ANNALES UMCS INFORMATICA DOI: 10.17951/AI.2016.16.2.20

New architecture of system intrusion detection and prevention

Mariusz Nycz
Faculty of Electrical and Computer Engineering
Rzeszow University of Technology
Rzeszow, Poland
mnycz@prz.edu.pl

Alicja Gerka
Rzeszow University of Technology
Rzeszow, Poland
137406@stud.prz.edu.pl

Mirosław Hajder
University of Information Technology and Management in Rzeszow
Rzeszow, Poland
miroslaw.hajder@gmail.com

***Abstract*— In this article there has been presented new intrusion detection and prevention algorithm implemented on Raspberry Pi platform. The paper begins with the presentation of research methodology in the field of Intrusion Detection Systems. Adequate supervision and control over network traffic is crucial for the security of information and communication technology. As a result of the limited budget allocated for the IT infrastructure of small businesses and the high price of dedicated solutions, many companies do not use mentioned systems. Therefore, in this order, there has been proposed monitoring solution based on the generally available Raspberry Pi platform. The paper is addressed to network administrators.**

***Keywords*— intrusion detection; network security; IDS; IPS**

I. THE DEFINITION OF TASKS AND RESEARCH AREA

Although the appearance of Intrusion Detection Systems (IDS) dates back to 1980, a crucial role in their development was played by the D. E. Denning's article "An intrusion-detection model" [1]. According to the most commonly quoted definition, IDS's are hardware and software systems that collect and analyze information from various points of the information system or computer network in order to detect and identify both attempted and actual violations of the protection system [2], [3], [4]. Contemporary IDS's are composed of three basic elements: the subsystem of information gathering, the detection subsystem, and the data presentation module [5]. The information gathering subsystem is used to collect initial information about the work of the protected system. The detection subsystem, based on a variety of analyses, looks for attacks and intrusions to protected information system. The data presentation subsystem, which is essentially a man-machine interface, provides the user the ability to monitor the IDS protected system status.

Both from an engineering as well as scientific point of view, a key component of the IDS is a detection subsystem consisting of one or more analysis modules. Using multiple analyzers increases detection efficiency. The data source for each analyzer is an information gathering subsystem that uses a set of sensors built into the operating system, furthermore there can also be used external sources of in-formation. In the current IDS, in order to detect potential threats there are used two alternative directions of system testing. The first one is directed to the detection of anomalies in the functioning of the protected system, while the other is focused on abusive behavior detection in functioning of the system. Each of these methods has its advantages and disadvantages, and in practice, most of the solutions are based on both approaches. The idea of the first method is based on whether a process that has caused harmful changes in the functioning of the system is the result of an intruder's actions. The second of the methods used is looking for a sequence of events that occur in the event of an intrusion.

As recommended by literature [6], [7], [8], [9], [10], [11], the protection of the information system should be multi-level. This means that the IDS should not be the only protection subsystem. Just like any other method of providing security, IDS has its advantages and disadvantages. The disadvantages should be addressed first are:

1. *Limited effectiveness*. An attempt to create a system that detects any anomalies in the functioning of the system always leads to distortion of the essence of the IDS. Note that each detected type of interference may require the use of its own analyzer consuming some resources during operation. In turn, the appearance of attacks is determined on the basis of the finding of more than standard consumption of resources. It is therefore likely that the IDS will diagnose an increase in resource use as an attempted attack. The set of IDS rules, which allows only indirect relationships between events makes it difficult to filter those situations.
2. *Unsatisfactory update options*. The IDS components are all subject to the moral aging. While the hardware components aging takes place relatively slowly, it is a very rapid

process in relation to software components. In particular, it concerns the methods of detecting the threats that underlie the analyzers construction. The solutions applied by the authors in system concerned allow for flexible replacement of software components with newer versions or enhancement of detection techniques with methods implemented by the user himself. Also, the use of a widely available hardware platform simplifies the IDS upgrade, especially in cases of application of new sensors.

