Different Approaches for the Detection of SSH Anomalous Connections

S. González[a], Á. Herrero[b], J. Sedano[a], UrkoZurutuza[c], and E. Corchado[d]

[a]*Instituto Tecnológico de Castilla y León, C/ López Bravo 70, Pol. Ind. Villalonquejar, 09001, Burgos, Spain. E-mail: javier.sedano@itcl.es, silvia.gonzalez@itcl.es*

[b]*Department of Civil Engineering, University of Burgos, University of Burgos,C/ Francisco de Vitoria s/n, 09006 Burgos, Spain. E-mail: ahcosio@ubu.es*

[c]*Electronics and Computing Department, Mondragon University, Goiru Kalea, 2, 20500 Arrasate-Mondragon, Spain.E-mail: uzurutuza@mondragon.edu*

[d] *Departamento de Informática y Automática, Universidad de Salamanca, Plaza de la Merced, s/n, 37008 Salamanca, Spain. E-mail: escorchado@usal.es*

Abstract. The Secure Shell Protocol (SSH) is a well-known standard protocol, mainly used for remotely accessing shell accounts on Unix-liked operating systems to perform administrative tasks. As a result, the SSH service has been an appealing target for attackers, aiming to guess root passwords performing dictionary attacks, or to directly exploit the service itself. To identify such situations, present paper addresses the detection of SSH anomalous connections from an Intrusion Detection perspective. The main idea is to compare several strategies and approaches for a better detection of SSH-based attacks. To test the classification performance of different classifiers and combinations of them, SSH data coming from a real-world honeynet are gathered and analysed. For comparison purposes and to draw conclusions about data collection, both packet-based and flow data are analysed. A wide range of classifiers and ensembles are applied to these data, as well as different validation schemes for better analysis of the obtained results. The high-rate classification results lead to positive conclusions about the identification of malicious SSH connections.

KEYWORDS: Secure Shell Protocol, SSH, Honeynet, Intrusion Detection, Classifier, Ensemble, Cross-Validation.

1. Introduction

The Secure Shell Protocol (SSH) is a standard application-layer (under the TCP/IP stack) protocol for remote login but also used for other secure network services over an insecure network. It consists of three major components:

- Transport Layer Protocol: provides server authentication, confidentiality, and integrity with perfect forward secrecy.
- User Authentication Protocol: authenticates the client to the server.
- Connection Protocol: multiplexes the encrypted tunnel into several logical channels.

The main usage of SSH protocol is for remotely accessing shell accounts on Unix-liked operating systems. As a result, most of the tasks and activities performed over this protocol are related with administrative purposes, such as user management, device configuration, permission assignment, etc. For this reason, the SSH service has been for years an attractive target for attackers, as crucial information travel over it. Intruders then try to guess passwords for malicious purposes, performing dictionary attacks, or to directly exploit the service itself. Weak passwords are targeted as there is no need for attackers to get the password of a root user account; there are many ways to increase the privileges of a user once logged-in. Furthermore, getting SSH access to remote hosts may be one of first steps for further attacks over SSH tunneling, such as SPAM sending.

Differentiating from other remote-communication protocols (File Transfer Protocol or Telnet), SSH encrypts the login session as a prevention mechanism to avoid the collection of unencrypted data (passwords and some other crucial data). Actually, SSH was conceived as a secure protocol to replace previous unsecure solutions to run sessions on remote host. However, the SANS Institute's Internet Storm Center [1] keeps monitoring an average of 100,000 targets being attacked on SSH default port number every day in Internet. From time to time, some attacks peaks are produced, as the one on March 2014 (8x baseline).

As a result of the above mentioned, being able of distinguishing between malicious and benign SSH traffic for server administration, may play an indispensable role in defending system administrators against malicious adversaries. This is one of the targets of Intrusion Detection Systems (IDSs) [2-4], that have become an essential asset in addition to the computer security infrastructure of most organizations. In the context of computer networks, an IDS can roughly be defined as a tool designed to detect suspicious patterns that may be related to a network or system attack. Intrusion Detection (ID) is therefore a field that focuses on the identification of attempted or ongoing attacks on a computer system (Host IDS - HIDS) or network (Network IDS - NIDS). While prevention mechanisms are aimed at avoiding intrusions, ID relies on the idea that intrusions will succeed and then, they must be identified for security response. In this case, identification of SSH anomalous connections while they are being run, certainly is an important task as it may reduce the impact of attacks thanks to abortion mechanisms.

