Modeling Protocol Based Packet Header Anomaly Detector for Network and Host Intrusion Detection Systems

Solahuddin B. Shamsuddin and Michael E. Woodward

Department of Computing, School of Informatics
University of Bradford, United Kingdom
{S.B.Shamsuddin,M.E.Woodward}@bradford.ac.uk

Abstract. This paper describes an experimental protocol based packet header anomaly detector for Network and Host Intrusion Detection System modelling which analyses the behaviour of packet header field values based on its layer 2, 3 and 4 protocol fields of the ISO OSI Seven Layer Model for Networking. Our model which we call as Protocol based Packet Header Anomaly Detector (PbPHAD) Intrusion Detection System is designed to detect the anomalous behaviour of network traffic packets based on three specific network and transport layer protocols namely UDP, TCP and ICMP to identify the degree of maliciousness from a set of detected anomalous packets identified from the sum of statistically modelled individually rated anomalous field values.

Keywords: Anomaly, Data base, Network Intrusion Detection System.

1 Introduction

The advent of Intrusion Detection System (IDS) technologies have contributed a lot to the Network Security domain which have been the much talked about issues after a wave of the infamous 'code red' worm and its like i.e. 'self propagating malicious code' flooding and choking the internet traffic which almost caused a nearly catastrophic effect to the internet connected network infrastructures during this early part of the decade. Two major technologies which are commonly used in the design and development of the IDS are the signature based and anomaly based IDSs. We are focusing our IDS model based on the anomalous behaviour of the packet headers which behaves differently depending on the protocol used in the transmisson of a particular packet at network and transport layers.

In this experiment, we used MIT Lincoln Lab 1999 off-line intrusion detection evaluation data set [1] as the training and testing data as this data set has become one of the *de facto* standards for test data set among the IDS researcher community. A lot of well documented experiments have been published using this data set i.e. [2], [3], [4], [5], [6], [7], [8] and [9]. By using a skilfully crafted publicly available data set with a large quantity of rich background traffic, we would foresee that the result of our experiment would be very appealing as it can be compared with the published results by a number of researchers from renowned research institutions.

F. Bao et al. (Eds.): CANS 2007, LNCS 4856, pp. 209–227, 2007.

The rest of the paper is organized as follows. In section 2, we discuss other related works in intrusion detection system. In section 3, we describe PbPHAD model which include its design concept, process flow and statistical modelling. In Section 4, we discuss PbPHAD experimental results on 1999 DARPA evaluation data set. In section 5, we compare PbPHAD experimental results with the 1999 DARPA IDS evaluation best system results on poorly detected attacks. In section 6, we discuss the conclusion of our experiment. We present our future work in section 7.

2 Related Work

The fundamental inspiration behind our experiment was drawn from a Technical Report written by M.V. Mahoney and P.K. Chan that learns the normal range of values for 33 fields of the Ethernet, IP, TCP, UDP and ICMP protocols using a generic statistical model for all values in the packet headers for all protocols [10]. Our experiment in essence is to expand this idea of using just the packet header field values to learn the anomalous behaviour of the packets during transmission in any TCP/IP network traffic. We extend the statistical analysis by modelling the detection algorithm based on three specific network and transport layer protocols namely UDP, TCP and ICMP. Future analysis will be done using the combination of knowledge engineering methodologies which would eventually determine to some extent the degree of maliciousness of the detected anomalous packets in a cluster which is suspected to be intrusive through their assigned anomaly scores.

3 Protocol Based Packet Header Anomaly Detection (PbPHAD) Model

Fig. 1. [11] shows of an isolated test bed network for the 1999 DARPA offline evaluation. Scripting techniques were used to generate live background traffic which is similar to traffic that flows between the inside of one fictional Eyrie Air force base created for the evaluation to the outside internet. Rich background traffic was generated in the test bed which looks as if it were initiated by hundreds of users on

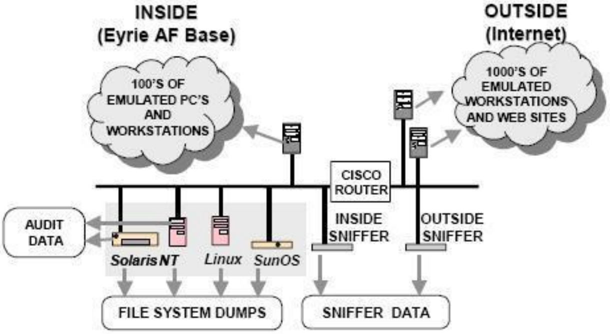

Fig. 1. Block Diagram of DARPA 1999 Test Bed

thousands of hosts. Automated attacks were launched against the UNIX victim machines and the router from outside hosts. Machines labelled 'sniffer' in Figure 1 run a program named *tcpdump* to capture all packets transmitted over the attached network segment. 5 weeks of data which comprise of 3 weeks of training data and 2 weeks of testing data are made available for evaluation in *tcpdump* format.

The packet header field values are taken from layer 2, 3 and 4 protocols which are the IP, Ethernet, TCP, UDP and ICMP which summed up to 33 fields as depicted in the Field Name column in Table 1. We designed our PbPHAD anomaly statistical model based on 3 specific protocols which are TCP, UDP and ICMP because of their unique behaviour when communicating among hosts, client and servers depending on the purpose and application used for a particular session. With this in mind, a more accurate statistical model with finer granularity which represents the 3 chosen protocols can be built for detecting the anomalous behaviour of the testing data.