3. *Moderate portability*. Most of current IDS are dedicated to a particular information system, and their movement to another system may be difficult. This is due to the dedication of applied solutions for specific hardware and software platforms – IDS has based its functionality on hardware and software system resources. To avoid this, the authors decided to build an external IDS based on its own software and hardware re-sources. Theoretically, the migration process should be seamless. In practice, however, many complications arise. Note that switching from a system with single-level system access list to a system with multi-level list is relatively complicated and requires substantial modification.
4. *The lack of intuitiveness in installation and operation*. IDS are dynamically changing products significantly different from the other applications that improve the security of the information system.
5. *Unknown performance*. The evaluation of the IDS operation effectiveness in real conditions is a complex and insufficiently-described in the literature task, with no clear, universally accepted evaluation criteria. In the proposed solution, the effectiveness depends on the external component of the protected information system, so the determination of the IDS effectiveness should be easier.
6. *The lack of a universally accepted design methodology*. Analysis of the publications from a given area illustrates discretion in the area of IDS design and implementation. Various architectures are used, there is a terminological confusion. Focusing on the construction of external IDS based on the Raspberry PI platform, proposed by the authors, in particular the satisfactory results, is partially ordering chaos in this area.
7. *The limitation of the testing techniques*.

In this publication the authors present the results of their work on the construction and implementation of a low-cost hardware and software platform for detection attacks on information systems. The result is a (described further) platform, based on the original software, operating on hardware resources of Raspberry Pi. In order to eliminate some of the above mentioned disadvantages, it was decided to use a lowcost hardware platform and four parallel analyzers operating on the basis of diametrically different methods. By using Raspberry for the platform construction, its low price allows for wide range of applications. The architecture of the system is shown in Fig. 1.

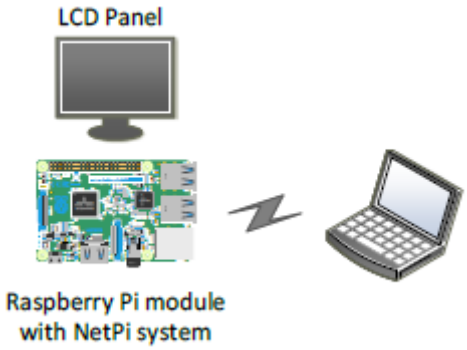

Fig. 1The system architecture

II. The Functional Architecture Of The Detection Systems

Information systems are now being used extensively in all areas of socioeconomic human activity. Their definition and description are contained in many source materials [12], [13], [14]. Also, the protection of systems and the information contained therein is a task extensively described in the literature [6], [8], [15]. From a functional point of view, the information system consists of five basic interrelated subsystems:

1. *Information processing and storage subsystem* providing the collection, processing, storage and sharing of information;
2. *Archiving subsystem protecting the information system* from loss of accumulated information;
3. *Information system status monitoring subsystem* ensuring permanent verification of the correctness of system's functioning;
4. *Information system management subsystem*, which is responsible for controlling information system components.
5. *Information system protection subsystem* responsible for protecting the system against attacks in accordance with the implemented security policy.

In practice, the composition of the protection subsystem may vary depending on the specific implementation, type of business and organizational and legal form of the organization and the relevant security requirements of normative documents created by regulators.

Structurally, intrusion detection systems consist of the following components:

1) Knowledge acquisition subsystem that collects information about the work of a protected information system. It bases its action on a set of agents analyzing various aspects of the functioning of the protected system. Increasing the number of agents and the types of information they analyze enhances the efficiency of attack detection. The size of the set of agents is bounded below by the minimum acceptable level of intrusion attempt detection and bounded above by the costs and number of tracked processes occurring in the system. For medium-size systems this number is in the range from several to a

dozen. Information about potential attacks goes not only to the expert system described below, but also to the knowledge creation subsystem and the decision maker involved in the database content creation process.

2) Attack detection expert subsystem. In the case at hand, it bases its action on the mathematical analysis of data about events occurring in the system, provided by the agents described above and the knowledge of the characteristic symptoms contained in the symptom database.
3) Signaling subsystem providing simple communication with the user or sys-tem administrator when an attack is detected. This element is built in accordance with the principles defined by the theory of human-machine interaction.

The interconnection of the components listed above, forming a fully functional attack detection system is shown in Fig. 2.

