It is assumed in standard information protection technologies that there are owners of this information who put forward requirements for protection. In secret voting systems, the information belongs to the community of citizens, and to protect it, vote organizers must create conditions that allow each voter to make sure that the vote secrecy and accuracy of vote counting are preserved. In developed democracies, this issue is resolved through a widely available audit of all procedures that may be mistrusted. Any voter can conduct such an audit. The anxiety of citizens of democratic countries is based on the idea that if electronic voting is introduced, it will be impossible to conduct such an audit. The article proposes principles of auditing all those software and hardware tools and processes of the online voting system that can generate voter distrust. This audit is carried out using a dedicated server open to voters and their fiduciaries. This server provides continuous monitoring of actions of the service staff in terms of possible interference in the operation of the voting system. Also, due to this server, auditors receive data on the integrity of the voting system hardware and software including its audit tools and an alarm signal in the event of a threat. It was possible to reduce the average time of processing the voter requests to two seconds. This means that processing a maximum of 2,500 voter requests at a vote station will take no more than two hours. Simultaneous access of 50 voters to the server will not make them wait in the queue for more than 2 minutes. Implementation results were described and links were given for conducting experimental voting on the Internet

Keywords: audit of online voting system, data protection, exclusion of illegal influence on voters

UDC 004.78
DOI: 10.15587/1729-4061.2021.238259

DEVELOPMENT OF AUDIT AND DATA PROTECTION PRINCIPLES IN ELECTRONIC VOTING SYSTEMS

Yuriy Khlaponin
Corresponding author
Doctor of Technical Sciences, Professor,
Head of Department*
E-mail: y.khlaponin@gmail.com
Volodymyr Vyshniakov
PhD, Associate Professor*
Viktoriia Ternavska
PhD, Associate Professor
Department of Labor Protection and Environment**
Oleksandr Selyukov
Doctor of Technical Sciences, Professor, Senior Researcher*
Oleg Komarnytskyi
PhD, Deputy Head of Department
Department of Strategic Planning
Department of Transport Infrastructure
of the Kyiv City State Administration
Leontovycha str., 6, Kyiv, Ukraine, 01030
*Department of Cyber Security and Computer Engineering**
**Kyiv National University of Construction and Architecture
Povitroflotsky ave, 31, Kyiv, Ukraine, 03037

Received date 05.07.2021
Accepted date 09.08.2021
Published date 31.08.2021

How to Cite: *Khlaponin, Y., Vyshniakov, V., Ternavska, V., Sieliykov, O., Komarnytskyi, O. (2021). Development of audit and data protection principles in electronic voting systems. Eastern-European Journal of Enterprise Technologies, 4 (2 (112)), 47–57. doi: https://doi.org/10.15587/1729-4061.2021.238259*

1. Introduction

Computerization steadily embraces all processes of human activity including making the most important decisions where it is customary to use the procedure of mass secret expression of will. However, the deep concern of citizens of advanced democracies where electoral fraud is never suspected should be recognized. They do not want to lose the achievements of democracy because of the introduction of new voting technologies. Their experience is exemplary for countries where elections end in scandals and protests. At the same time, the absence of suspicion of fraud is achieved due to openness to the audit of all processes that may cause distrust. Any voter can conduct an audit. Thus, as we can see, the recipe for absolute trust is not complicated but it requires activity on the part of public and political will from organizers of the electoral process. Without these two components, it is difficult to imagine a society in which fair democratic elections are possible.

It is stated in the rights of voters declared in Article 25 of the UN International Covenant of December 16, 1966, that elections must be valid, periodic, free, secret, and equal. Therefore, developers of innovative methods have focused their attention exactly on these requirements, and assurance of trust was forgotten for a long time. Although the requirement of trust was not spelled out in the International Covenant, citizens can reject innovations without it because they will not want to exchange democratic values for a more convenient way of expressing their will. Until the developers provide truly convincing methods of auditing anything that might cause distrust, success in the development of electronic democracy cannot be expected. This is precisely what recommendations of the EU Council for electronic voting standards state in paragraph 39: "The electronic voting system shall be auditable. The audit system shall be open and comprehensive and actively report on potential issues and threats".

The vast majority of electronic voting systems that have allegedly been implemented have not paid enough attention to assurance of trust. However, one cannot rely on the fact that the system was created and maintained by only honest people

and therefore one should not demand complete transparency from innovative methods. All this fails to convince the true defenders of democracy. In fact, there is no other way to ensure voters' confidence than by giving them the opportunity to audit. Therefore, the only sure way to achieve trust consists in the creation of truly auditable systems in which voters have the opportunity to check anything that might cause distrust.