The aim of the present study is to assess data collection, classifiers and ensembles in the useful task of detecting bad-intentioned SSH connections. As a result, best practices for SSH connection filtering may be proposed. To do so, real data, coming from the Euskalert honeynet[5] are gathered and analysed as described in the remaining sections of the paper.

A honeypot has no authorised function or productive value within the corporate network other than to be explored, attacked or compromised [6]. Thus, a honeypot should not receive any traffic at all. Any connection attempt with a honeypot is then an attack or attempt to compromise the device or services that it is offering. From the security point of view, there is a great deal of information that may be learnt from a honeypot about a hacker's tools and methods in order to improve the protection of information systems.

In a honeynet, all the traffic received by the sensors is suspicious by default. Thus every packet should be considered as an attack or at least as a piece of a multi-step attack. But, in the case of SSH, a honeynet also receives legitimate connections for the administration of the honeynet itself. As a result, for present study, data from "normal" SSH connections is also available. Numerous studies propose the use of honeypots to detect automatic large scale attacks; honeyd [7] and nepenthes [8] among others. The first Internet traffic monitors known as Network Telescopes, Black Holes or Internet Sinks were presented by Moore *et al*. [9].

The Euskalert honeynet [10], whose SSH data are analysed in present paper, has been monitoring attacks against well-known services. A network of honeypots has been deployed in the Basque Country (northern Spain), where eight companies and institutions have installed one of the project's sensors behind the firewalls of their corporate networks. The honeypot sensor transmits all the traffic received to a database via a secure communication channel. These partners can consult information relative to their sensor (after a login process) as well as general statistics in the project's website. Once the system is fully established, the information available can be used to analyse attacks suffered by the honeynet at network and application level. Euskalert is a distributed honeypot network based on a Honeynet

GenIII architecture [11]. As previously mentioned, the Euskalert sensors have also recorded the SSH sessions used to administer and maintain the different devices of the infrastructure.

ID has been previously approached from several different points of view; many different Computational Intelligence techniques - such as Genetic Programming [12], Data Mining [13-15], Expert Systems [16], Fuzzy Logic [17], or Neural Networks [18-20] among others - together with statistical [21] and signature verification [22] techniques have been applied mainly to perform a 2-class classification (normal/anomalous or intrusive/non-intrusive). More precisely, attacks to SSH service have attracted researchers' attention for a long time. Song et al. [23] analysed timing and keystroke attacks. Researchers have also used honeypots to study and analyse attacks to this protocol, focusing on login attempts and dictionary attacks [24], [25]. In [25] authors analyse SSH attacks on honeypots focusing on visualisation of the gathered data.

Considering the data capture, as previously introduced, the present study takes advantage of the Euskalert project. Its data have been analysed and processed in different ways to determine the best approach for the detection of SSH anomalous connections.

In this contribution, section 2 presents the approaches under study, applied to the SSH data described in section 3. Experimental results are described in section 4, while some conclusions and lines of future work are introduced in section 5.

2. Proposed Approaches

Many different formulae could be applied for the detection of SSH-based attacks. Present study analyses some of them, based on three main stages:

1. Data collection: network data may be summarized in several different ways. In this work, Honeynet data is proposed to be collected at packet-level and as TCP flows.
2. Data analysis: several different classifiers and classifier ensembles are proposed as different combinations for the modeling of SSH connections.
3. Result evaluation: there exists different cross-validation schemes to check the significance of supervised classification results. In this work, 10 *K*-fold and 5x2 cross-validation have been applied.

Further details of the proposed approaches for each one of these stages are described in following sections.

2.1 Data Collection

As previously mentioned, data from Euskalert honeynet regarding SSH sessions are targeted in present work. Those data are proposed to be analysed in two different ways.

Firstly, data is collected at packet level. It means that the values in the headers at different layers are extracted for further analysis. Table 1 shows the features considered for every single packet targeting the honeynet address pool and SSH port.

Table 1. Collected features for SSH packets.