For each protocol, if we index each field as i, i=1,2,…,n, the model is built based on the ratio of the normal number of distinct field values in the training data, R_i, against the total number of packets associated with each protocol, N_i. The ratio, $p_i = R_i/N_i$ represents the probability of the network seeing normal field values in a packet. Thus, the probability of anomalies will be 1- p_i for each corresponding field. Each packet header field containing values not found in the normal profile will be assigned a score of $1 - p_i$ and will be summed up to give the total value for that particular packet.

$$\text{Score}_{packet} = \sum_{i=1}^{n} (1 - p_i), \qquad i = 1,2,\ldots n \tag{1}$$

As the value of R_i varies greatly, we use log ratio in our model. The value of column TCP, UDP and ICMP in Table 1 is calculated based on:

Relative percentage ratio of 1-log(R_i/N_i)

to give the total probability of 1 for each protocol.

Table 1 shows PbPHAD statistical model. It is obvious from the PbPHAD model that the bigger the number of anomalous fields (R), the smaller the anomaly score will be. The anomaly score of 0.000 shows that particular field is not related to that particular protocol. From table 1 we can see the distinct value of destination IP (ipdest=1934) and source IP (ipsrc=1918) fields which depict the number of hosts simulated in the DARPA 1999 Test Bed as shown in Fig. 1.

Fig. 2. shows the process flow of building the PbPHAD Network Intrusion Detection System model. The process flow can be divided into 3 stages as follows:

- **Stage I. Data Preparation.** In this stage, training and testing data are downloaded from MIT Lincoln Lab web site. The raw data are in the form of compressed *tcpdump* format. We wrote a C++ program to extract the data from the *tcpdump* files and write the output to comma separated values (.csv) files. We took this approach due to the volume of the raw data. By doing bulk copying into the Ingres database, the process will be a lot faster as the size of the raw data alone occupy almost 6GB of hard disk space. We used *ethereal* to read the data in *tcpdump* format in order to verify the converted data in the .csv file format.

Table 1. PbPHAD Statistical Model

				ANOMALY SCORE		
i	Field Name	R	N	TCP	UDP	ICMP
1	etherdesthi	9	12,814,738	0.045	0.057	0.060
2	etherdestlo	12	12,814,738	0.045	0.056	0.059
3	etherprotocol	4	12,814,738	0.048	0.060	0.063
4	ethersize	1456	12,814,738	0.031	0.040	0.041
5	ethersrchi	6	12,814,738	0.047	0.059	0.061
6	ethersrclo	9	12,814,738	0.045	0.057	0.060
7	icmpchecksum	2	7,169	0.000	0.000	0.038
8	icmpcode	3	7,169	0.000	0.000	0.037
9	icmptype	3	7,169	0.000	0.000	0.037
10	ipchecksum	1	12,715,589	0.052	0.065	0.068
11	ipdest	1934	12,715,589	0.031	0.039	0.040
12	ipfragid	12,489	12,715,589	0.025	0.032	0.034
13	ipfragptr	2	12,715,589	0.050	0.062	0.065
14	ipheaderlength	1	12,715,589	0.052	0.065	0.068
15	iplength	1463	12,715,589	0.031	0.040	0.041
16	ipprotocol	3	12,715,589	0.049	0.061	0.064
17	ipsrc	1918	12,715,589	0.031	0.039	0.040
18	iptos	4	12,715,589	0.048	0.060	0.063
19	ipttl	11	12,715,589	0.045	0.057	0.059
20	tcpack	6,015,527	10,617,293	0.008	0.000	0.000
21	tcpchecksum	2	10,617,293	0.049	0.000	0.000
22	tcpdestport	22,293	10,617,293	0.023	0.000	0.000
23	tcpflag	10	10,617,293	0.045	0.000	0.000
24	tcpheaderlength	3	10,617,293	0.048	0.000	0.000
25	tcpoption	3	10,617,293	0.048	0.000	0.000
26	tcpseq	7,357,319	10,617,293	0.007	0.000	0.000
27	tcpsrcport	22,293	10,617,293	0.023	0.000	0.000
28	tcpurgptr	2	10,617,293	0.049	0.000	0.000
29	tcpwindowsize	10,705	10,617,293	0.025	0.000	0.000
30	udpchecksum	2	2,091,127	0.000	0.056	0.000
31	udpdestport	8,050	2,091,127	0.000	0.027	0.000
32	udplength	129	2,091,127	0.000	0.042	0.000
33	udpsrcport	8,051	2,091,127	0.000	0.027	0.000
n	**TOTAL**	**13,463,719**		**1.000**	**1.000**	**1.000**

The attack identification file is available in the text format from the Lincoln Lab web site. We verified each attack in the testing table in the database using SQL query before converting it into .csv format file prior inserting it into the database. It is very interesting to note that the number of packets which constitute an attack instance differs greatly from only 1 packet for an attack (i.e. *land*, *syslogd*) to 179,983 packets for *udpstorm*. There are 201 attack instances embedded in the MIT Lincoln Lab evaluation data set for both inside and outside testing data. Out of 201 attack instances only 176 are found in the inside testing data used for this experiment. Our performance evaluation will be based on the 176 attack instances as we only use the inside testing data.