Relevance of the chosen line of study is emphasized by the fact that countries with developed democracies have abandoned in recent years electronic voting technologies in the online mode because of a lack of full-fledged audit mechanisms.

2. Literature review and problem statement

The possibility of using electronic voting in online mode to elect representatives in accordance with the people's will is denied in [1]. It is pointed out that technicians can easily change the result of such a vote. In doing so, there are three problems to be resolved:

1) confidentiality (anonymity of votes);

2) security (no fraud);

3) reliability (accurate vote count).

It is asserted that all these problems are completely solvable with the help of auditing using traditional voting technology. It is thanks to the audit that they (in Italy) never have doubts about the correctness of the vote count. With electronic voting, such an audit seems to be impossible.

Experience of voting using encrypted paper ballots was described in [2]. It was proposed by the author of [3]. At the same time, a widely available audit is possible which ensures trust. Such voting is called electronic voting because of the use of electronics for vote counting; however, these systems do not make it possible to vote online.

Two areas of study can be distinguished in present-day studies concerning online voting systems. The first is the development of the Estonian system which does not use *Blockchain* technology and the second is the systems based on *Blockchain* technology.

Paper [4] analyzes the main client attacks on the Estonian voting system. A trespasser was found to be able to re-vote using fake software. A more secure voting protocol was proposed. It was suggested in [5] to introduce additional checks in the Estonian system by users who are aware of cyber risks which, according to the author, should strengthen the confidence of voters. However, these studies do not touch upon the issues of a widely available audit of the hardware and software of the voting system.

Generalized analysis of the first line of studies presented in [6] showed that Estonia remains the only country where people have been voting online for many years. Their experience was not adopted by other countries because of the lack of a comprehensive audit required according to par. 39 of the EU Council recommendations on electronic voting standards [7].

The issue of ensuring confidence in voting systems is discussed in [8] where it is recommended that election management bodies maintain transparency of the vote-counting process so that everybody can be convinced of election legitimacy. This underlines the need for a full-fledged, widely available audit.

To ensure confidence in voting systems, the use of *Blockchain* technology has been proposed and patented [9]. Studies continue in this line which is reflected in [10–12] where proposals are introduced to improve security and anonymity for voters, as well as to counteract dishonesty on the part of candidates. The experience of using this technology at elections in Moscow described in [13] showed that delays in servicing voters reached an hour. Analysis of programs using *Blockchain* described in [14] showed the presence of significant weaknesses in terms of data protection. In other words, it is difficult to imagine an accessible audit in voting systems that use *Blockchain* technology since only a limited circle of specialists understand this technology.

Thus, analysis of studies in the field of online voting showed that there is a gap in terms of ensuring a widely available audit of software and hardware for recognition and counting of votes. Note that such an audit is required in accordance with par. 39 of the EU Council recommendations on electronic voting standards. This makes it possible to assert that studies on the development of principles for a widely available audit of software and hardware of online voting systems are expedient.

3. The aim and objectives of the study

The study objective implied the development of principles for a widely available audit of software and hardware for recognizing and counting votes in the online electronic voting systems. Such an audit should be available to both voters and their fiduciaries which will eliminate reasons for a grounded distrust of the voting system.

To achieve this objective, the following tasks were set:

– define the processes and means of an electronic voting system in which voters' distrust can be manifested;

– propose a set of measures to audit the processes and means to which voters may be distrustful;

– define methods of protection of confidential data when sending via Internet channels;

– analyze the operating speed of the proposed data protection methods in conditions of a simultaneous request of many voters to the voting server;

– propose a mechanism for eliminating illegal influence on the choice of voters through bribery or other methods of moral or force duress.

4. Methods and means of the study

The main study method involved full-scale modeling of electronic voting systems with audit tools followed by testing in conditions close to real ones. With the help of this method, certain technical solutions, software, and hardware were selected or rejected. Computer equipment and access to the Internet for studies were provided by the Scientific Research Institute of Automated Systems in Construction under the Ministry for Development of Communities and Territories of Ukraine. For seven years (2014 to 2021), hundreds of students of higher educational institutions of Ukraine took part in the selection of technical solutions and testing the voting system models. A leading role in these studies was undertaken by Kyiv National University of Construction and Architecture. Besides, students and lecturers at the National Aviation University and National Technical University of Ukraine (Igor Sikorsky Kyiv Polytechnic Institute) took an active part in the study. Since September

2020, the discipline Protocols and Algorithms of Electronic Voting which provides for a detailed study and practical use of the online voting system has been introduced there.

Since the issue of trust deeply affects the human factor, it cannot be considered a purely technical issue. At the same time, one cannot rely on the opinion of one or two experts. Their number should be in the tens, or better in the hundreds. In addition, experts should study the voting system deeply enough so that their opinion is based not on emotions but on knowledge in the IT field. The educational institutions where students study computer science can be considered the most suitable places for these studies. At the same time, students can simultaneously study, operate and refine the system, as well as act as experts.