Feature	Description
Src	IP address of the source host
Timestamp	Daytime when the packet was sent
Size	Size (number of bytes) of the packet

Numflags	Amount of different flags used

Then, the traffic has been processed in order to obtain the SSH sessions out of the packets. The approach for defining an SSH session was based on the TCP logic, using packets with the same source IP, same destination IP and a common source and destination port. Source port is a non-privileged port number that remains the same during any TCP session. The features that were extracted from each one of the sessions in the dataset are described in Table 2.

Table 2. Collected features for SSH sessions.

Feature	Description
Src	IP address of the source host
Length	Duration of the session
Numpac	Number of packets that the source host sent
Minlen	Minimum size of the packets
Maxlen	Maximum size of the packets
Size	Average size of the packets
Numflags	Amount of different flags used

2.2 Data Analysis

One of the most interesting features of IDSs would be their capability to automatically detect whether a portion of the traffic circulating the network is an attack or normal traffic. This task is more challenging when confronting brand-new bad intentioned activities with no previous samples.

Automated learning models (classifiers) [26] are well-known algorithms designed specifically for the purpose of deciding about previously-unseen data. This issue makes them suitable for the IDS task. Going one step further, ensemble methods [27] combine multiple algorithms into one usually more accurate than the best of its components. So, the main idea behind ensemble learning is taking advantage of classification algorithms diversity to face more complex data [28]. For this reason, present study proposes the combination of classifiers to get more accurate results when detecting anomalous and intrusive events.

A wide variety of automated learning techniques have been applied in this study to classify SSH connections. Several base classifiers as well as different ways of combining them have been considered for the analysis of Euskalert data. These base classifiers have been combined according to the ensemble paradigm by the following strategies: Bagging, Boosting, Adaboost, MultiboostingAB, and Rotation Forest.

2.3 Validation Schemes

For a more comprehensive study, two different cross-validation (CV) schemes have been applied in present work, namely 10 K-fold and 5x2 [29]. They are applied in order to check how suitable are the learning models for the addressed problem (detection of SSH malicious connections). In case that the best model is found for both CV schemes when applied to a certain data arrangement, it could be said that the classifier and the ensemble are clearly the best ones.

To fulfill the requirements of the intrusion detection task, a two-stage process is proposed, as depicted in Fig. 1. The first stage was already discussed in [30] and was carried out by using the features of the

SSH sessions to test the classification performance of different classifiers and ensembles, by means of Weka software [31]. The best combination (with a classification rate of 100%) of a base classifier and an ensemble was obtained by combining the Decision Table classifier and the Adaboost ensemble. Nevertheless, for real-life SSH ID it is important to take into account that the amount of training data will always be smaller than the amount of data gathered when working on a real context. As a consequence, the selection of data analysis models (both base classifier and ensemble) should be assessed with a more realistic validation scheme. Thereby, a second stage has been accomplished to assure that the chosen classifier may be applied in a real situation, where there are usually more samples for testing than for training. In the second stage, proposed in the present paper, the base classifier "Decision table" is applied. This classifier has been trained again in combination with 5 different ensembles for verifying that it is suitable for real data. For this new training, a 5x2 CV scheme has being used. The new ensembles and validation schemas have been developed specifically.

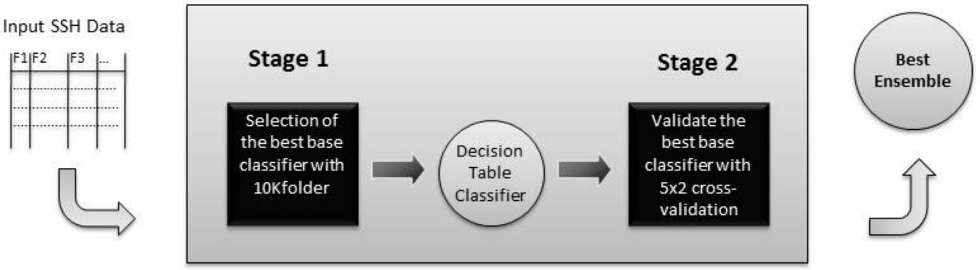
Input SSH Data
F1
F2
F3
...
Stage 1
Selection of the best base classifier with 10Kfolder
Decision Table Classifier
Stage 2
Validate the best base classifier with 5x2 cross-validation
Best Ensemble

Figure 1. The proposed two-stage method of validation.

