ISSN: 0974-5823 Vol. 7 No. 1 January, 2022

International Journal of Mechanical Engineering

Linguistic Processor (Simplified) For Decision Support In The Computer Networks Testing

Ademi Ospanova , Berik Tuleuov , Aizhan Zharkimbekova , Aizhan Tokkuliyeva , Bekzat Toksan

L. N. Gumilyev Eurasian National University

Abstract

An algorithm for intelligent processing of a limited language of a specialized program for testing the security of computer networks has been created. The algorithm filters the output of the program, analyzes the formal language of inference messages according to specific (technical) and linguistic criteria, generates interpreted messages without the use of specialized terms in the form of instructions, explanations and recommendations to ensure a timely response in urgent cases or to support decision-making in the process of testing and detection of vulnerabilities.

Clustering a selected set of program messages establishes a correspondence between the compiled database of the utility's messages and the levels of vulnerabilities and risks (meeting the recommendations of the standards). This provides the implementation of an automatic assessment of the security level of a computer network. In addition, the resulting clustered base is used to create a method for linguistic language processing.

To construct a (simplified) linguistic processor, the special concepts required to the formation of the knowledge base are introduced; examples are given. The processes in the operation of the presented algorithm are described in detail.

The algorithm was created for the purpose of implementation in the developed software and hardware package for testing the security of computer networks by users who are not specialized experts in the field of network protection and testing. The algorithm is applicable for both private computer networks and networks of small and medium-sized businesses.

Keywords: INTELLIGENT DATA PROCESSING, LINGUISTIC PROCESSOR, COMPUTER SECURITY, RISK ASSESSMENT, INTELLIGENT SYSTEM, RASPBERRY PI

1. Introduction

Under the grant of the project, completed by the authors in 2021, was developed an intelligent handheld network security testing device with recommendations for decision making. In the course of the project, an intelligent system with automatic risk assessment, report generation and recommendations was developed for a portable device based on a Raspberry Pi microcomputer, which allows users to perform initial or standard computer network security tests for non-specialists in this field and also without specific skills to work with applications that have a command line interface. For this, an intelligent data processing algorithm has been developed that analyzes messages from the formal language of the specialized utility Nmap ([1]) according to certain (technical) and linguistic criteria, performing an assessment of vulnerabilities and information risks and forming explanations, recommendations and instructions to ensure a timely response in case of vulnerabilities and high risks. A brief description of some of the stages of this development is given in ([2]).

The work carried out within the framework of the project can be attributed to the field of application of artificial intelligence methods in information security tasks, namely in the tasks of ensuring the security of computer networks. One of the central places among the areas of information protection is cybersecurity, which is due to the inalienable use of the Internet by modern organizations. Separately, it is worth noting the increased importance of the problem caused by COVID-19 ([3-5]).

Works [6-10] are devoted to the consideration of solutions to information security problems using artificial intelligence methods: decision-making methods, machine learning. The works [11-16] are devoted to the study of the possibilities of constructing syntactic, semantic analyzers and linguistic processors, including for the Kazakh language ([11]), and their implementation in order to create an intellectual component of question-search information systems.

In the project described above, an important task was the application of methods for the development of linguistic processors and the implementation of the corresponding intellectual component in the software of devices for monitoring and testing the security of computer networks in order to provide a solution applicable to all users. In this paper, a formal limited language of utilities is considered, which will allow achieving better accuracy in the work of the morphological, syntactic and semantic components. In addition, the purpose of the development (for non-specialists) also determines the specifics of the developed linguistic processor. We also note that the research carried out naturally continues the work previously carried out by the authors ([17-20]).

2. Intellectual decision support for testing computer networks

2.1. Intelligent device concept

The authors considered the possibilities of creating and effective functioning of a network security testing system for small organizations and business enterprises of the SOHO class, mainly in Kazakhstan. The available data ([21]) show the existence of such a common problem as the inability to have highly qualified specialists in the field of network security testing on the staff. Organizations, as a rule, turn to outsourcing companies, and the problem of limited budget does not allow the implementation of measures to ensure the protection and monitoring of networks with sufficient frequency and at the proper level.