5. The results obtained in the study of audit tools and data protection methods in electronic voting systems

5. 1. Defining the processes and means of the electronic voting system as the objects of voter distrust

It is known that people/s distrust of electronic voting concerns the preservation of the secrecy of their votes and honest counting [15].

Disclosure of the vote secrets makes it possible to force voters to vote not of their own free will but by the will of another person using bribery or other methods. Therefore, if there is a danger of illegal influence on voters, protection against such influence should be provided. To this end, it is necessary to exclude the possibility of finding out a single voter's voting result. In other words, it is necessary that no one, except the voter himself, knew the result of his vote.

It is necessary that the audit system make it possible to check technical means of vote accounting. This is explained by the fact that the voter will not believe in the flawlessness of the technical means as long as they represent a "black box" for him. If the auditor will not see anything except pictures on the screen during the audit, then he may suspect that these pictures were drawn by an imitator. At the same time, in order to achieve trust, it is necessary to show that the vote-counting server is exclusively an electronic analog of a transparent box for collecting ballots. Auditors should see that this server is limited in its technical characteristics and cannot perform any additional tasks such as simulating fair work.

The main process that gives rise to voter distrust consists in the actions of the staff managing the voting server. Therefore, the audit system should ensure continuous monitoring of such activities and provide evidence of the absence or presence of any threat.

The server program that receives and counts votes is the tool that can cause mistrust in the first place. This program should be as simple as possible and open to verification. In operation of this program, auditors must have evidence of its integrity and absence of any outside interference in its work.

5. 2. Measures applied to audit the processes and means as the objects of voter distrust

The main principle of constructing the system under consideration consists in relieving the voting server of those functions that do not affect the secrecy of votes and their counting. This makes it possible to use a well-known mini-computer model as a server which will facilitate the audit of technical means. Although knowledge in the field of computer science is required to conduct such an audit, the number of voters capable of such an audit is growing every year. It should be noted that informatics useful for this case is studied in a sufficient scope in present-day schools.

The first step in conducting an audit consists in observing the process of installing the voting server. Since this is done when there is no critical information on the server yet, any voter or his fiduciary can have a possibility of performing this stage. The auditor's task consists only in checking the accuracy of the actions performed according to the published instructions.

Subsequent auditing requires continuous monitoring of the processes running on the voting server. To this end, auditors use a special audit server that can be installed on their own or with the help of their trusted representatives. Connections between the main blocks of the voting system are shown in Fig. 1.

It should be noted that suspicion of the voter register forgery may be the reason for people's distrust. This can be easily identified by publishing data on the number of voters for every street within a polling station, for every house within a street, and for every apartment within a house. Then the voters themselves will find extra tenants in their apartments, extra apartments in their houses, and extra houses on their streets. Programmer efforts are not needed here but only the political will of the election organizers is enough.

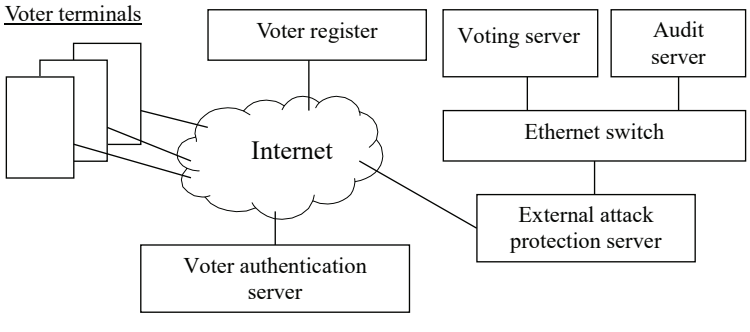

Fig. 1. Block diagram of the electronic voting system

Creation or update of the voter register is the preparatory stage of voting. For those who vote *online*, identifiers and passwords are entered into the register. Passport numbers can be taken as identifiers. Passwords are stored encrypted. The function of the degree of the primitive element of the final field was used for encryption where the concatenation of an identifier with a password is the index. The final field was chosen such that the discrete logarithm problem has not been solved. This ensures absolute protection against disclosure. When registering, the voter is provided with a temporary password and a link to the site to check or change his password. Before each vote, a file with encrypted identifiers and passwords (74 bytes of cipher for each voter) is created and sent to the voting server for voters who vote *online*.

Voting servers were implemented on well-known mini-computers of Raspberry Pi 3, Model B, type (price does not exceed USD 50). They are open to external inspection. Each such server serves one voting station. According to the laws of Ukraine, the number of voters at the voting station cannot exceed 2,500. A general view of the server with mounting elements is shown in Fig. 2.

