfeasor Guess		3
Masquerader Port-scan		27
Misfeasor Port-scan		29
Masquerader Phf		1
Misfeasor Phf		1
Masquerader RLogin		0
Misfeasor Rlogin		1
Masquerader RSh		2
Misfeasor RSh		2
Masquerader RCp		1
Misfeasor RCp		1
Masquerader		165
Misfeasor		175

Table 3. Structure of DARPA1999 dataset after adding MAC address

Item	Number of occurrences in DARPA1999 dataset before adding the MAC address as a feature	Number of occurrences in DARPA1999 dataset after adding the MAC address as a feature
Features	6	7
Patterns	3399	6798
Normal patterns	2000	2000
Attacks patterns	1399	4798
Types of attacks	7	23
Dos	177	177
U2r	74	74
R2l	425	425
Probe	686	686
Data	10	10
Data-U2r	14	14
Data-R2l	13	13
Masquerader DOS		32
Misfeasor DOS		145
Masquerader U2r		23
Misfeasor U2r		51
Masquerader R2l		41
Misfeasor R2l		384
Masquerader Probe		36
Misfeasor Probe		650
Masquerader Data		1
Misfeasor Data		9
Masquerader Data-U2r		2
Misfeasor Data-U2r		12
Masquerader Data-R2l		0
Misfeasor Data-R2l		13
Masquerader		1000
Misfeasor		1000

Table 4. Summary of the training data for models A, B and C

	DARPA98		DARPA99	
Training Data	**Features**	**Labels**	**Features**	**Labels**
T_1^A	[X0, X1:X8]	[C0, C1: C6]	[X0, X1, X2, X3, X7, X8]	DOS, U2R, R2L, PROBE, DATA, DATA-U2R, DATA-R2L and normal.
T_1^B	[X0, X1:X8]	NORMAL And ATTACK.	[X0, X1, X2, X3, X7, X8]	NORMAL and ATTACK.
T_2^B	[X0, X1:X8]	[C1: C6, C14 : C25]	[X0, X1, X2, X3, X7, X8]	[C7: C13, C26 : C39]
T_3^B	[X0, X1:X8]	[C0, C40, C41]	[X0, X1, X2, X3, X7, X8]	[C0, C40, C41]
T^C	[X0, X1:X8]	[C0 ,C1: C6, C14 : C25, C40, C41]	[X0, X1, X2, X3, X7, X8]	[C0 ,C7: C13, C26 : C41]

V. RESULTS

Using the DARPA1998 dataset [9], the six classifiers are investigated on the three proposed models using the weighted average false negative rate (WFN), weighted average false positive rate (WFP) and Accuracy. WFN results are illustrated in Figure 4.

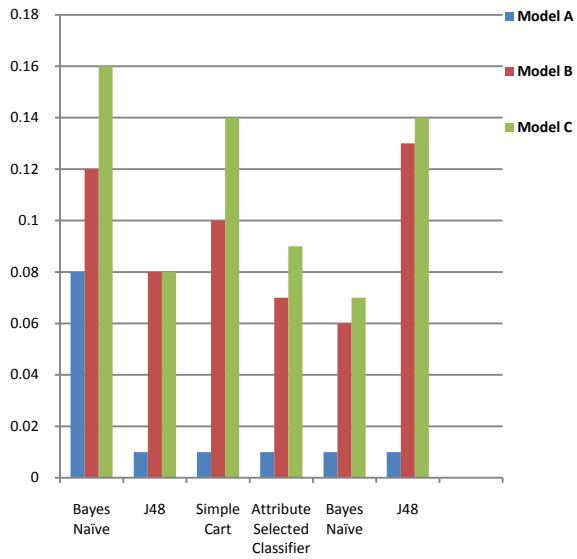

Figure 4. WFN using six classifiers on the three models using DARPA1998 dataset

It shows that the Rule Based Model (Model A), which relies on expert knowledge, always produces the lowest WFN when using the DARPA1998 dataset [9] and thus yields better performance. Results of the WFP are shown in Figure 5. It shows that the Rule Based Model (Model A), does not necessarily outperform the other models. The reason for that is that Model C yields the lowest WFP which means best performance when using the Bayes Naïve classifier. Model B yields the lowest WFP when using Simple Cart, NB Tree and Attribute Selected Classifiers. However Model A results in the lowest WFP when using BF tree. Figure 6 shows results in terms of Accuracy. The Rule Based Model (Model A), which requires expert knowledge, again outperforms the other models. It yields the highest accuracy results and thus best performance.