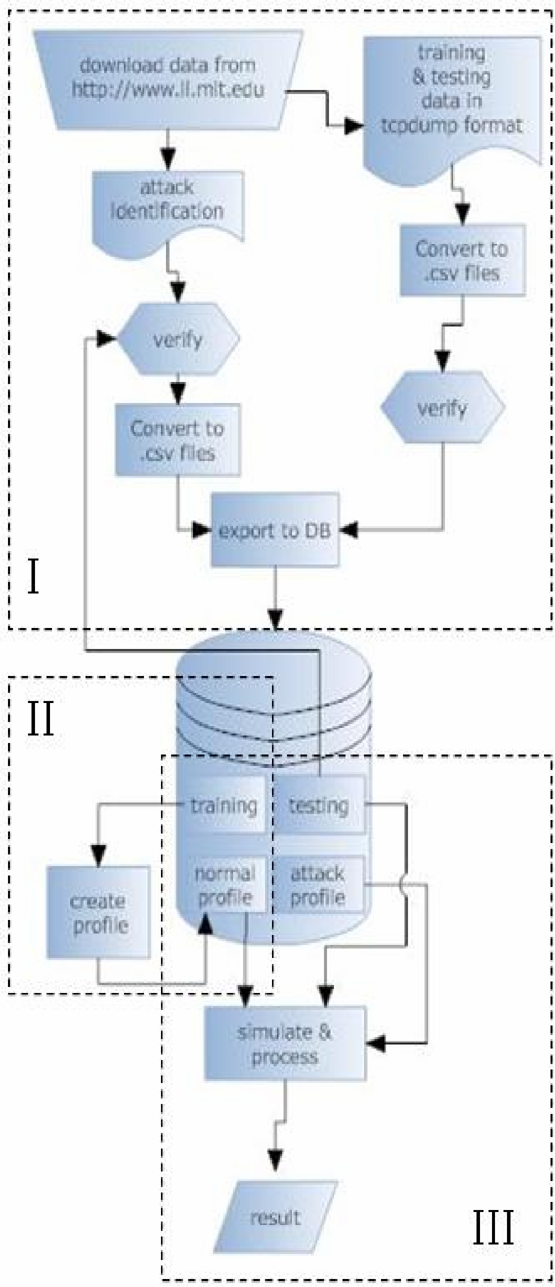

Fig. 2. PbPHAD Process Flow

Table 2. Distribution of all Attack Categories by Protocol

Category	**TCP**	**UDP**	**ICMP**	**TOTAL**
(a)	**(b)**	**(c)**	**(d)**	**(e)**
Probe	30	7	8	45
DOS	37	10	7	54
U2R	27	0	0	27
R2L	54	3	0	57
Data	4	2	0	6
Total	**152**	**22**	**15**	**189**

The distribution of all attacks in the inside testing data is as follows:

The total attacks shows 13 extra attacks (189 – 176) which is caused by duplicated protocols in the attacks. i.e. one attack instance uses more than 1 protocol.

- **Stage II. Building the Normal Profile.** In this stage, we wrote a program to build a normal profile table which was taken from week 3 of the training data. Distinct values for each of the 33 fields in the TCP/IP packet were inserted into normal profile table to be used in the experiment to detect anomalous packet header field values.
- **Stage III. Running the Experiment.** In this stage, we simulate the network traffic for the 2 weeks of the testing data and used our model to detect the anomalous packets. Each one of the 33 fields in the packet (depending on the protocol) was compared with its corresponding normal profile. If a field value was not found in the normal profile, an anomaly score will be assigned to the packet as was statistically modelled in Table 1. If the sum of all its anomalous field values surpassed a certain preset threshold, it will be captured into a detected anomalous table. Another program was run to compare the detected anomalous packets against the attack database to classify each and every packet into either true positive or false positive.

4 Experimental Results on the 1999 DARPA IDS Evaluation Data Set

4.1 Network-Based PbPHAD

We tested our model on the 2 weeks of the inside testing data which comprises of 22,095,072 packets and managed to detect 121 attack instances as depicted in Table 3 below:

This is the result from all detected anomalous packets which had surpassed certain preset thresholds (TCP=0.041, UDP=0.128, ICMP=0.034) of the anomaly score. The detected anomalous packets represents about 10% of the total test data including all false positives. It should be noted that no attack was detected on 30/03/1999 as the test data for this particular date was missing from the test data set.

Table 3. Detection Result

Date	ICMP	UDP	TCP	Sub-Total
29/03/1999	0	1	10	11
30/03/1999	0	0	0	0
31/03/1999	1	1	1	3
01/04/1999	2	0	9	11
02/04/1999	4	0	6	10
03/04/1999	1	0	1	2
04/04/1999	0	0	0	0
05/04/1999	6	0	9	15
06/04/1999	0	3	14	17
07/04/1999	0	1	14	15
08/04/1999	1	1	9	11
09/04/1999	0	4	17	21
10/04/1999	0	0	5	5
Total	**15**	**11**	**95**	**121**

The distribution of detected attack categories by protocol is tabulated in Table 4. Only one attack instance is counted even though it was detected through more than 1 protocol. The success rate percentage in column (f) is in relation to total attacks in the testing data as shown in column (e) of Table 2. Total success rate of 68.75% is calculated based on 176 total attack instances found in the experimented inside testing data.