3. Experimental Validation on Real Data

The performance of the proposed approaches has been assessed using real datasets, coming from the Euskalert project [5]. Detailed information about the data and the run experiments is provided in this section.

The experimental study has been performed by extracting SSH data related to 34 months of real attacks and administration tasks that reached the 8 sensors of the Euskalert project. Data from a so long time period guarantees that a broad variety of situations are considered. This honeynet system receives 4,000 packets a month on average.

The complete dataset from Euskalert contains a total of 2,647,074 packets, including TCP, UDP and ICMP traffic received by the distributed honeypot sensors. For this experiment, SSH data from connections happened between May 2008 and March 2011 have been selected. Additionally, traffic containing real attacks to the SSH port, and SSH connections to the system management port have been filtered out. This way, system management traffic is considered as benign traffic, and SSH connections coming from unknown IP sources are considered as malicious.

As stated in section 2.1, two different datasets have been generated: the first dataset contains the packet-level data, and amounts to 209 administration (legitimate) packets and 37,990 anomalous packets. Table 3 shows the range of each selected feature (described in Table 1), depending on the nature of the session (administration or anomalous).

Table 3. Range of features for SSH packets.

Feature	Type	Anomalous	Administration

Src	inet	---	---
Timestamp	time	00:00:02 – 23:58:20	00:02:40 – 23:09:37
Size	integer	40 – 220	40 - 380
Numflags	integer	2 – 194	2 - 24

Out of the packets, the session-based dataset was also extracted, containing 82 administration sessions and 8,477 anomalous sessions. Table 2 shows the range of each feature, depending on the nature of the session (administrator or attack).

Table 4. Range of features for SSH sessions.

Feature	**Type**	**Anomalous**	**Administration**
Src	inet	---	---
Time	interval	00:00:00 – 352 days 09:48:19.891	00:00:00.004 – 519 days 18:24:05.446
Numpac	integer	1 – 95	1 - 23
Minlen	integer	40 – 64	40 - 380
Maxlen	integer	40 – 220	40 - 380
Avglen	numeric(8,2)	40 – 96	40 - 380
Numflags	integer	1 – 6	1 - 4

4. Results

This section presents the results obtained by the different approaches proposed in present paper. For clarity and brevity, the average classification rate (%) for each ensemble applied to the *n* folders is provided, together with the maximum value and the standard deviation in Tables 5 and 6. Figures 2 and 3 show the boxplots associated to the classification rates of the different ensembles when applying the two alternative validation schemes. For independent analysis, classification rates for both packet-based and TCP-sessions datasets are presented. Results are then discussed for each one of the stages to which alternatives are proposed.

Table 5. Classification results from ensembles on packet-level dataset.

		10 K-fold			5x2 CV		
#	Ensemble	Average	Max	Deviation	Average	Max	Deviation
1	Bagging	99.93	99.97	0.000474790	99.92	99.95	0.000158140
2	Boosting	99.83	99.97	0.000981427	99.79	99.87	0.000550786
3	Adaboost	99.83	99.95	0.000792667	99.78	99.90	0.000640341
4	MultiboostingAB	99.84	99.92	0.00646569	99.85	99.87	0.00022946
5	Rotation Forest	99.91	99.97	0.000413983	99.85	99.88	0.00017873
-	Average	99.867	99.958	0.000661887	99.838	99.892	0.00035149