The problem outlined above was a prerequisite for setting the task of creating an inexpensive software and hardware device with a convenient interface, which automatically carries out the initial standard check of the enterprise network; performs some actions to correct or configure network elements, configurations and interfaces; issues recommendations and instructions in an "interpreted" language that is understandable to a non-specialist user. Note that the ability to carry out a full check has been implemented, and not only based on the Nmap utility, but also additional software modules have been implemented that provide important information for analyzing the network state.

The development stages for this project are briefly described in [2], as well as in [22]. This work describes the stage of creating an algorithm for intelligent processing of the results issued by the utility. The algorithm analyzes the formal language of these messages according to specific (technical) and linguistic criteria, and also generates instructions and explanations to ensure a timely response in the absence of specialists in the field or to support decision-making during testing and detection of vulnerabilities.

The set of selected Nmap messages is clustered according to the vulnerability and risk severity label, based on which an automatic risk assessment is implemented, as well as a knowledge base for the linguistic processor, consisting of specially crafted sentences and facts. The algorithm forms an interpreted message by processing certain data from the knowledge base. The limited language of the utility led to the idea of creating special phrasal templates, with the help of which semantic interpretation and morphological analysis are easier to implement.

In addition to the linguistic processor that processes utility messages, software modules have been developed that provide information for in-depth analysis of the state of networks (can be used by specialized specialists), as well as additional functionality.

2.2. Clustering utility messages by hazard identifier

Clustering of the selected set of Nmap messages has been performed. Based on the results obtained, messages are classified according to the criteria of the type of vulnerability, threats and the degree of possible damage. This, on the one hand, makes it possible to implement an automatic assessment of the security level of a computer network (denote it by $KB-$ [22]), which is used in the general algorithm for intelligent data processing, implemented in the work of the above-mentioned device ([2]). On the other hand, the results are needed to create a method for linguistic processing of messages in the utility's formal language, and this is the focus of this work.

Let us briefly describe the message selection procedure and cluster features. Based on the set $C = \{c_1, \dots, c_k\}$ of k selected Nmap commands, taking into account the presence of a set of parameters $P = \{p_1^i, \dots, p_{n_i}^i\}_{i=1}^k$ for their call (n_i is the number of parameters under consideration of i-th command; $i = 1, \dots, k$), we denote the selected set of messages by $M = \{m_1, \dots, m_{n_1}, m_{n_1+1}, \dots, m_{n_k}\}$ – table 1.

Table 1. Formation of clustering objects

Commands (set C)	Command parameters (set P)	Set of the utility M outputs
1	1	m_1
	2	m_2
	...	...
	n_1	m_{n_1}
...	...	...
k	$n_{k-1}+1$	$m_{n_{k-1}+1}$
	...	...
	n_k	m_{n_k}

The setM is clustered by the label of the number of severity levels. Four degrees of gradation were chosen, the corresponding clusters will be denoted byK_1, K_2, K_3, K_4. For signs of nearness are taken:

- the states of certain ports, distributed according to the degree of threat of damage;

- discovered software, distributed among the available vulnerabilities and exploits.

The resulting clustered base KB is given in ([22]). If objects (messages) in clusters are denoted by

$$K_i = \left\{m_{r_{q_i}}, \dots, m_{r_{p_i}}\right\}, \quad i = 1, 2, 3, 4, \qquad (1)$$

Then$\sum_{i=1}^{4}(r_{p_i} - r_{q_i} + 1) = n_k$,$\bigcup_{i=1}^{4} K_i = m_1, m_2, \dots, m_{n_k}$, the sets$K_i$ do not intersect. Thus, the sets shown in Table 2 represent the distribution of the set$m_1, m_2, \dots, m_{n_k}$ of utility messages into 4 disjoint clusters (full clustering was used).