Considering the DARPA1999 dataset [10], the six classifiers are also investigated on the three proposed models using the weighted average false negative rate (WFN), weighted average false positive rate (WFP) and Accuracy. Results of WFN are shown in Figure 7. It shows that the Rule Based Model (Model A) yields best results in terms of WFN when using J48, Simple Cart and BF Tree. The hierarchical classification model (Model B) yields best results when using Bayes naive, NB tree and Attribute Selected Classifiers. Figure 8 shows the results of WFP using the DARPA1999 dataset [10]. It shows that the Hierarchical Classification Model (Model B), yields best results on all classifiers, especially for the NB Tree and Attribute Selected Classifiers. Moreover, results in terms of accuracy are shown in Figure 9. They show that the Rule Based Model (Model A) yields best results in terms of accuracy when using J48, Simple Cart and BF Tree. The Hierarchical Classification Model (Model B) produces best results in terms of accuracy when using Bayes Naïve, NB Tree and Attribute Selected Classifiers.

Table 5 summarizes the results of Model A, B and C with using different classifier to implement D1A, D1B, D2B, D3B and DC. It also compares the weighted average false negative rate (WFN), weighted average false positive rate (WFP) and accuracy of the three proposed models. Examining Table 5 one can generally come to the conclusion that the Rule Based Model (Model A) outperforms the other models. This is of course an expected result since the model is built using explicit expert classification rules (for D1A). However, the Hierarchical Classification Model (Model B) and the Composite Feature Model (Model C) still yield acceptable performance considering the small amount of data available for training. In addition both models B and C do not rely on explicit domain knowledge with Model C being the most generic model. Currently we are running simulation experiments to compare the performance of the three proposed models on a real network data; where the availability of more training data is expected to enhance the performance of models B and C to more accurately approximate the rule based model.

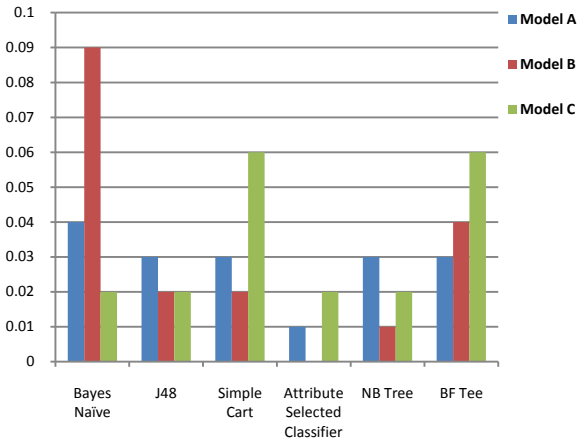

Figure 5. WFP using six classifiers on the three models using DARPA1998 dataset

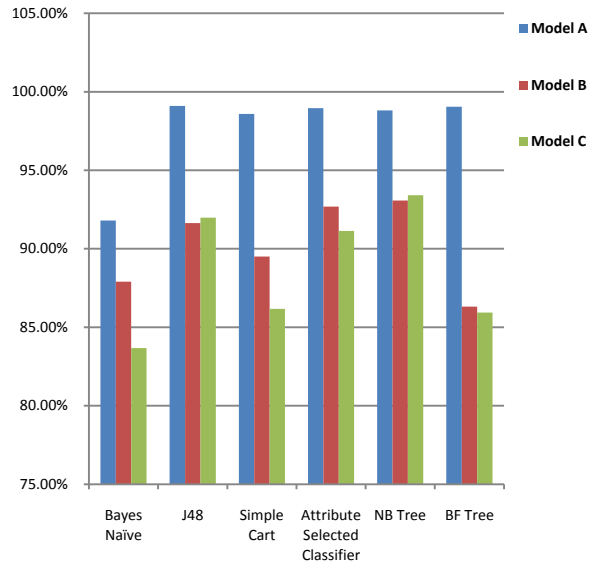

Figure 6. Accuracy using six classifiers on the three models using DARPA1998 dataset