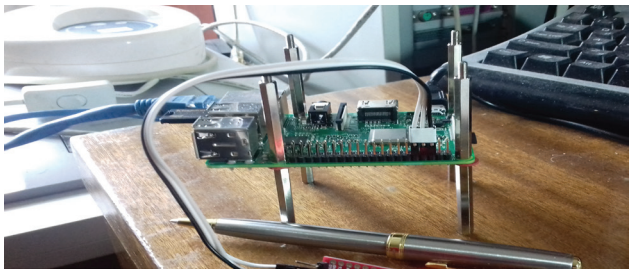

Fig. 2. Server based on minicomputer *Raspberry Pi 3, Model B*

The presence of auditors during the start of the server eliminates the suspicion that hardware or the operating system (OS) has been forged. OpenBSD v. 6.6 was used as OS in a minimal configuration. The importance of demonstrating technical means is determined by the fact that auditors can make sure that equipment is not powerful enough to build a simulation model of the voting process. OpenBSD is an open OS and its authenticity can be verified by downloading it to another computer and then comparing the files with each other, as described in [15]. This OS monopolizes the server until the power outage occurs and makes impossible any external interference in the operation of application programs. In addition, the audit server automatically detects all processes that may be dangerous. The principle of operation of the audit server is as follows. Periodically, every few seconds, this server contacts the voting server (via the SSH protocol) for information about the currently active processes (ps-aux command). The audit server does not react to the processes launched by auditors and the operating system. All other processes are logged and cause an alarm signal. This is how all potentially dangerous processes are identified.

The server for protection against external attacks belongs to the Internet service provider. This server should be transparent to voters, however, if the provider agrees to cooperate with attackers, this server can host the means to create an intermediary attack called Man In The Middle (MITM) [16]. This attack is a very dangerous threat that can disclose and spoof votes. The difficulty of dealing with such a threat lies in the fact that it can be implemented at any time on any intermediate server between the voter and the voting server. Thanks to the audit server, detection of the middleman attacks is straightforward but requires the participation of voters in their identification. To protect against this threat, each voter is provided with easy-to-use tools to detect the presence of a threat at the moment when a connection to the voting server is established. These tools are discussed in subsection 5.3.

In addition to clarifying identity based on biological or other characteristics, the server for additional authentication of voters makes it possible to provide protection against illegal influence on voters by bribery or other methods to force them to vote contrary to their own opinion. Since such authentication complicates the procedure of expressing will, it is advisable to cancel the additional authentication stage in the case of voting at meetings or sessions in online mode. At the same time, the means of protection against violation of the secrecy of votes and falsification of their counting can be preserved in full. The list of requirements to the voting server which must be met to ensure the voter trust is given in Table 1.

Let us consider the meaning of each of the listed requirements and the features of their execution. The first requirement is ignored by most developers believing that inspection of hardware by citizens or their fiduciaries is unnecessary. However, in this case, the voting server will be a "black box" even for the voters who are well versed in computer technology. At the same time, it is impossible to eliminate the suspicion that this "black box" contains an imitator program demonstrating to voters their allegedly fair votes but in fact revealing and replacing their votes. Therefore, it is impossible to gain voter trust without opening up the hardware for auditing. The choice of a well-known mini-computer with open mounting elements makes it possible to completely eliminate the suspicion of a "black box" with a hidden simulator. It is impossible to create a simulator on this computer board due to a lack of resources. In addition, all of those resources will be hijacked by the OpenBSD operating system installed and launched under the control of public representatives. Such control is allowable at a time when there is no critical information on the server though this requires the direct presence of citizens in the room where the server equipment is installed. It is enough just to check the accuracy of instructions for installing and launching the operating system. Highly qualified specialists are not needed for such a check because the instructions are simple and open. This ends up the presence of controllers in the server room. Subsequent control occurs remotely based on the ps-aux command which displays parameters of all active processes running on the server at the current time. The result of this command execution is shown in Fig. 3.

Table 1

Requirements to the voting server to ensure voter trust

No.	Requirement	Execution of requirement
1	Accessibility of hardware auditing without personal restrictions	Selection of well-known hardware means with assembly elements open for inspection and a possibility of replacing them in a case of suspicion of counterfeiting
2	Availability of OS audit during its installation and during the entire operation period	Choice of an open OS that makes it possible to create the user's controllers that are forbidden to execute commands that can harm the server but permitted to read all files and execute safe commands
3	Impossibility of secret replacement of hardware and OS during the entire period of operation after implementation of par. 1 and 2	Provision of remote access to the audit server to all voters and their fiduciaries and installing additional audit servers to continuously check the state of the voting server hardware
4	Controlling the staff actions including download of software packages to manage the server	Adoption of the rules according to which the staff must execute the history>haabbccdd.txt command before terminating the management session where instead of aabbccdd the following is entered: aa for month, bb for date, cc for hours, dd for minutes. This makes it possible to control all actions of the staff
5	Availability of checking the application software	Choice of well-known computer languages (HTML and JavaScript) and a simple programming style, as well as openness of the software and availability of its tests
6	Prompt notification in case of violations detected	Establish clear rules of accepting reports of violations and react accordingly