Table 2. Clustered messages

Cluster	Cluster objects	Cluster	Cluster objects
K_1	$m_{r_{q_1}}, \dots, m_{r_{p_1}}$	K_3	$m_{r_{q_3}}, \dots, m_{r_{p_3}}$
K_2	$m_{r_{q_2}}, \dots, m_{r_{p_2}}$	K_4	$m_{r_{q_4}}, \dots, m_{r_{p_4}}$

Note that clusterscharacteristics meet the ISO/IEC 27005: 2018 standard ([23]).

2.3. Clustering (Simplified) linguistic processor

Special words and divisioninto classes

Key words are selected from the utility messages – the base of signal words. A signal word will mean one or more words in the message issued by the utility, which in one way or another determine the situation on the checked network section, as well as, possibly, this section itself (network node). The numbering of the set of signal words formed from the messages (1) written out in Table 2 is shown in Table 3.

Table 3. Designation of signal word groups in accordance with messages M

Messages (M set)	Groups set of signal words S	Signal words$\overline{s^i}$
m_1	$\overline{s^1}$	$s_1^1, \dots, s_j^1, \dots, s_{t_1}^1$
…	…	…
m_{n_1}	$\overline{s^{n_1}}$	$s_1^{n_1}, \dots, s_j^{n_1}, \dots, s_{t_{n_1}}^{n_1}$
…	…	…
m_{n_k}	$\overline{s^{n_k}}$	$s_1^{n_k}, \dots, s_j^{n_k}, \dots, s_{t_{n_k}}^{n_k}$

Here$S = \{\overline{s^i}\}_{i=1}^{n_k}$ is theset of signal words groups extracted from the corresponding messagesM. A group of signal words$\overline{s^i} = (s_1^i, \dots, s_j^i, \dots, s_{t_i}^i)$,$i = 1, \dots, n_k$, is extracted from the messagem_i; there can be several words in a group:$t_i \geq 1$.

Table 3 also demonstrates that the groups of signal words are also divided into 4 classes according to their correspondence to the messages of the setM from which these words were extracted. The database with data on such a division is denoted byA.

Many signal words can be attributed to the same type. For example, the port number; reserved words "open", "filtered", "closed"; versions or numbers of updates of one type of software; numbers of network interfaces, etc. Let us denote by n the number ofvarious types of signal words (or groups of signal words)$A_1, A_2, \dots, A_n$.The elements of the set$A_i = \{a_1, \dots, a_l\}$, respectively, are data of the same typei ($i = 1, \dots, n$),$n, l \in (1,2,\dots)$. For example, the numbers of various ports appearing in the utility messagesM are data of the typeA_i = «Networkport», elements of the set of this type can be$\{a_1, \dots, a_l\} = \{22, 443, 21,23, 139, 445\}$.

Let us further introduce sets of word-parameters,$B_1, B_2, \dots, B_s s \in (1,2,\dots)$,which are parameters of a soft call or some available data that can be used for a deeper analysis of the state of the tested network. For example, it can be information with data from previous network checks, data from configuration files, saved settings of a firewall, network equipment. These parameters and data can be used in the formation of intelligent messages at the output of the linguistic processor. When defining words-parameters in the work, they were also correlated to one of 4 levels of danger (in cases where it is permissible) - the basis of complianceB.

Table 4 shows examples of signal word typesA_i and parameter word typesB_j, $i \in (1, \dots, n)$,$j \in (1, \dots, s)$.

Table 4.Examples of indicator words and parameter words

No.	Type A_1	Type A_2	Type A_3	Type B_1	Type B_2	Type B_3	Type B_4
1	1	22	Open	Inspection date	Fixed network state	Yes	1
2	2	443	Closed	Dates of previous inspections	Saved network data	No	2
3	3	23	Filtered	...	...	Conditionally	3
4	...	139	...				4

Here, A_1 is the number of the network interface on the host; A_2 −port number; A_3 −the state of the tested network port; B_1 −test dates; B_2 −saved parameters of the current and past checks (there may also be parameters and configuration data that must be set for this network); B_3 −data on "allowed / denied IP"; B_4 −the hazard level established by clustering and subsequent classification.