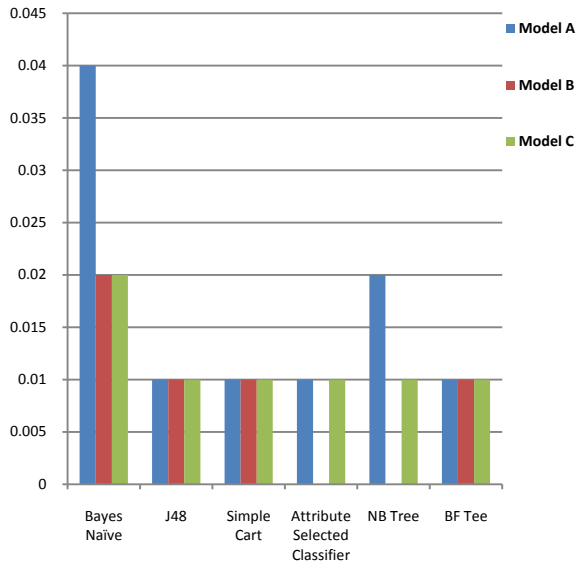

Figure 8. WFP using six classifiers on the three models using DARPA1999 dataset

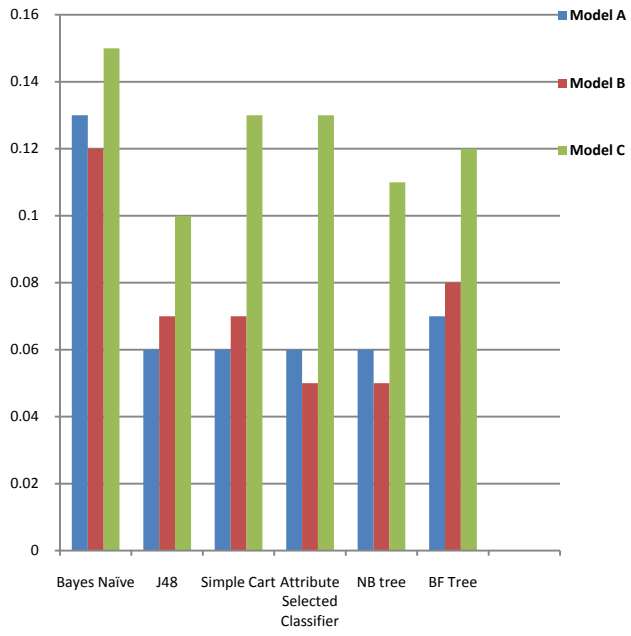

Figure 7. WFN using six classifiers on the three models using DARPA1999 dataset

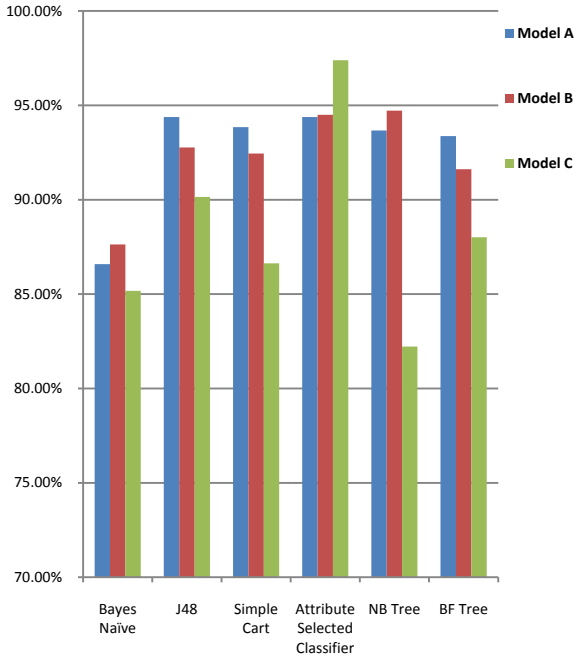

Figure 9. Accuracy using six classifiers on the three models using DARPA1999 dataset

VI. CONCLUSIONS AND FUTURE WORK

Misfeasors have ways of evading others and being stealthy, one of which is spoofing their IP addresses. Commonly, the detection of IP spoofing attacks is carried out by firewalls using packet filtering rules. This study presents a network-signature based approach for detecting some types of misfeasors attacks using machine learning. The Mac address is used as an augmented feature to the original feature sets (i.e. the feature set that is used to detect external attacks) to identify the hardware address of a user and to detect IP spoofing attacks. Three recognition models are proposed and evaluated using different types of classifiers. The models differ mainly in the amount of expert knowledge required of the problem. The *Rule Based Model* uses explicit domain classification rules given by expert to detect insiders. The *Hierarchical Classification Model* uses some domain specific knowledge to manufacture the training data in order to construct the hierarchy in the recognition model. The *Composite Feature Model* on the other hand attempts to discover classification rules directly from the training data without any prior knowledge. We consider this model to be the most generic model presented in this work so it can be easily applied to other data sets. All three proposed classification models are tested on two small benchmark data sets and are evaluated using different performance measures. Experiments reveal that using machine learning at different levels in the proposed models yield a good approximation for the classification rules for the problem of misfeasor detection. Currently we are developing a simulation model to verify the performance of the three proposed models on a real network data.