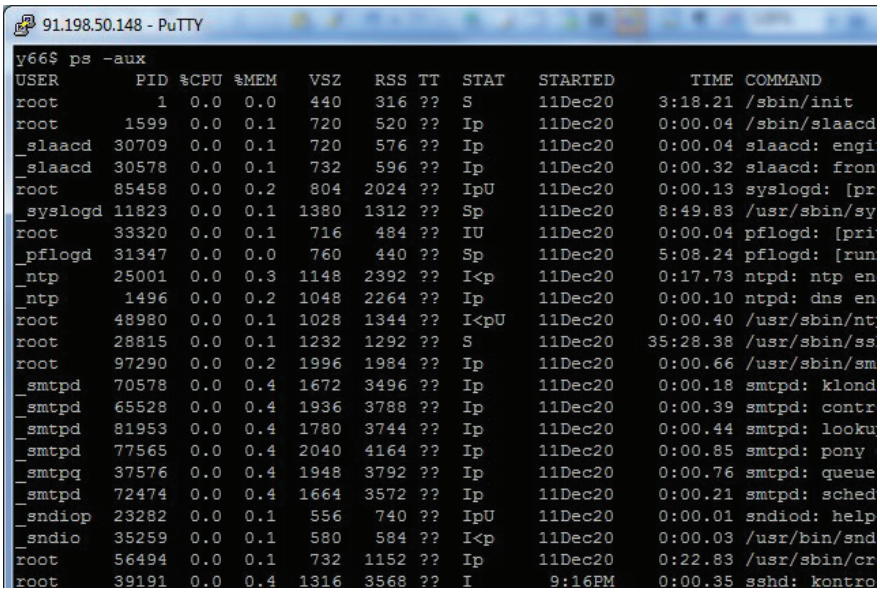

Fig. 3. The result of *ps—aux* command execution

The *ps-aux* command makes it possible to detect any attempts of intervention in the work of the server because each intervention is accompanied by the appearance of a new active process. To identify these interventions, it is sufficient to analyze only the first two and one last attributes of the entire set of attributes of active processes. The first column (the USER attribute) shows the identifier of the user who launched this process. The second column (the PID attribute) contains a random number assigned to the process at the time of launch and remains unchanged until its termination time. The last column (the COMMAND attribute) contains the command that initiated this process. At the moment of starting the OS, more than 20 active processes of the OS itself arise. The first of them is assigned PID=1 and the others are assigned unpredictable random PID values. Therefore, it is impossible to restart the OS secretly since this will change all PID values, except for the first one. This is easy to spot by remembering the initial result of the ps-aux program execution. It is clear that replacement of hardware is completely impossible without restarting the OS. The administrator instructions suggest that it is necessary to create two users named admin and control after starting the OS. The admin user is granted administrator rights to work with files, but only within the */home/admin/* directory and the control user is granted the auditor rights. In addition to these users, a *root* user with full rights must be created in the system. It is impossible to configure the system without this step but access to the server with the root user rights is denied after performing actions that require full rights. After that, only the admin user with limited rights can manage the server. Names of all OS users are stored in the /etc/group file. This makes it possible to check the correctness of the execution of the adduser command to be used by the administrator to create users. Fig. 3 shows 23 processes implementing the functions of the OS launched on December 11, 2020 (as evidenced by the value of the STARTED attribute). Also, 4 processes associated with the ps-aux command entered by the control user and one process started by the admin user are visible.