Further, on the basis of messages in the cluster setsK_1, K_2, K_3, K_4, as well as taking into account the correspondingly selected groups of signal words, phrasal templatesPhT are created. Phrasal templates are groups of 2-4 sentences with indicator words designed to automatically insert other necessary words instead of them. The structure of the indicator word is a combination of a letter and a number that determine the corresponding group of signal words and its type or parameter. A letter is defined by the set $\{A_i\}_{i=1}^{n}$ of different types of signal word groups (ntypes in total) or the set $\{B_j\}_{j=1}^{s}$ of different types of parameter word groups (stypes in total). The number is determined by the corresponding element in the set$A_i = \{a_1, \dots, a_l\}$,$l \in (1,2,\dots)$ or in the set$B_j = \{b_1, \dots, b_v\}$,$v \in (1,2,\dots)$. Thus, instead of indicator words, groups frompre-compiled bases of signal words $\{A_i\}_{i=1}^{n}$ and parameter words $\{B_j\}_{j=1}^{s}$ are used. Table 5 below shows examples of phrase patterns and indicator words, the meanings of which can be deciphered using Table 4.

Table 5. Examples of phrasal patterns and construction of indicator words

No.	Phrase templates
1	On the network interface $A_1 1$, the port $A_2 3$ is on the state: $A_3 1$
2	In the last report $B_1 2$, the port $A_2 4$ was $A_3 3$
3	In the list of allowed Internet addresses: $B_3 1$
4	Danger level:$B_4 1$!
5	It is recommended to close the port $A_2 3$
6	Compare the parameters with the last check. The data $B_1 1$, $B_1 2$ and $B_2 1$, $B_2 2$ are compared

Depending on the values of the corresponding word-signals and word-parameters, which are divided into classes according to the basesA andB, phrasal templates of four groups are formed: $PhT_1, PhT_2, PhT_3, PhT_4$.

In addition, 4 small groups of phrasesPh_1, Ph_2, Ph_3, Ph_4 without indicator words have been developed, designed to form and supplement messages at the output of the linguistic processor with typical messages; the index corresponds to the level of danger. Examples of such phrases are shown in Table 6.

Table 6. Examples of group phrases Ph_1, Ph_2, Ph_3, Ph_4

Hazard level /Ph_i	**Examples of phrases**
1	Network connection is dangerous
2	Show the result of monitoring to a specialist
3	It is recommended to close the port
3	It is recommended to send a report to a specialist
1	It is recommended to immediately show the monitoring results to a specialist.
2	It is recommended to disconnect from the local and global network and contact specialists
1	An attacker can remotely control a computer and make changes to the file system
4	The routine check was successful. No deviations found

These phrase groups and phrase patterns are included in the linguistic processor knowledge baseKB.

About morphological analysis

The next implemented step is morphological processing of generated messages; let us denote this stage asMA. The task was carried out first for the Kazakh language. The bases of words-indicators and words-parameters were coordinated (table 4) in accordance with the semantic and morphological analysis of phrase patterns (table 5) of phrases of groupsPh_1, Ph_2, Ph_3, Ph_4 (table 6). The following three factors allow us to say that this step is not laborious, and explain the name of the described linguistic processor "simplified". First, operating with a limited set of processed sentences; secondly, the fact that signal words are determined from the messages of a limited natural language - the formal language of the utility; thirdly, the introduced phrase patterns with indicator words and word-signals – all this allowed to change the usual approach to the construction of an intelligent linguistic processing mechanism.