All components of the model below were made on the basis of elements of the Raspberry PI system. The system operates in two basic modes:

Learning mode, in which the database of symptoms is created. It contains records that associate a specific type of attack with a motion parameter vector composed of the values indicated by the measurement sensors. In creating the database of symptoms also participates the Decision maker, which classifies symptoms in cases where a decision cannot be made automatically.

Attack detection mode, when the values of the motion parameters specified by the measurement sensors are applied directly to the input of the expert subsystem. On the basis of the predefined analysis, the expert subsystem determines whether the analyzed network is operating in normal mode or is subject to attacks. In the latter case, the signaling subsystem is started.

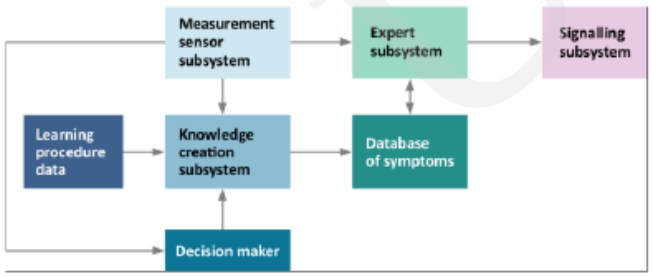

Fig. 2 An attack detection system model

III. The Functioning Of The Detection System

The main task facing any attack detection system designer is to build a knowledge base that allows efficiently identify attacks. Detection should be based on frame analysis of the protocol used, in this case IEEE 802.11. The framework of the named protocol has been described in detail in the literature.

The algorithm starts with the creation of the matrices, in which the captured F frames are stored, the P parameters separated from them, and the PW matrix containing the model parameter values obtained in the learning process of the identification of attacks. In addition, there are created: T matrix of network activity statistics in the time interval Δt and M matrix containing the list of access points of the wireless network. Furthermore, in the MW matrix there is prepared information about trusted access nodes containing its physical address, used channel number, and acceptable encryption and authentication protocols.

In the next step of the algorithm, data from the network sensors that describes the current traffic are collected. They are subjected to a preliminary analysis aimed at determining motion parameters stored in the F matrix. Then, event tag vectors are prepared and sequentially added to P matrix. Their values are normalized and passed to the input of intrusion detection module's classifiers. Another block, based on a prepared model containing a set of decision rules classifies security events. Classification is based on comparing the values of the symptoms with the model values in the table PW. If, as a result of the comparison, the value of the symptom vector corresponds to the value of any harmful activity, the input level of the L classifier signal is analyzed. If the Lpr threshold value has been reached or exceeded, the event information is passed to the decision module generating a corresponding message on the system operator's console. Further on the basis of prepared scenarios, there are prepared sequences of actions aimed at neutralizing the attack. For this purpose, protective measures are implemented on the system devices and, if possible, activities that directly affect the intruder's device are also carried out.

Assume that Lmin is the minimum signal value corresponding to the occurrence of forbidden network activity. Therefore, if the current L value of the classifier signal is in the range $Lmin \leq L < Lpr$, the analyzed event vector is placed in the A database of suspicious activity for further analysis based on other system tools. Upon reaching the threshold value of identical events, a query suggesting the presence of malicious activity in recorded events is sent to the administrator console. Depending on the response of the administrator, the classification model is changed (or not) through training. In addition, the S signature database can be supplemented with additional records about events.

These actions allow to effectively detect most types of attacks. In order to increase the effectiveness of the system in the event of certain types of attacks appear in T table, the statistics of the number of frames of the specified type appearing in the given time interval Δt with the same values of the key parameters (e.g. sender's address, recipient's address, frame type) is currently being performed, as well as higher-level protocol types from the predefined set. In addition, during system operation, the M table which contains information about active wireless access points is updated regularly.

One of the principles on which the proposed system works is based on, assumes that one of the real scenarios (for example, for DoS attacks) will be sending a large number of frames of the same type. Therefore, the information about the number of K identical frames appearing on the protected interface is gathered using a set of sensors. If $K > Kpr$, the administrator receives information about the probable DoS attack in order to determine and remove its source. If the data from the M-set does not match the modeling data MW, there is automatically

generated message about fake device that has appeared in the network.