Let us explain the four active processes associated with the ps-aux command. The first process with PID=39191 is a request for the sshd service. The purpose of this service implied the authentication of clients and encryption of messages. Since the control client is a registered user of the OS, it is possible to create another process of the same service (PID=13885). The third process (PID=52043) enables the control user to enter his commands. This process controls whether the user has permission to execute certain commands. The process with PID=7063 implements the execution of the safe ps-aux command by the control user. The admin user process (PID=3771) is a program that decrypts and counts votes. The auditing task consists in checking the fact that the administrator has launched a regular program stored in the voting server in the /home/admin/ directory. This program has been published in advance on a voter-accessible website. Both files with the program are open for copying on the Internet. By comparing these files, it is easy to make sure that there is no substitution or modification of the program. The process of starting the program is controlled through a file with command history which, according to the instructions, must be created by the administrator after each server management session. Thus, voters or their fiduciaries can use simple procedures to verify the validity of the current application. Automatic detection of dangerous active processes on the voting server is as follows. After the next execution of the ps-aux command, each line of the result is checked against the following three lists of security features. The first list contains PID values of the constantly running OS processes. PID of the application is also added to this list after it has been verified for correctness. The second list shows the USER attribute values, namely control and some user IDs assigned by the OS itself to support unattended operation. The third list shows values of the COMMAND attribute which includes the sshd, -ksh commands, and a few other commands that the OS automatically executes to keep its own running. A process is considered safe when it detects at least one tag of these three lists. Thus, any interference with the OS including regular actions of the administrator is automatically detected, logged, and initiates sending of a message about a possible danger. The openness and simplicity of the program application make it possible to analyze all the transformations associated with encryption and counting which enables making sure that there are no harmful tabs.

Implementation of the latter requirement for responding to violations in the event of their occurrence depends only on the political will of the vote organizers. No matter how flawless the technical means of detecting threats, the vote organizers can ignore information obtained with their help.

5. 3. Data protection during transmission over the Internet channels

In addition to automatic detection of dangerous processes on the voting server, each voter can check the server operation and identify possible information threats using the audit key. This key can open a web page for auditing the voting station server. The general view of this web page is shown in Fig. 4.

Using the OS Command button on the audit web page, voters can initiate an additional check on their polling station server at any time by executing the sysctl hw command, as shown in Fig. 5.

The sysctl hw command displays basic specifications of the voting server including the type of computer (Raspberry Pi 3, Model B, Rev 1.2) and amount of RAM (959225856). Virtually all important characteristics of the server's firmware can be verified remotely by entering many permitted safe commands.

It is important that thanks to the audit server, each voter can ensure that there is no middleman attack when exchanging confidential data with the voting server. To do this, he must use the Connection Log button (Fig. 4) to view the connection codes, among which he must find the code of his connection to the server. The general view of the message with the personal connection code and the connection log page are shown in Fig. 6, 7, respectively.

Although middleman attacks are unlikely due to the complexity of implementation and the possibility of bringing the ISP to justice, given that these attacks are a very dangerous threat, they cannot be ignored.

To protect data during the exchange of information between the voter and the voting server, the Vernam cipher is used which provides absolute protection against opening and is mathematically proven in [17]. Although the use of this cipher requires the execution of special conditions, it has the advantage that data disclosure during transmission becomes absolutely impossible. This is an important component for ensuring the voter's trust. Table 2 provides a list of conditions for absolute data protection during transmission.