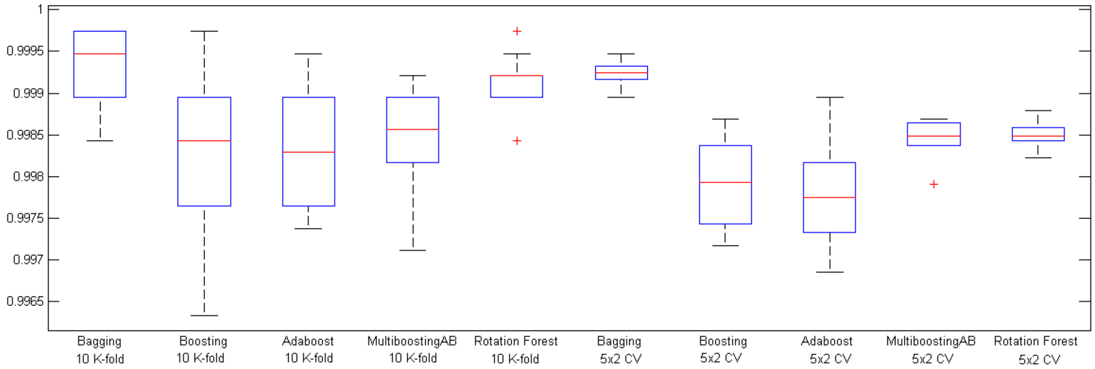

Figure 2. Boxplot results from ensembles on packet-level dataset.

Table 6. Classification results from ensembles on session-based dataset.

#	Ensemble	10 K-fold			5x2 CV		
		Average	Max	Deviation	Average	Max	Deviation
1	Bagging	99.82	100	0.00099298	99.74	99.84	0.0008003
2	Boosting	99.61	100	0.00270065	99.07	99.58	0.0035945
3	Adaboost	99.47	99.88	0.0022871	99.18	99.46	0.0029858
4	MultiboostingAB	99.46	99.65	0.00157671	99.32	99.51	0.0016225
5	Rotation Forest	99.72	100	0.00228562	99.58	99.70	0.0012137
-	Average	99.616	99.907	0.001968612	99.378	99.617	0.00204341

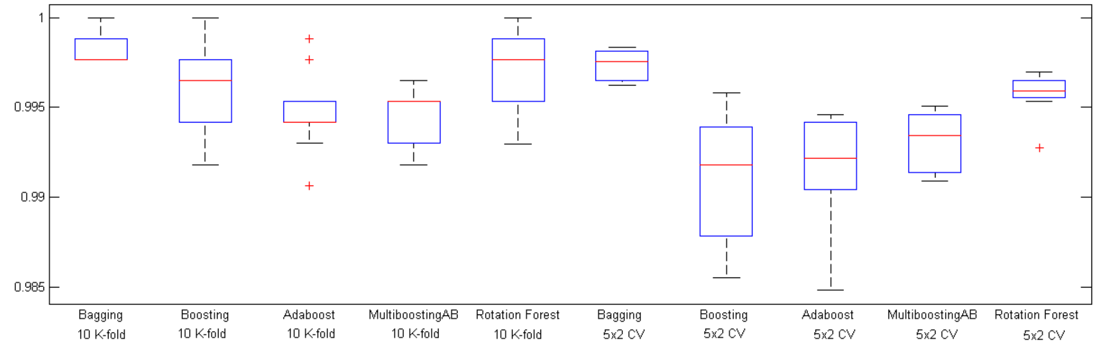

Figure 3. Boxplot results from ensembles on session-based dataset.

As can be seen in Tables 5 and 6, the best-performance classifierin the different proposed alternatives is Bagging. Thus, table 7 shows the confusion matrices for such ensemble. The amounts in this table are the cumulative results, according to the validation schemes (10 K and 5x2). In the case of the 10-K folder scheme, data in confusion matrices are the sum of the results for the 10 folders. As each one of then applies to 10% of the data, the total amount of data is the size of the dataset (10 times 10%), that is 38,199 packets and 8,559 sessions. On the other hand, for the 5x2 validation scheme, each folder comprises 50% of the data and there are 10 folders as well. Thus, matrices in the right side of Table 7 are related to 190,995 packets and 42,795 (10 times 50%).

Table 7. Confusion matrix for the best results – Bagging ensemble.

			10 K-fold			5x2 CV		
			Output class			Output class		
			Admin	Attack		Admin	Attack	
SSH packets	Target class	Admin	198 (00.518%)	11 (00.028%)	(00.547%)	951 (00.498%)	94 (00.049%)	(00.547%)
		Attack	17 (00.044%)	37973 (99.408%)	(99.453%)	53 (00.028%)	189897 (99.425%)	(99.453%)
			(00.562%)	(99.438%)		(00.526%)	(99.474%)	
SSH sessions	Target class	Admin	68 (00.794%)	14 (00.164%)	(00.958%)	322 (00.753%)	88 (00.206%)	(00.009%)
		Attack	1 (00.012%)	8476 (99.030%)	(99.042%)	24 (00.056%)	42361 (98.985%)	(99.041%)
			(00.806%)	(99.194%)		(00.808%)	(99.192%)	

One of the main issues to be highlighted from Tables 5 and 6 is that Bagging is the ensemble attaining the best performance in all cases (different datasets and validation schemes). Its average classification rate (for the 10 classifiers) is always superior to that of other ensembles. Additionally, the best classification rate (Max) for a single classifier is always one of the highest for such ensemble.