The key task solved during system design is to create an effective classification model. This model is extremely difficult to define in one empirical or analytical step. Therefore, the authors proposed for this purpose a group of alternative methods based on different, distant thematic areas of computer science. The following methods were used for this purpose: the Support Vector method, the k-Nearest Neighbors method, the neural network method and the decision tree method. These methods have been discussed, among others, in [9].

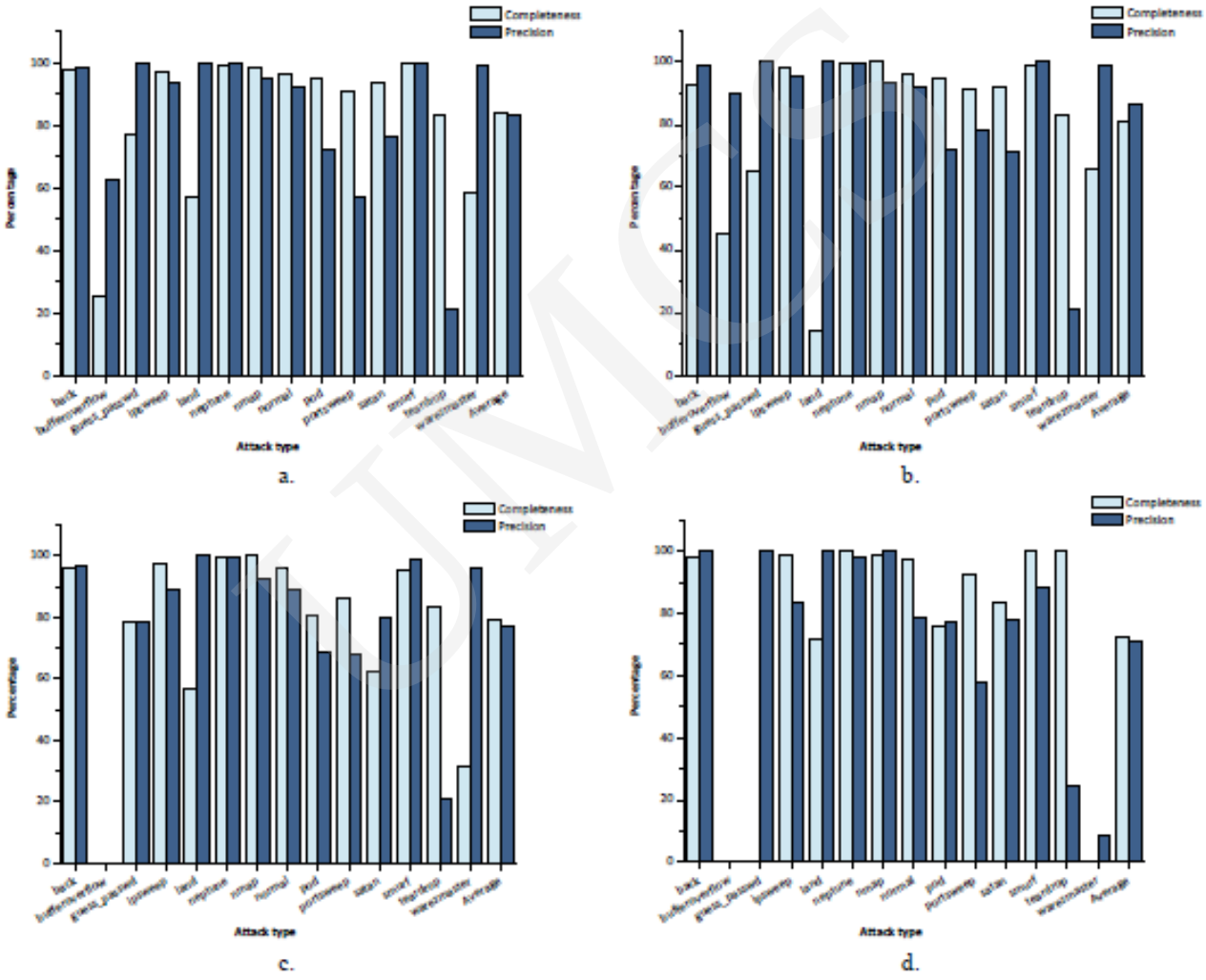

Fig. 3 The effectiveness of the attack detection system: a. The Support Vector method, b. The k-Nearest Neighbors method, c. The neural network method, d. The decision tree method