Description of the operation of the intelligent component of the device

Within the framework of the described project, 7 groups of scripts$\{SG_1, \dots, SG_7\}$ were prepared for a complete network check by various parameters, as well as a groupSG_8 for a regular / quick check. To start the linguistic processor, a prepared group of scriptsSG_i ($i \in (1, \dots, 8)$) is launched. The subset$M^* \subseteq M$, $M^* = \{m_1^*, \dots, m_{n_1^*}^*, m_{n_1^*+1}^*, \dots, m_{n_{k^*}^*}^*\}$ obtained as a result of the work is sent for linguistic processing. Based on the clustering performed earlier, the classification of the detected situations on the checked network section is performed. By means of the selected signal wordsS on the basis of the databaseA data, a correspondence is established between a certain message$m_i^* \in M^*, i = 1, \dots, n_{k^*}^*, n_{k^*}^* \leq n_k$ and a quantitative expression (from 1 to 4) of the degree of vulnerability in the vulnerability and risk assessment systemKB ([22]). In addition, on the basis of the running additional program modules, data is collected, on the basis of which the word-parameters are determined, examples of which are given in Table 4. Further, the word-signals and word-parameters are inserted into phrase templatesPhT, combined with the corresponding phrasesPh_1, Ph_2, Ph_3, Ph_4. The next step is the stage of morphological analysisMA and approval of proposals. The process described here is shown schematically in Figure 1.

In order to create a rapid prototype based on signal words, a base of explanatory words or phrases has been developed using generally accepted vocabulary without the use of highly specialized terms. This database of "interpreted" messages was developed in natural languages – Kazakh and Russian. Interpreted messages are natural language messages that explain the essence of specific utility messages in a general language style without the use of specialized terms. The base in the Kazakh is described in [22].

Within the framework of the project, not only a mechanism for generating messages (recommendations and instructions) was developed for the user of a hardware-software device-non-specialist in the field of computer networks, but also the generation and sending of reports on the testing performed, testing on a schedule, and other functionality were implemented.

Figure 1. Functional diagram of the intelligent component of the application

As part of the algorithmSW presented in Figure 2 for the operation of the entire application, the intelligent message processing algorithmIPL (described in Figure 1) is depicted as one of the alternative processesAP_1 andAP_2.

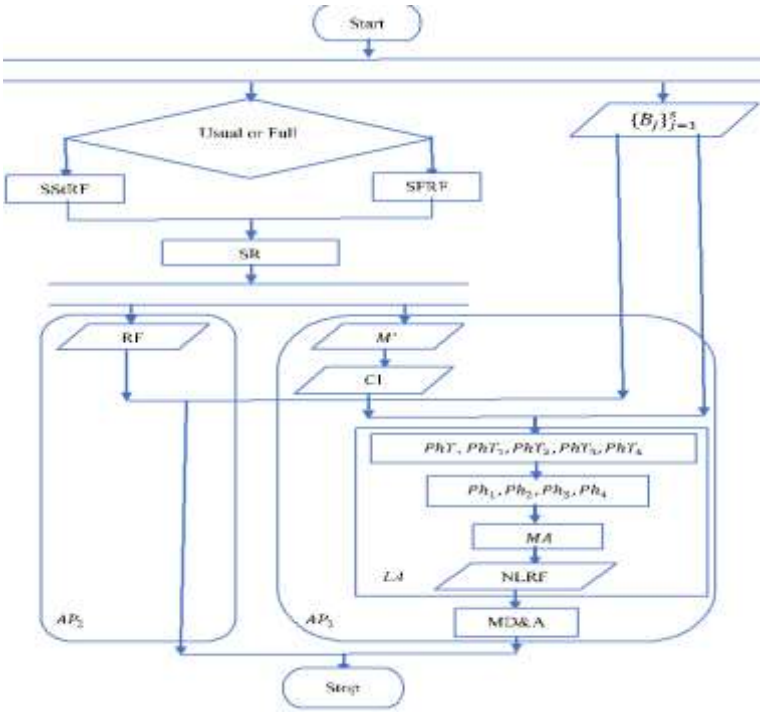

Figure 2. Block diagram ofNmap smart message processing algorithm IPLand SW algorithm of the application without taking into account additional functionality

We use next abbreviations in Figure 2: SStRF - formation of the standard scanning mode; SFRF - formation of a full scan mode; SR - scan mode; Cl - classification; NLRF - Natural Language Recommendations Generation; RF - reports generation; MD&A - Decision Making and Action.