Table 5. Comparison of the performance of Model A, B and C

		Darpa98			Darpa99		
Models	Models	WFN	WFP	Accuracy	WFN	WFP	Accuracy
Model (A)	Bayes Naïve	0.08	0.04	91.80%	0.13	0.04	86.59%
	J48	0.01	0.03	99.10%	0.06	0.01	94.38%
	Simple Cart	0.01	0.03	98.59%	0.06	0.01	93.85%
	Attribute Selected Classifier	0.01	0.01	98.96%	0.06	0.01	94.38%
	NB Tree	0.01	0.03	98.81%	0.06	0.02	93.67%
	BF Tree	0.01	0.03	99.05%	0.07	0.01	93.37%
Model (B)	Bayes Naïve	0.12	0.09	87.90%	0.12	0.02	87.63%
	J48	0.08	0.02	91.64%	0.07	0.01	92.77%
	Simple Cart	0.10	0.02	89.50%	0.07	0.01	92.45%
	Attribute Selected Classifier	0.07	0.00	92.69%	0.05	0.00	94.50%
	NB Tree	0.06	0.01	93.07%	0.05	0.00	94.72%
	BF Tree	0.13	0.04	86.31%	0.08	0.01	91.62%
Model (C)	Bayes Naïve	0.16	0.02	83.67%	0.15	0.02	85.17%
	J48	0.08	0.02	91.98%	0.10	0.01	90.15%
	Simple Cart	0.14	0.06	86.17%	0.13	0.01	86.63%
	Attribute Selected Classifier	0.09	0.02	91.14%	0.13	0.01	87.39%
	NB Tree	0.07	0.02	93.41%	0.11	0.01	82.22%
	BF Tree	0.14	0.06	85.94%	0.12	0.01	88.01%

REFERENCES

[1] AlGhamdi, Ghazi A. et al., 2008. Modeling Insider User Behavior Using Multi-Entity Bayesian Network. 10th International Command and Control Research and Technology Symposium.

[2] Ali, F, 2007. IP Spoofing. The Internet Protocol Journal, Vol.10, No. 4.

[3] Chen, R., et al., 2009. Using Rough Set and Support Vector Machine for Network Intrusion Detection. International Journal of Network Security & Its Applications (IJNSA). Vol. 1, No 1, pp.01-13.

[4] Cole, E., et al., 2008a. SECURITY 401.1: Security Essentials Bootcamp Style, Networking Concepts. SANS Institute.

[5] Cole, E., et al., 2008b. SECURITY 401.2: Security Essentials Bootcamp Style, Defense in Depth. SANS Institute.

[6] Liu, A., Martin, C., Hetherington, T., and Matzner, S., 2005. A Comparison of System Call Feature Representations for Insider Threat Detection. Proceedings of the 2005 IEEE Workshop on Information Assurance. West Point, United States.

[7] Maloof, Marcus A. (Ed.), 2006. Machine Learning and Data Mining for Computer Security. Springer.

[8] Mathew, S., et al., 2009. A Data Centric Aprroach to Insider Attack Detection in Database Systems.

[9] MIT Lincoln Labs, 1998, sample data for the 1998 DARPA Intrusion Detection Evaluation programme. [Online].

[10] MIT Lincoln Labs, 1999, 1999 DARPA Intrusion Detection Evaluation programme. [Online].

[11] Osareh, A., Shadgar, Bita. 2008. Intrusion Detection in Computer Networks based on Machine Learning Algorithms. International Journal of Computer Science and Network Security (IJCSNS). Vol.8, No.11, pp. 15-23.

[12] Pietraszek, T., Tanner, A., 2005. Data mining and machine learning –Towards reducing false positives in intrusion detection. Information Security Technical Report 10, 169-183. ELSEVIER.

[13] SANS, 2009. SECURITY 502.2: Perimeter Protection In Depth, Firewalls, NIDS and NIPS. SANS Institute.

[14] Sebastiaan, T., 2007. Improving Intrusion Detection Systems through Machine Learning. Technical Report Series no. 07-02, ILK Research Group, Tilburg University.