Table 4. Distribution of Detected Attack Categories by Protocol

Category	TCP	UDP	ICMP	TOTAL	Success Rate
(a)	**(b)**	**(c)**	**(d)**	**(e)**	**(f)**
Probe	27	4	8	39	86.67%
DOS	23	7	7	37	68.52%
U2R	16	0	0	16	59.26%
R2L	28	0	0	28	49.12%
Data	1	0	0	1	16.67%
TOTAL	**95**	**11**	**15**	**121**	**68.75%**
Percentage	**62.5%**	**50%**	**100%**	**68.75%**	

18 packet header fields have been observed to have contributed to the anomaly score for the detected attacks. The distribution of the frequency of anomalous fields is tabulated in Table 5.

The rest of the 15 packet header fields have been noted as non-contributors to the anomaly scores of the detected anomalous packets. From Table 5, we can design our next model by just taking into account the contributing packet header fields only so that the processing time to detect anomalous packets can be reduced.

Table 5. Distribution of Contribution of Anomalous Packet Header Fields to Detected Attacks

Ser	Packet Header Field	Frequency
1	tcpseq	83
2	ipsrc	60
3	ipfragid	53
4	tcpack	50
5	ipdest	34
6	tcpsrcport	16
7	tcpdestport	11
8	tcpwindowsize	8
9	udpsrcport	8
10	ipfragptr	7
11	udpdestport	6
12	udplen	6
13	iplength	5
14	tcpflag	4
15	tcpurgptr	3
16	tcpchecksum	2
17	etherdesthi	1
18	etherdestlo	1

Table 6, 7 and 8 shows top 5 anomaly scores for ICMP, UDP and TCP protocols respectively. Anomalous field column shows fields that contributed to the score.

Duplicate attack names indicate the same attack on different destination hosts at different time of the day which are to be counted as separate attack instances.

Table 6. Top 5 Anomaly Scores for ICMP Packets

Ser	Attack Name	Score	Anomalous Field
1	ipsweep	0.132	ipfragid=20751; ipdest=204.233.047.021
2	pod	0.109	ipfragptr=x2000; ipsrc=202.077.162.213
3	pod	0.109	ipfragptr=x2000; ipsrc=202.077.162.213
4	smurf	0.109	ipfragptr=x2000; ipsrc=202.077.162.213
5	pod	0.109	ipfragptr=x2000; ipsrc=010.011.022.033

From table 6, for ICMP packets, it shows that ICMP protocol fields themselves are not exploited in the attack. For TCP and UDP packets, their corresponding protocol fields contributed significantly to the anomaly score for the detected anomalous packet.

Table 7. Top 5 Anomaly Scores for UDP Packets

Ser	Attack Name	Score	Anomalous Field

1	illegalsniffer	0.217	etherdesthi=x00104B;
			etherdestlo=xA26739;
			ipfragid=33248;
			ipdest=172.016.112.097;
			udpdestport=1024

2	portsweep	0.217	iplength=28;
			ipfragid=38809;
			ipsrc=153.010.008.174;
			udpsrcport=60716;
			udpdestport=513;
			udplen=8

3	teardrop	0.160	ipfragptr=x2000;
			ipsrc=207.230.054.203;
			udpsrcport=17631;
			udpdestport=23

4	teardrop	0.160	ipfragptr=x2000;
			ipsrc=199.227.099.125;
			udpsrcport=24891;
			udpdestport=23

5	syslogd	0.154	iplength=32;
			ipsrc=172.003.045.001;
			udpsrcport=514;
			udplen=12

Table 8. Top 5 Anomaly Scores for TCP Packets

Ser	Attack Name	Score	Anomalous Field

1	portsweep	0.236	iplength=28;
			ipfragid=58448;
			ipfragptr=x2000;
			ipsrc=206.048.044.050;
			tcpsrcport=50460;
			tcpseq=3192052884;
			tcpchecksum=x77F7

2	portsweep	0.175	ipsrc=192.168.001.001;
			ipdest=172.016.118.010;
			tcpdestport=63432;
			tcpseq=3269601754;
			tcpack=3303464411;
			tcpwindowsize=4128

3	phf	0.174	ipfragid=46639;
			ipsrc=206.048.044.050;
			tcpseq=242486627;
			tcpflag=x01;
			tcpchecksum=x9397

Table 8. (*continued*)

4	portsweep	0.173	ipfragid=47803;
			ipdest=153.010.008.174;
			tcpdestport=49998;
			tcpseq=1320219032;
			tcpack=36059013;
			tcpwindowsize=9112

5	dosnuke	0.165	ipfragid=59399;tcpseq=47711425;
			tcpack=47585391;
			tcpflag=x39;
			tcpurgptr=196

4.2 Host-Based PbPHAD

For Host-based PbPHAD, we built the normal profile for each host by taking the packet header field values from layer 3 and 4 protocols only which are the IP, TCP, UDP and ICMP without its layer 2 protocol, the ethernet. The total fields tested for anomaly in this model is 27 as depicted in the field name column in Table 1 minus the first 6 field names which belong to ethernet protocol. We built 2 different normal profiles, one for incoming packets and the other for outgoing packets for each inside host with the intention to acquire a more accurate statistical model with finer granularity for each of the 3 chosen protocols; TCP, UDP and ICMP.