According to the data collection issue, it can be seen that classification results are on average (for the different ensembles) higher in the case of packet-based SSH dataset. Furthermore, the maximum classification rates are also obtained for this dataset, despite the fact that it reaches 100% for some of the base classifiers combined with certain ensembles on session dataset.

Through the evaluation by 10 K vs. 5x2, as shown inTables 5 and 6, it can be said that 5x2 is more strict as the classification rates are always lower than in the case of 10 K-fold (except for the Multi-boostingAB when applied to packet dataset). On the other hand, for the best results (Table 7) it can be said that False Negative Rate is lower for 5x2 in the case of SSH packet dataset.

From Table 7 it can be said that False Positive Rates (FPR) and False Negative Rates (FNR) are good enough to validate the proposed stages of SSH ID. The FNR reaches its lowest value (00.012%) when validating the SSH session dataset by 10 K, as only 1 attack session is misclassified. In the case of packet-based dataset, the FNR is lower for the 5x2 CV scheme. On the other hand, the FPR takes its lowest value (00.028%) when validating the SSH packet dataset by 10 K.

5. Final Conclusions

As The Secure Shell Protocol is mainly used for administration purposes, and hence it manages critical information, it is potentially a dangerous protocol. In present paper, several strategies and approaches are compared to validate its performance on the detection of SSH-based attacks.

The following main conclusions can be drawn from the proposed alternatives at the different ID stages, according to the experimental results in previous section:

1. Data collection: by gathering data at the packet level, obtained classification results are better (on average) than those from session-based data.
2. Data analysis: Bagging is the ensemble attaining the best performance in all cases.
3. Result evaluation: 5x2 is more strict than 10 K-fold.

The promising classification results obtained by ensemble classifiers in present study could be applied to other network protocols and services. Mainly, it would be interesting its application to the attacks received by the honeynets, such as those based on HTTP, SNMTP, or even FTP, learning from the honeypots classification models that will later prevent detected attacks surpass the organization networks causing any damage.

Acknowledgments

This research is partially supported through projects of the Spanish Ministry of Economy and Competitiveness with ref: TIN2010-21272-C02-01 (funded by the European Regional Development Fund), and Junta de Castilla y León projects SA405A12-2 and BIO/BU09/14.

References

1. SANS Institute's Internet Storm Center https://isc.sans.edu/port.html?port=22

2. Computer Security Threat Monitoring and Surveillance. Technical Report. James P. Anderson Co (1980)

3. Denning, D.E.: An Intrusion-Detection Model. IEEE Transactions on Software Engineering 13 (1987) 222-232

4. Chih-Fong, T., Yu-Feng, H., Chia-Ying, L., Wei-Yang, L.: Intrusion Detection by Machine Learning: A Review. Expert Systems with Applications 36 (2009) 11994-12000

5. Euskalert, Basque Honeypot Network, http://www.euskalert.net

6. Charles, K.A.: Decoy Systems: A New Player in Network Security and Computer Incident Response. International Journal of Digital Evidence 2 (2004)

7. Provos, N.: A Virtual Honeypot Framework. 13th USENIX Security Symposium, Vol. 132 (2004)

8. Baecher, P., Koetter, M., Holz, T., Dornseif, M., Freiling, F.: The Nepenthes Platform: An Efficient Approach to Collect Malware. 9th International Symposium on Recent Advances in Intrusion Detection (RAID 2006), Vol. 4219. Springer Berlin / Heidelberg (2006) 165-184

9. Moore, D., Shannon, C., Brown, D.J., Voelker, G.M., Savage, S.: Inferring Internet Denial-of-service Activity. ACM Transactions on Computer Systems 24 (2006) 115-139