[15] Stallings, W., 2005. Cryptography and Network Security Principles and Practices, Fourth Edition. Pearson Prentice Hall, Upper Saddle River, New Jersey.

[16] Stallings, W., 2008. Cryptography and Network Security Principles and Practices, Pearson International Edition. Pearson Prentice Hall, Upper Saddle River, New Jersey.

[17] Subbulakshmi, T., et al., 2010, Real time classification and clustering of ids alerts using machine learning algorithms. International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 1, No.1.

[18] Udoeyopv, A., Sheldon, Frederick T., Kirkpatrick, Michael S., 2009. Heuristic Identification and Tracking of Insider Threat Prospectus.

[19] Witten, I.H., Frank, E, 2005. Data Mining: Practical Machine Learning Tools and Techniques, Second Edition. Morgan Kaufmann, San Francisco.

Author Biographies

Nesrine Sameh was born in Cairo, Egypt in 1985. She received her B.Sc. in Computer Engineering from the Arab Academy for Science and Technology Egypt and her M.S.c in Information Security from the Nile University Egypt in 2008 and 2010, respectively. She joined the Nile University in September 2008 as a research assistant in the Data Mining and Machine Learning research team. Nesrine Sameh's research interest merges between Information Security and Machine Learning. She is certified from four SANS courses, which are GIAC Security Essentials (GSEC), GIAC Certified Incident Handler (GCIH), GIAC Certified Firewall Analyst (GCFW) and GIAC Assessing and Auditing Wireless Networks (GAWN) certificates. Nesrine has also been one of the organizers of MCS2010 workshop on Multiple Classifier Systems and ANNPR2010 the 4th IAPR International Workshop on Artificial Neural Networks in Pattern Recognition on Machine Learning and Data Mining in Egypt, 2010.

Neamat El Gayar was born in Alexandria, Egypt in 1966. She received her B.Sc. and her M.Sc. in Computer Engineering from University of Alexandria Egypt in 1989 and 1993, respectively. She obtained her Ph.D. in Computer Science in March 1999 from the Faculty of Engineering, University of Alexandria, Egypt, jointly supervised from the Department of Neural Information Processing, Faculty of Computer Science, University of Ulm, Germany. From September 2000 Dr El Gayar was appointed as an assistant professor and then an associate professor in the Department of Information Technology, Faculty of Computers and Information, Cairo University. She joined the Nile University in September 2008 as an associate professor. Currently she is a visiting scientist at CENPARMI in Montreal, Canada. Dr Neamat's research interests lie mainly in the fields of pattern recognition and machine intelligence, artificial neural networks, fuzzy systems , multiple classifiers, soft computing, data mining and intelligent data-analysis techniques. She currently has over 50 refereed publication and has served in the organization and in program committees of several conferences. Dr El Gayar is a member of the program committee of the International Journal of Graphics, Vision and Image Processing. In recent years she has been appointed as the vice chair of the IAPR Technical Committee 3 on Neural Networks & Computational Intelligence.

Nashwa Abdelbaki received her Doctor of Engineering (Dr.-Ing.) degree in the field of multimedia networking from Faculty of Engineering, Ulm University, Germany. Supported by her German DAAD scholarship to work on her Ph.D. degree, she led a research program focused on the future integrated multimedia networking architecture and services in a collaborative environment. She received her M.Sc. degree from Faculty of Engineering, Ain Shams University in Cairo, Egypt, and B.Sc. from Faculty of Engineering, Cairo University. Currently she is Information Security Program Director, School of Communication and Information Technology, Nile University, Cairo, Egypt. She is also leading the research group cloud computing and interactive multimedia to introduce multiparty multiparticipant communication system especially for the university community. Nashwa Abdelbaki is an early Internet pioneer, bringing Internet connectivity to Egypt in the early 1990s. As part of this process, she helped to build Egypt's national networks both locally and regionally and took technical lead of the Egyptian Universities Network (EUN). From 1993 to 1999, she led, managed, organized and helped with teaching and educating a number of the international IT conferences and networking workshops (e.g., ISOC/INET) that targeted the methodologies and techniques to study, build, manage and maintain national networks. She was one of the four that submitted the first proposal to establish AfriNIC. Dr. Abdelbaki was nominated and served as an ICANN ccNSO Council member (2007-2010) and a member of the African Top Level Domain (AFTLD) Executive Committee. She established the basics of the new automated system of the central registry of the Egyptian Top Level Domain .EG.