Note that there is an MD&A process in this processAP_1, which is responsible for making decisions by the non-specialist user, as well as performing certain actions automatically. For example, these can be actions to change the settings and configurations of

the network in case of detection of obvious vulnerabilities or malicious scripts. Figure 2 also shows the process of working directly with the linguistic analyzerLA of the scanning results, it consists of applying word signals and word parameters to phrasal templates and prepared phrases and the subsequent agreement of sentences in the processMA.

The second alternative processAP_2 is generating reports based on the results of running Nmap scripts (without processing messages) and the results of the work of additional program modules. This process works independently of the first, and its results are intended for specialists in the field of computer network security testing. The application implements the ability to send reports on the results of the processAP_2.

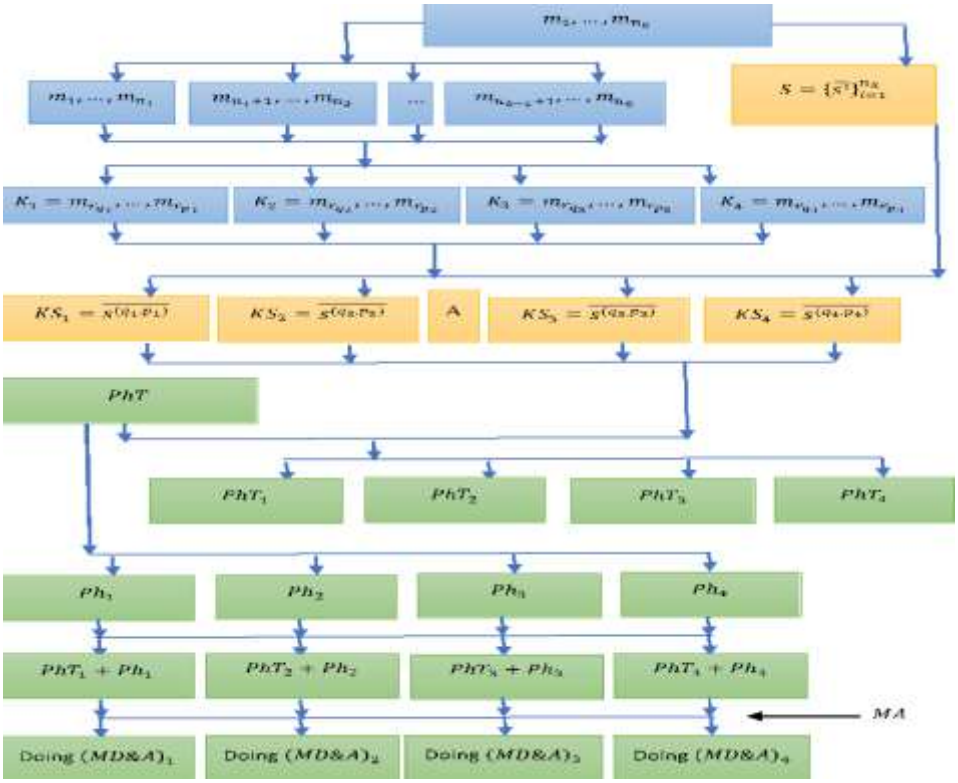

Figure 3. Stages of processing and transformation of messages in the IPL algorithm

The stages of language processing from receiptM^* to decision and action MD&A's in IPL are presented in Figure 3.