We tested Host-based PbPHAD on the 2 weeks of the inside testing data which comprises of 22,095,072 packets. This is the same data set we used for testing the Network-based PbPHAD. Host-based PbPHAD managed to detect more attacks compared to its peer, the Network-based PbPHAD by 33 attacks (154 – 121) even though it only used layer 3 and 4 protocol fields for anomaly detection. See Table 9.

Table 9. Detection Result for Host-based PbPHAD

Date	ICMP	UDP	TCP	Sub-Total
29/03/1999	0	0	12	12
30/03/1999	0	0	0	0
31/03/1999	1	1	13	15
01/04/1999	2	0	11	13
02/04/1999	4	0	9	13
03/04/1999	3	0	0	3
04/04/1999	0	0	0	0
05/04/1999	4	0	10	14
06/04/1999	0	3	19	22
07/04/1999	0	1	17	18
08/04/1999	1	1	11	13
09/04/1999	0	4	22	26
10/04/1999	0	0	5	5
Total	**15**	**10**	**129**	**154**

This is quite a significant improvement as it shows an increment of 27.27%.

Table 10. Distribution of Detected Attack Categories by Protocol for Host-based PbPHAD

Category	TCP	UDP	ICMP	TOTAL	Success Rate
(a)	**(b)**	**(c)**	**(d)**	**(e)**	**(f)**
Probe	26	3	8	37	82.22%
DoS	28	7	7	42	77.78%
U2R	27	0	0	27	100.00%
R2L	45	0	0	45	78.95%
Data	3	0	0	3	50.00%
TOTAL	**129**	**10**	**15**	**154**	**81.48%**
Percentage	**84.87%**	**45.45%**	**100%**	**81.48%**	

Table 10 shows that Host-based PbPHAD managed to detect all attacks in U2R category as compared to its Network-based PbPHAD peer as depicted in Table 4. It decreases slightly by 4.45% on Probe category and increase by 9.26% on DoS category. For R2L category, it increases quite significantly by 29.83% and a bigger increment can be observed for attack category of Data which is 33.33%.

Host-based PbPHAD shows a significant improvement in terms of detecting number of anomalous fields as shown in Table 11. Host-based PbPHAD managed to detect 25 anomalous fields compared to only 18 by Network-based PbPHAD. Table 11 shows that the Host-based model could detect anomalous fields with a finer granularity. 9 packet header fields (Serial No. 17-25) are new anomalous fields detected by Host-based PbPHAD which are not detected by Network-based PbPHAD.

Table 11. Distribution of Contribution of Anomalous Packet Header Fields to Detected Attacks for Host-based PbPHAD

Ser	Packet Header Field	Frequency for Network-based PbPHAD	Frequency for Host-based PbPHAD
1	tcpseq	83	125
2	ipsrc	60	96
3	ipfragid	53	15
4	tcpack	50	55
5	ipdest	34	13
6	tcpsrcport	16	64
7	tcpdestport	11	49
8	tcpwindowsize	8	22
9	udpsrcport	8	6
10	ipfragptr	7	9
11	udpdestport	6	7
12	udplen	6	7

Table 11. (*continued*)

13	iplength	5	38
14	tcpflag	4	5
15	tcpurgptr	3	0
16	tcpchecksum	2	0
17	ipheaderlen	-	1
18	Iptos	-	1
19	Ipttl	-	1
20	ipprotocol	-	3
21	ipchecksum	-	1
22	tcpheaderlength	-	3
23	udpchecksum	-	2
24	icmptype	-	6
25	icmpcode	-	1

As described for the Network-based PbPHAD above, duplicate attack names indicate the same attack on different destination hosts at different time of the day which are to be counted as separate attack instances. Different anomalous field values for the same anomaly score shows each host has its own outgoing and incoming normal profile and the anomaly score for each host differs from other hosts as the normal profile for each host is unique to that particular host only as each host interact with different set of incoming and outgoing packets during training.

Table 12. Top 5 Anomaly Scores for ICMP Packets

Ser	Attack Name	Score	Anomalous Field

1	ipsweep	0.340	iplength=38;
			ipfragid=104;
			ipsrc=194.027.251.021;
			icmptype=8

2	ipsweep	0.340	iplength=38;
			ipfragid=2811;
			ipsrc=194.007.248.153;
			icmptype=8

3	ipsweep	0.339	iplength=38;
			ipfragid=15514;
			ipsrc=207.136.086.223;
			icmptype=8

4	ipsweep	0.339	ipsrc=204.233.047.021;

5	portsweep	0.318	ipdest=208.240.124.083;
			icmptype=3;
			icmpcode=3

From table 12, for ICMP packets, in contrary to network-based PbPHAD, Host-based PbPHAD managed to detect anomalous ICMP protocol fields. This shows that the ICMP fields are indeed being exploited in some of the attacks. This is a new interesting finding as the Network-based PbPHAD failed to detect any anomalous ICMP fields being exploited in any of the attacks. For UDP and TCP packets as shown in Table 13 and Table 14, their corresponding protocol fields contributed significantly to the anomaly score for the detected anomalous packets as similar as shown by the Network-based PbPHAD in Table 7 and Table 8 respectively.