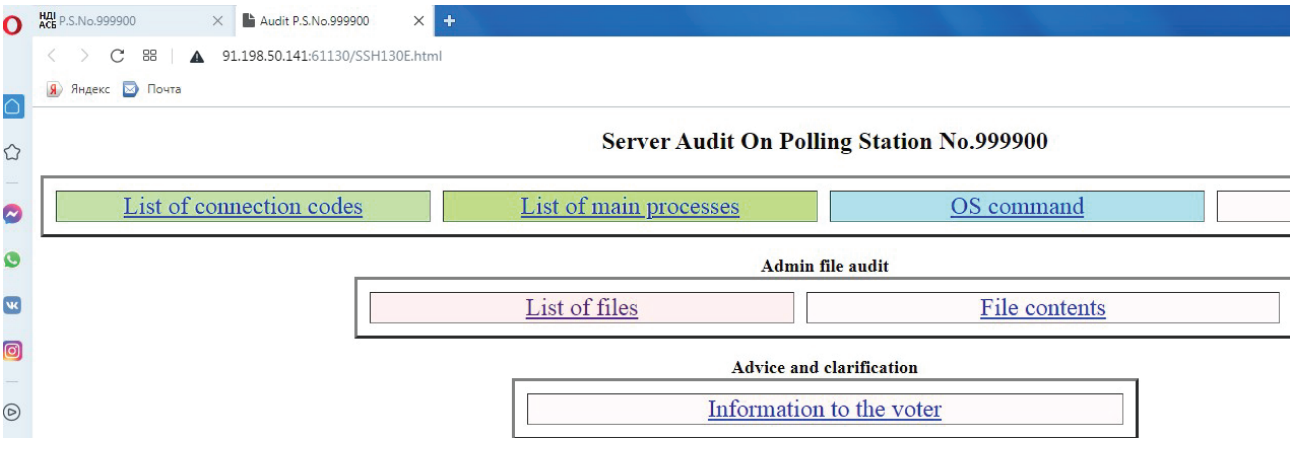

Fig. 4. The web page of audit of the polling station server

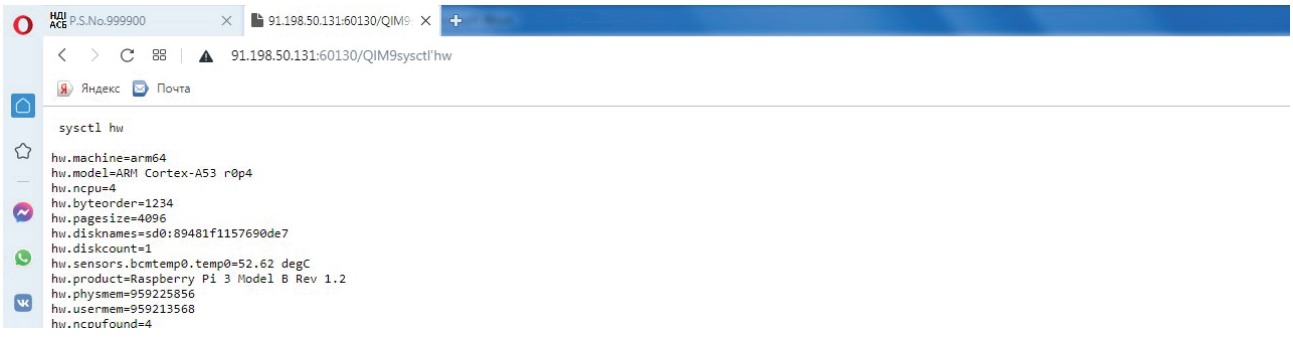

Fig. 5. The result of executing the sysctl hw command

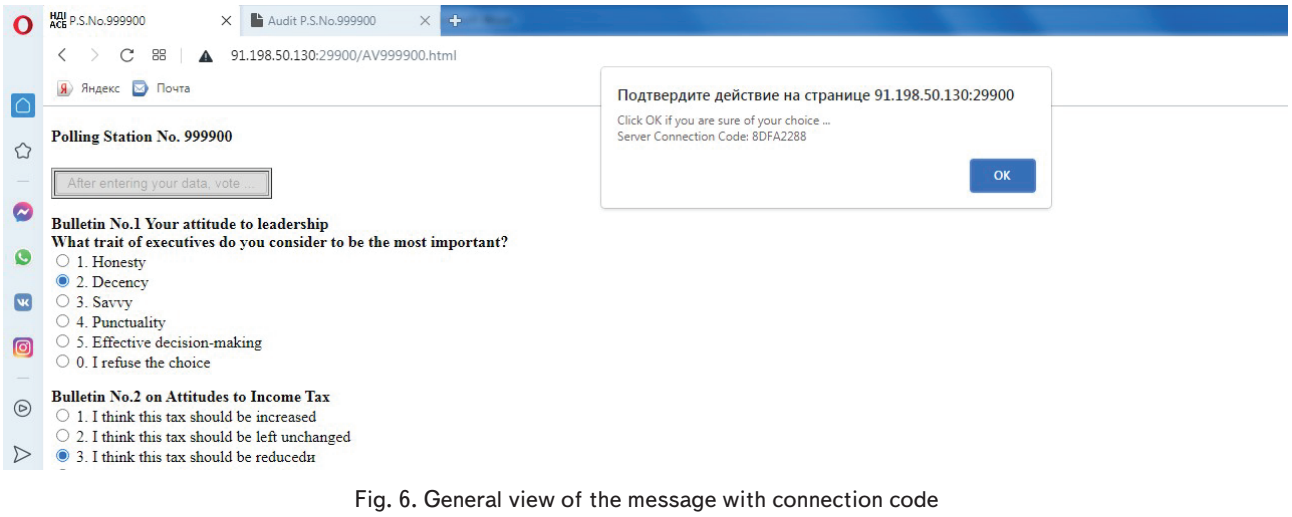

Fig. 6. General view of the message with connection code

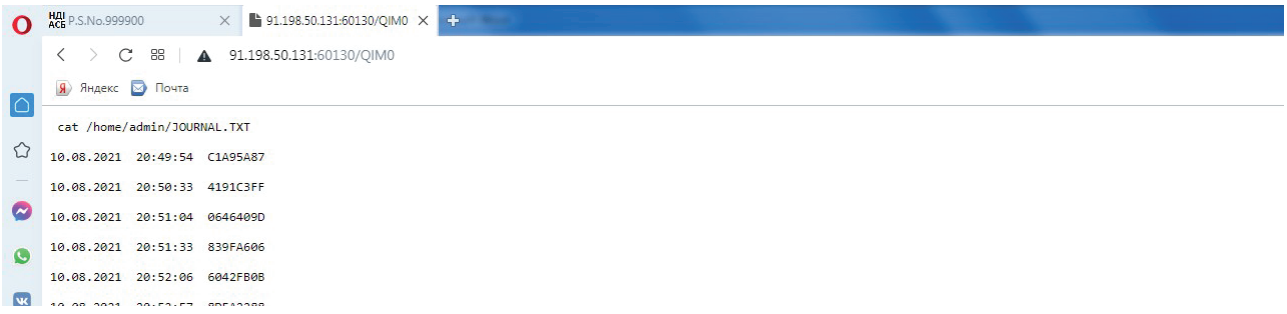

Fig. 7. General view of the connection log page

Table 2

Conditions for ensuring absolute data protection during transmission

Condition	Condition fulfillment
Generating random bit sequences (not pseudo-random sequences)	A method for generating random (not pseudo-random) bits has been implemented which makes it possible to generate random sequences on any computer, as described in [18]
Each random bit sequence can be used for encryption only once	For each communication session, random bit sequences are generated independently of each other
An absolutely secure communication channel should be used to exchange with random bit sequences	Exchange with random bit sequences occurs according to the Diffie-Hellman algorithm [19] with such parameters for which there is no possibility of data disclosure in current conditions

Study [20] has substantiated the choice of parameters of the Diffie-Hellman algorithm for the problem of electronic voting. Parameters of the algebraic group for implementation of the algorithm are selected based on two conditions. Firstly, to ensure the impossibility of disclosing data and, secondly, to ensure that time of cryptographic transformations does not exceed the value that would make it possible to vote within the time established by the election rules.

5. 4. Analysis of the performance of applied data protection methods

Since cryptographic transformations require much more time than conventional arithmetic operations, it is necessary to find out the effect of these consumptions on the average time of the voter service.

In Ukraine, 12 hours are provided for the voting procedure when the number of voters at polling stations does not exceed 2,500 persons. Since the voting server is the bottleneck in the data processing chain, this means that it should not spend more than 17 seconds on processing the voter data. Although the number of voters voting remotely cannot reach 100 % because of the uneven flow of requests and the possibility of pauses, it is desirable to reduce the time for processing the voter data to 3–5 seconds. For voting during meetings where the number of voters does not exceed 500 persons, the voting process should not take more than an hour. This means that the average time for processing data of one voter by the server should not exceed 7 seconds. In order to prevent data disclosure, an algebraic group was chosen in the form of a Galois field with characteristic 2 and a degree which is a safe prime number from the series 503, 563, 587, 719. Since the solution of the discrete logarithm problem for such fields is not yet