Here KS_1,KS_2,KS_3,KS_4–are, respectively, classes of groups of signal words that form the baseA. The sets$(MD\&A)_i$,$i = 1,2,3,4$, are formed from phrasal templates and prepared groups of phrases, and the corresponding process is the generation of recommendations and explanations, as well as the automatic execution of some of the actions described above.

Knowledge base and withdrawal rules

The linguistic processor knowledge baseKB includes groups of phrasesPh_i,$i = 1,2,3,4$, and phrasal templates$PhT = \{PhT_i\}_{i=1}^{4}$. Inference rules$IR$ in the constructed linguistic processor can be represented using first-order predicate logic. Some examples of resolution rules are given below.

Let's denote by$m \in M$ some selected Nmap message from the previously prepared database. By lev denote a cluster label – a hazard level that takes values1,2,3,4. By $r(lev)$denote the recommendations generated by the algorithmIPL. The correspondence between$m$ and$r(lev)$ imagine as a predicate relation$(m, r(lev))$. Then the ruleIR_1 for representing a message$m$ in the formal language of the utility in an adapted form can be written as follows:

$$P(m, r(lev)) : - (m \leftrightarrow lev) \wedge \bar{s}(m) \Rightarrow KB\ (lev, \bar{s}),$$

$$P(m, r(lev)) : - (m \leftrightarrow K_i) \wedge (\bar{s}(m)), KS_i, PhT(KS_i) \Rightarrow (PhT_i \wedge Ph_i).$$

Relationships for knowledge base components:

$$\mathrm{KB}\ (\mathrm{lev}, \bar{s}) : - \ \mathrm{PhT}(\mathrm{lev}, \bar{s}) \wedge \bigcup_{i=1}^{4} \mathrm{Ph_i}\ (\mathrm{lev}, \bar{s}),$$

$$\mathrm{KB} = \{\mathrm{PhT}, \mathrm{Ph_1}, \mathrm{Ph_2}, \mathrm{Ph_3}, \mathrm{Ph_4}; \mathrm{IR}\} = \{\mathrm{PhT_1}, \mathrm{PhT_2}, \mathrm{PhT_3}, \mathrm{PhT_4}, \mathrm{Ph_1}, \mathrm{Ph_2}, \mathrm{Ph_3}, \mathrm{Ph_4}; \mathrm{IR}\}.$$

3. Conclusion

A software package has been created that implements the developed algorithm with automatic decision support for users who are not specialists in the field of information security. In addition, the ability to monitor on a schedule, receive important notifications, and automatically generate and send reports are implemented ([22]).

The authors research([17-20], [2]) represents developed examples of the effective use of Raspberry Pi for various purposes: teaching certain disciplines, organizing a specialist's workplace, including testing the security of computer networks, as well as creating fault-tolerant secure systems for exchanging information and conducting knowledge slices. In this work, an urgent task is completed, the idea of which naturally grew on the basis of the described studies. Smart mobile hardware and software devices that are easy to use by lay users in the field of computer network security will advance the overall challenge of increasing digital literacy and awareness of the severity and magnitude of cyber threats. The potential for the commercialization of the results obtained can also be noted.

4. Acknowledgements

This research is funded by the Science Committee of the Ministry of Education and Science of the Republic of Kazakhstan (Grant No. AP09561712).

11. References

[1] Nmap Free Security Scanner. – URL: https://nmap.org/ (date of treatment 10/11/2021).

[2] A. B. Ospanova, B. I. Tuleuov, A. K. Tokkuliyeva, L. T. Kussepova, A. T. Zharkimbekova. Intelligent Mobile Hardware-software Device for Automated Testing and Monitoring of Computer Networks Based on Raspberry Pi. Proc. 15th Int. Conf. APEIE, (**2021**) Nov. 19-20; Novosibirsk, Russia.

[3] Information security forecasts for 2021. - URL:https://tinyurl.com/2pcbcp6n.

[4] Cyberattacks. – 2021. – URL:tadviser.ru/index.php/Article: Cyberattacks.