Table 13. Top 5 Anomaly Scores for UDP Packets

Ser	Attack Name	Score	Anomalous Field

1	teardrop	0.312	ipfragptr=x2000;
			ipsrc=207.230.054.203;
			udpsrcport=17631;
			udpdestport=23;
			udplen=36

2	teardrop	0.312	ipfragptr=x2000;
			ipsrc=199.227.099.125;
			udpsrcport=24891;
			udpdestport=23;
			udplen=36

3	satan	0.277	ipsrc=209.030.070.014

4	syslogd	0.272	iplength=32;
			ipsrc=172.003.045.001;
			udpsrcport=514;
			udpdestport=514;
			udplen=12

5	portsweep	0.272	iplength=28;
			ipsrc=153.010.008.174;
			udpsrcport=60716;
			udpdestport=513;
			udplen=8

Table 14. Top 5 Anomaly Scores for TCP Packets

Ser	Attack Name	Score	Anomalous Field

1	portsweep	0.594	iplength=28;
			ipfragptr=x2000;
			ipsrc=206.048.044.050;
			tcpsrcport=49826;
			tcpdestport=514;
			tcpseq=2162256216;
			tcpack=1767401816;
			tcpheaderlen=x69

2	mscan	0.431	iplength=44;
			ipfragid=30133;
			ipdest=207.136.086.223;
			ipprotocol=6;
			tcpsrcport=25;
			tcpdestport=13074;
			tcpseq=1865002828;
			tcpack=3222202810;
			tcpheaderlen=x60

Table 14. (*continued*)

3	ipsweep	0.401	tcpsrcport=1885;
			tcpdestport=80;
			tcpseq=3295102387

4	dosnuke	0.356	ipfragid=46087;
			ipsrc=206.048.044.018;
			tcpsrcport=1734;
			tcpdestport=139;
			tcpseq=43860484;
			tcpflag=x02;
			tcpwindowsize=8192

5	tcpreset	0.319	ipfragid=35357;
			tcpdestport=26398;
			tcpseq=487325652;
			tcpack=3809752458

5 Comparison with the 1999 DARPA IDS Evaluation Best System Result

We made a comparison between PbPHAD with the combined 1999 DARPA evaluation best systems in each category of attack results on poorly detected attacks as documented by Lippman et al [11]. This analysis was performed to determine how well all 18 evaluated intrusion detection system models submitted by 8 research groups taken together detect attacks regardless of false alarm rates. The best system was first selected for each attack as the system which detects the most instances of that attack which will serve as a rough estimation for upper bound on composite system performance. Our results are in column (f) and (g) as shown in Table 15 below for Network-based PbPHAD and Host-based PbPHAD respectively.

5.1 Network-Based PbPHAD

Our initial analysis shows that Network-based PbPHAD managed to detect 48 attacks as compared to only 15 attacks detected by the composite best systems. This result shows an increment of 39.76% on detection rate for the poorly detected attacks. Our model managed to detect 9 out of 10 attacks which were not detected by all evaluated systems as compared to only 4 attacks we did not detect which were detected by the best systems.

Both Network-based PbPHAD and all DARPA evaluated systems failed to detect 1 attack which is *snmpget*. As for the type of attacks detected (58 total), Network-based PbPHAD managed to detect 53 attack types as compared to 48 attack types for composite systems. On this aspect, PbPHAD demonstrated an increment of 8.62% on the detection rate.

5.2 Host-Based PbPHAD

Column (g) in Table 15 shows attacks detected by Host-based PbPHAD for attacks which are classified as ‘poorly detected’ by the 1999 DARPA evaluation best

Table 15. Comparison between the 1999 DARPA Evaluation Best Systems and PbPHAD on Poorly Detected Attacks

Ser	Name	Cat.	Tot. Inst.	Instance Detected by Best System	Network-Based PbPHAD	Host-Based PbPHAD
(a)	**(b)**	**(c)**	**(d)**	**(e)**	**(f)**	**(g)**
1	ipsweep	Probe	7	0	7	7
2	lsdomain	Probe	2	1	2	2
3	portsweep	Probe	13	3	13	13
4	queso	Probe	4	0	2	3
5	resetscan	Probe	1	0	1	1
6	arppoison	DoS	5	1	0	0
7	dosnuke	DoS	4	2	4	4
8	selfping	DoS	3	0	1	1
9	tcpreset	DoS	3	1	2	2
10	warezclient	DoS	3	0	3	3
11	ncftp	R2L	5	0	4	5
12	netbus	R2L	3	1	2	2
13	netcat	R2L	4	2	0	4
14	snmpget *	R2L	4	0	0	0
15	sshtrojan	R2L	3	0	1	1
16	loadmodule	U2R	3	1	0	2
17	ntfsdos *	U2R	3	1	0	0
18	perl	U2R	4	0	3	3
19	sechole	U2R	3	1	1	2
20	sqlattack	U2R	3	0	1	2
21	xterm	U2R	3	1	1	3
Total			**83**	**15**	**48**	**61**
Percentage Detected				**18.07%**	**57.83%**	**73.49%**
Increment					**39.76%**	**55.41%**

systems. Host-based PbPHAD shows a significant improvement in terms of detection of number of attacks and new attacks. Host-based PbPHAD managed to detect 2 new attacks which were not detected by Network-based PbPHAD which are *netcat* and *loadmodule*. 2 attack instances which have been marked by an asterisk (* *snmpget* and *ntfsdos*) - are attacks which are only found in outside testing data, which PbPHAD did not attempt to detect as we only used inside testing data in our experiment.