10. Herrero, Á., Zurutuza, U., Corchado, E.: A Neural-Visualization IDS for Honeynet Data. International Journal of Neural Systems 22 (2012) 1-18

11. Friedman, J.H., Tukey, J.W.: A Projection Pursuit Algorithm for Exploratory Data-Analysis. IEEE Transactions on Computers 23 (1974) 881-890

12. Abraham, A., Grosan, C., Martin-Vide, C.: Evolutionary Design of Intrusion Detection Programs. International Journal of Network Security 4 (2007) 328-339

13. Julisch, K.: Data Mining for Intrusion Detection: A Critical Review. In: Barbará, D., Jajodia, S. (eds.): Applications of Data Mining in Computer Security. Kluwer Academic Publishers (2002) 33-62

14. Giacinto, G., Roli, F., Didaci, L.: Fusion of Multiple Classifiers for Intrusion Detection in Computer Networks. Pattern Recognition Letters 24 (2003) 1795-1803

15. Chebrolu, S., Abraham, A., Thomas, J.P.: Feature Deduction and Ensemble Design of Intrusion Detection Systems. Computers & Security 24 (2005) 295-307

16. Kim, H.K., Im, K.H., Park, S.C.: DSS for Computer Security Incident Response Applying CBR and Collaborative Response. Expert Systems with Applications 37 (2010) 852-870

17. Tajbakhsh, A., Rahmati, M., Mirzaei, A.: Intrusion Detection using Fuzzy Association Rules. Applied Soft Computing 9 (2009) 462-469

18. Sarasamma, S.T., Zhu, Q.M.A., Huff, J.: Hierarchical Kohonenen Net for Anomaly Detection in Network Security. IEEE Transactions on Systems Man and Cybernetics, Part B 35 (2005) 302-312

19. Herrero, Á., Corchado, E., Gastaldo, P., Zunino, R.: Neural Projection Techniques for the Visual Inspection of Network Traffic. Neurocomputing 72 (2009) 3649-3658

20. Zhang, C., Jiang, J., Kamel, M.: Intrusion Detection using Hierarchical Neural Networks. Pattern Recognition Letters 26 (2005) 779-791

21. Marchette, D.J.: Computer Intrusion Detection and Network Monitoring: A Statistical Viewpoint. Springer-Verlag New York, Inc. (2001)

22. Roesch, M.: Snort–Lightweight Intrusion Detection for Networks. 13th Systems Administration Conference (LISA '99) (1999) 229-238

23. Song, D.X., Wagner, D., Tian, X.: Timing Analysis of Keystrokes and Timing Attacks on SSH. Proceedings of the 10th conference on USENIX Security Symposium, Vol. 10. USENIX Association, Washington, D.C. (2001) 25-25

24. Coster, D.D., Woutersen, D.: Beyond the SSH Brute Force Attacks. 10th GOVCERT.NL Symposium (2011)

25. Koniaris, I., Papadimitriou, G., Nicopolitidis, P.: Analysis and Visualization of SSH Attacks Using Honeypots. IEEE European Conference on Computer as a Tool (IEEE EUROCON 2013) (2013)

26. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2007)

27. Seni, G., Elder, J.: Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions. Morgan and Claypool Publishers (2010)

28. Sedano, J., Berzosa, A., Villar, J.R., Corchado, E., de la Cal, E.: Optimising Operational Costs using Soft Computing Techniques. Integrated Computer-Aided Engineering 18 (2011) 313-325

29. Villar, J., González, S., Sedano, J., Corchado, E., Puigpinós, L., de Ciurana, J.: Meta-heuristic Improvements Applied for Steel Sheet Incremental Cold Shaping. Memetic Comp. 4 (2012) 249-261

30. González, S., Sedano, J., Zurutuza, U., Ezpeleta, E., Martínez, D., Herrero, Á., Corchado, E.: Classification of SSH Anomalous Connections. In: Herrero, Á., Baruque, B., Klett, F., Abraham, A., Snášel, V., Carvalho, A.C.P.L.F., Bringas, P.G., Zelinka, I., Quintián, H., Corchado, E. (eds.): International Joint Conference SOCO'13-CISIS'13-ICEUTE'13, Vol. 239. Springer International Publishing (2014) 479-488

31. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. ACM SIGKDD Explorations Newsletter 11 (2009) 10-18