[5] Key Cybersecurity Trends for 2021. - URL:https://www.osp.ru/articles/2020/0802/13055968.

[6] J. N.Al-Karaki, A. Gawanmeh, I. T. Almalkawi, O. Alfandi. Probabilistic analysis of security attacks in cloud environment using hidden Markov models. Trans Emerging Tel Tech. 2020;e3915 (**2020**).

[7] H.Mezni, M.Sellami, J.Kouki. Security-aware SaaS placement using swarm intelligence. J Softw Evol Proc. 2018;1932 (**2018**).

[8] R. Roshan, O. P. Rishi. Application of Intelligent Data Analysis in Intelligent Transportation System Using IoT. In Intelligent Data Analysis (**2020**).

[9] M. Sundaresan, D. Boopathy. Use of Machine Learning in Design of Security Protocols. In Design and Analysis of Security Protocol for Communication (**2020**).

[10] S. Khan, K. Kifayat, B. A. Kashif, A. Gurtov, M. Hassan. Intelligent intrusion detection system in smart grid using computational intelligence and machine learning. Trans Emerging Tel Tech (**2020**).

[11] V. B.Barakhnin, L. Kh.Lukpanova, A. A. Soloviev. An algorithm for constructing word forms using inflectional classes for systems of morphological analysis of the Kazakh language. Vestn. NSU. (**2014**), Vol. 12, No 2, pp. 25–32.

[12] K. J. Lyytinen. Implications of Theories of Language for Information Systems. MIS Quarterly 9 (**2020**),No. 1.

[13] N. L. Gurin, Ya. A. Zhuk. Morphological analysis of the text for generating the knowledge base of the dialogue information system. Proceedings of BSTU, (**2016**), No. 6, pp. 156–159.

[14] M. Z. Kurdi. Natural Language Processing and Computational Linguistics 1: Speech, Morphology and Syntax (**2016**).

[15] M. Z. Kurdi. Natural Language Processing and Computational Linguistics 2: Semantics, Discourse and Applications (**2017**).

[16] S. Xian, N. Jing, W. Xue, J. Chai. A New Intuitionistic Fuzzy Linguistic Hybrid Aggregation Operator and Its Application for Linguistic Group Decision Making. Int. J. Intell. Syst. (**2017**).

[17] A. B. Ospanova. Network Security Tools Based on Raspberry Pi. The 4th int. sci. and pract.conf. «Intellectual information and communication technologies as tools for realization of the 3rd industrial revolution devoted for the strategy Kazakhstan-2050», (**2017**), Astana, Kazakhstan, pp. 380-382.

[18] A. B. Ospanova, B. I. Tuleuov. Prospects for using the Raspberry Pi microcomputer in the effective digitalization of Kazakhstan. Bulletin of ENU,(**2018**),No. 4 (125), pp. 95-107.

[19] A. Zharkimbekova, A. Ospanova, K. Sagindykov, M. Kokkoz. Implementation and Commercialization of the Results of the "Multidisciplinary Mobile Computer Classroom Based on Raspberry Pi" Project. iJET, (**2020**) Vol. 15, No. 13, pp. 116-135.

[20] A. Ospanova, B. Tuleuov, A. Zharkimbekova, L. Kussepova, M. Mangmuryn. Mobile Devices and Portative Classroom Based on Raspberry Pi Computers. Proc. 12th National Conf. with International Participation "Electronica 2021", (**2021**), May 27-28, Sofia, Bulgaria, pp. 62-65.

[21] Republic of Kazakhstan'sStatistics CommitteeOfficial website. - URL: tinyurl.com/yckhk7b5.

[22] A. Ospanova, B. Tuleuov, A. Zharkimbekova, L. Kussepova. Report about research work AP09561712 "Intelligent hardware and software system based on Raspberry Pi for network security testing generating recommendations for decision-making". State registration number: 0121RK00559.

[23] ISO/IEC 27005: 2018. Information security risk management. - URL: iso.org/standard/75281.html.