For the 1999 DARPA category of 'poorly detected' attack, Host-based PbPHAD fails to detect only one attack which is *arppoison* and managed to detect 7 attacks which were totally missed by all systems participated in the second 1999 DARPA off-line intrusion detection evaluation. *Arppoison* operates at layer 2, which is the data link layer which Host-based PbPHAD excluded from its model.

6 Conclusions

Our PbPHAD model has been demonstrated as a very promising model to be used for an anomaly based IDS model by analyzing anomalous behaviour of the packet header fields on three prominent protocols.

To summarize, Network-based PbPHAD has shown the following results worthy of note:

- On the overall category of attack, Network-based PbPHAD has shown a good percentage of detection rate which is 68.75%. Network-based PbPHAD demonstrated a high percentage of detection rate for Probe and DOS which is 86.67% and 68.52% respectively.
- On the type of attacks by protocol, Network-based PbPHAD managed to detect 62.5% for TCP, a perfect 100% for ICMP and an average performance achievement for UDP at 50%. It can be seen from Table 2 and Table 4 that Network-based PbPHAD shows to be a perfect model to detect Probe and DOS attacks exploiting ICMP protocols.
- In comparison with the combined 1999 DARPA best systems for the best attack rate on poorly detected attacks, Network-based PbPHAD achieved 39.76% increment on the detection rate.
- On the number of attack types detected, Network-based PbPHAD demonstrated an increment of 8.62% on the detection rate as compared to all 1999 DARPA evaluated systems.
- On the number of ‘poorly detected’ attack instances which were not detected by all 1999 DARPA evaluated systems, Network-based PbPHAD is better by 60%. All DARPA evaluated combined systems failed to detect 10 attack instances as compared to only 4 attack instances not detected by Network-based PbPHAD. This clearly shows that Network-based PbPHAD could cover different attack space that could not be covered by all 1999 DARPA evaluated IDS models.

Host-based PbPHAD has demonstrated quite a significant improvement compared to its peer, the Network-based PbPHAD. Our Host-based PbPHAD anomaly based IDS model has shown that it has succeeded in complementing the existing techniques implemented by all 18 IDS models evaluated in the 1999 DARPA off-line intrusion detection evaluation exercise. This experiment has shown that it has paved a way for discovering new dimension of attack space. This shall bequeath a very promising optimism for IDS researcher community in designing new IDS model based on anomaly and host profiling.

By analyzing the detection results on both network and host based models, we can see that Network-based PbPHAD is better in terms of detecting number of attacks for Probe attack category compared to Host-based PbPHAD. This is not surprising as the Network-based PbPHAD is capable of seeing bigger attack horizon compared to the Host-based PbPHAD. Network-based PbPHAD model can see both horizontal and vertical scannings whereas Host-based PbPHAD is not capable to detect horizontal scanning as it only analyzes packets attacking its own IP only. These results show that deploying both Network-based and Host-based IDS models in a particular network installation could give a broader coverage of attack space in defending network infrastructure from malicious attacks.

7 Future Work

The percentage of false positive is still quite big for the detected anomalous packets based on the statistical model alone. Thus, we will be working on expert production rules to reduce the number of false positives. The format of the production rules is similar to other rules found in artificial intelligence techniques in the form of antecedent and consequent. Some example of the rules which will be inferred to the detected anomalous packets will be in the form as shown below:

Rule 1
Antecedent
IF destination IP address is anomalous
AND destination port number is the well known server port number which is in normal profile for that particular host
AND session is initiated by the inside host
Consequent
THEN Reduce the anomaly score by the destination IP anomaly value
i.e. normal internet connection for HTTP traffic using port 80.

Rule 2
Antecedent
IF source IP address is anomalous
AND destination port number is the well known server port number which is in normal profile for that particular host
AND session is initiated by the outside host
Consequent
THEN Reduce the anomaly score by the source IP anomaly value
i.e. normal FTP traffic for downloading file using port 21 as normal service offered by the inside host.

Fig. 3. shows a new detection process flow chart when expert production rules are included as part of the detection process. The process can be segregated into 3 stages as stage I, II & III as depicted in Fig. 3. In stage I, each packet will be examined for its anomaly using the statistical model and will be assigned an anomaly score accordingly. If the anomaly score is greater than the threshold for its protocol, it will go to stage II. Before entering stage II, the packet will be segregated based on its protocol. An ICMP packet will branch out to be inferred by an ICMP expert production rule whereas UDP and TCP packet will branch out to another production rule. In stage II, expert production rule will be inferred to the packet to examine its anomaly and a new anomaly score will be calculated. If the score is still greater than the threshold, the packet will have to go to stage III to be inferred with another layer of expert production rule. After completion of stage III, the packet anomaly score will be examined once again. If the score is still greater than the threshold it will be recorded as anomalous.

Our aim is to reduce the number of false positives to a maximum of only 10 FPs per day for our next performance evaluation benchmark.