IV. The Research Experiment

RapidMiner environment and the software-hardware prototype of the designed device were used to conduct the research experiment. It consisted of two steps. In the first one, the procedure of filling the database of symptoms with attack patterns was completed. During its execution, there has been performed the normalization of numeric parameter values and the conversion of text parameters to binary values, as well as the classification of database records. In the second step, using the software-based test sequence generator, which imitates sensor output, has been determined the precision and completeness of the attack detection. The results of the experiment for various types of attacks are shown in Fig. 3.

V. Summary And Further Work

Ensuring information security is a key issue in ensuring the stability of the functioning of public and industrial institutions and protecting the identity of users of the global network. The successive development of the information society and the intensive growth of new services have increased the vulnerability of designed systems to threats of unauthorized access to information. In addition, the increase in data rates and the saturation of Internet services caused the transfer of very large datasets that forced the use of more computationally efficient threat analysis systems. Consequently, this requires high-cost solutions, which due to limited budgets may not be applicable to institutions or businesses. In most cases, the

available solutions are characterized by a template approach to threats detection, which can lead to rapid immunization and significantly reduce their effectiveness. That is why it is so important to develop methods for both threat analysis and implementation on low-cost platforms.

The results suggest that with a small amount of computing power it can be easy to detect the most common attacks in real network data. The proposed algorithms have a low computational complexity, which makes it possible to implement them on Raspberry PI devices.

Further authors' work will focus on widening the database of detected attacks and commercializing the solution.

REFERENCES

[1] E. Denning D., "An intrusion-detection model ," IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, vol. 13, no. 2, pp. 222-232, 1987.
[2] S. Axelsson, "Research in Intrusion-Detection Systems," Göteborg, 1998.
[3] H. Debar, M. Dacier, and A. Wespi, "Towards a Taxonomy of Intrusion-Detection Systems ," Rüschlikon, 1998.
[4] A. Lazarevic, V. Kumar, and J. Srivastava, Intrusion detection: A survey. Minneapolis: Computer Science Department, University of Minnesota, 2005.
[5] Z. Zhou, L. Liu, and G. Han, "Survival Continuity on Intrusion Detection System of Wireless Sensor Networks," in International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Kraków, 2015, pp. 775-779.
[6] D. W. Straub, S. Goodman, and R. L. Baskerville, Information Security. Policy, Process and Practices. London: M.E. Sharpe, 2008.
[7] H. F. Tipton and M. Krause, Information Security Management Handbook, 6th ed. Boca Raton: CRC Press, 2012.
[8] J. S. Tiller R. O'Hanley, Information Security Management Handbook. Boba Raton, USA: CRC Press, 2014.
[9] M. Hajder, P. Hajder, and M. Nycz, "Inteligentna analiza danych jako metoda detekcji ataków na sieci," in Innowacyjna gmina. Bezpieczeństwo i ekologia. Rzeszów: Wyższa Szkoła Informatyki i Zarządzania z siedzibą w Rzeszowie, 2013, pp. 7-25.
[10] M. Stamp, Information Security: Principles and Practice, 2nd ed. New York: Wiley & Sons, 2011.
[11] T. R. Peltier, Information security policies and procedures : a practitioner's references. Boca Raton: CRC Press, 1998.
[12] A. Laukaitis and O. Vasilecas, "Formal concept analysis and information systems modeling," in Proceedings of the 2007 international conference on Computer systems and technologies, Burgas, Bulgaria, 2007, pp. 1-6.
[13] G. Marakas and J. A. O'Brien, Introduction to Information Systems, 16th ed. New York, NY: McGraw-Hill, 2013.
[14] W. R. Bitman, "Information systems modeling: an object oriented development method," in Proceedings of the ninth Washington Ada symposium on Ada: Empowering software users and developers, McLean, Virginia, USA, 1992, pp. 93-105.
[15] R. Bejtlich, Practice Of Network Security Monitoring. San Francisco, USA: no starch press, 2013.