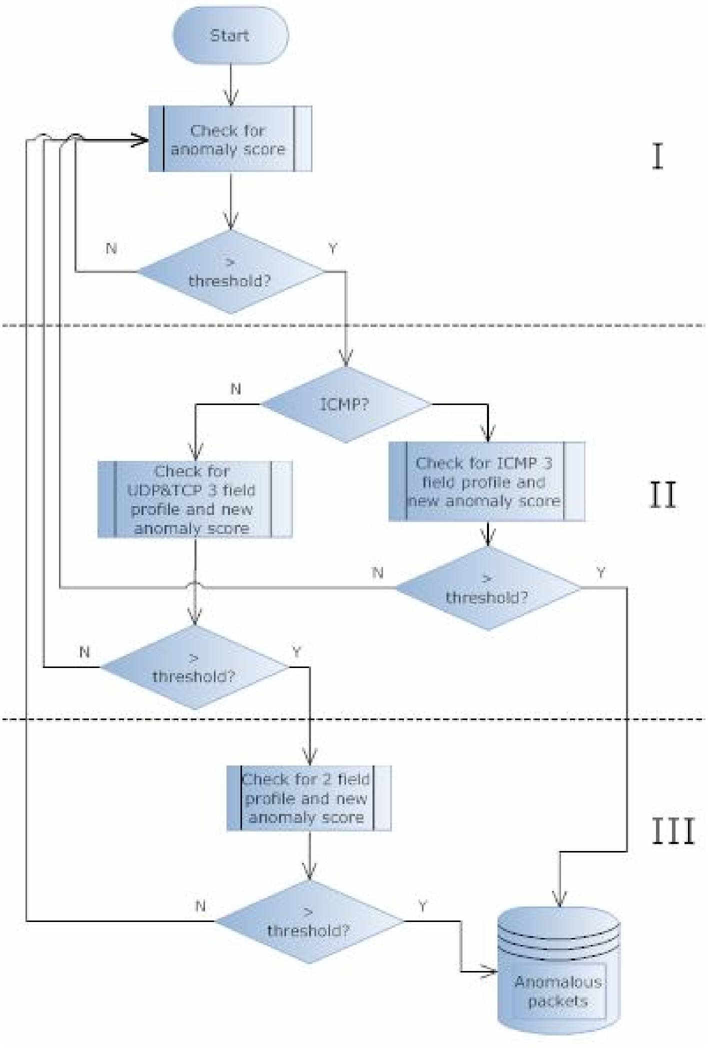
Start
Check for anomaly score
I
N
Y
> threshold?
N
ICMP?
Check for UDP&TCP 3 field profile and new anomaly score
Check for ICMP 3 field profile and new anomaly score
II
N
> threshold?
Y
N
> threshold?
Y
Check for 2 field profile and new anomaly score
III
N
> threshold?
Y
Anomalous packets

Fig. 3. PbPHAD Detection Process Flow Chart

References

1. MIT Lincoln Laboratory 1999 DARPA Intrusion Detection Data Sets (1999), http://www.ll.mit.edu/IST/ideval/data/1999/1999_data_index.html
2. Wang, K., Stolfo, S.J.: Anomalous Payload-based Network Intrusion Detection. In: Jonsson, E., Valdes, A., Almgren, M. (eds.) RAID 2004. LNCS, vol. 3224, pp. 201–222. Springer, Heidelberg (2004)
3. Mahoney, M.V., Chan, P.K.: Learning Rules for Anomaly Detection of Hostile Network Traffic. In: Proceeding of the 3rd IEEE International Conference on Data Mining (2003)

4. Luo, S., Marin, G.A.: Modeling Networking Protocols to Test Intrusion Detection Systems. In: LCN 2004. Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (2004)
5. Ertoz, L., Eilertson, E., Lazarevic, A., Tan, P.N., Dokas, P., Kumar, V., Srivastava, J.: Detection of Novel Network Attacks Using Data Mining. In: Proc. of SIAM Conf. Data Mining (2003)
6. Bolzoni, D., Etalle, S., Hartel, P., Zambon, E.: POSEIDON: A 2-Tier Anomaly Based Intrusion Detection System. In: IWIA 2006. Proceedings of the Fourth IEEE International Workshop on Information Assurance, pp. 144–156 (2006)
7. Vliet, F.V.: Turnover Poseidon: Incremental Learning in Clustering Methods for Anomaly based Intrusion Detection. In: Proceedings of Twente Student Conference on IT, University of Twente (2006)
8. Barbara, D., Couto, J., Jajodia, S., Popyack, L., Wu, N.: ADAM: Detecting intrusions by data mining. In: Proc. of the IEEE Workshop on Information Assurance and Security (June 2001)
9. Yin, C., Tian, S., Huang, H., He, J.: Applying Genetic Programming to Evolve Learned Rules for Network Anomaly Detection. In: Wang, L., Chen, K., Ong, Y.S. (eds.) ICNC 2005. LNCS, vol. 3612, pp. 323–331. Springer, Heidelberg (2005)
10. Mahoney, M.V., Chan, P.K.: PHAD: Packet Header Anomaly Detection for Identifying Hostile Network Traffic. Technical report, Florida Tech., technical report CS-2001-4 (April 2001)
11. Lippmann, R.P., Haines, J.W., Fried, D.J., Korba, J., Das, K.: The 1999 DARPA Off-Line Intrusion Detection Evaluation. MIT Lincoln Lab Technical Report (2000)