known, such protection for present-day voting systems is absolute. Because the amount of data transmitted during a voter-server communication session does not exceed 500 bits, it is sufficient to use the $GF(2^{503})$ field. At that, the time for cryptographic transformations on the server is from 6 to 7 seconds on one processor core. To ensure the failure-free operation of the system, the processing of voter requests must be synchronized with their arrival. In the event of a synchronicity failure, as shown in [21], a loss of stable operation of the network occurs which can lead to a significant investment of time to restore stable operation. Since the Raspberry Pi 3 processor has 4 cores, it was possible to fulfill the synchronization requirements by parallel processing of requests on four cores according to the approach proposed in [22]. Thanks to this measure, the average processing time for voter requests was reduced to two seconds. This result fully satisfies the performance requirements for the voting systems. At a simultaneous call to the server from 50 voters, the waiting time for completion of the service procedure will not exceed two minutes.

5. 5. Elimination of illegal influence on voters

According to article 25 of the UN International Covenant, voters should have the right to freedom of expression. However, there are cases of bribery and other cases of illegal influence when voters are forced to vote not of their own free will but at someone's orders. At the same time, attackers need to find out the voter's voting result, otherwise, they will not be able to achieve their goal. The only place where a voter can truly hide his choice is his own memory. In this case, the system should not show the result of anyone's voting, otherwise, it may become available to an attacker. This problem was solved by introducing another procedure proposed in [20] which coincides in time with the period of updating the voter lists. Clarification usually takes place within two weeks before the start of voting. Such clarification is necessary in any case since voters have the right to refuse remote voting in specific elections and they must be included in the list in order to receive a paper ballot. Thus, for remote voting, voters need to contact the server twice which will work according to the schedule shown in Fig. 8.

First voters' addressing to the voting server takes place during the period of updating the lists which, according to the legislation of Ukraine, has a duration of 2 weeks. The form of the voter's dialogue with the server during this period is shown in Fig. 9.

The following three tasks are solved with the help of this dialogue:

– entering a code that will replace the password during voting;

– the voter's confirmation by the fact of his personal vote (using the server of additional authentication (Fig. 1));

– receiving the voter's final decision on remote participation in the given voting.

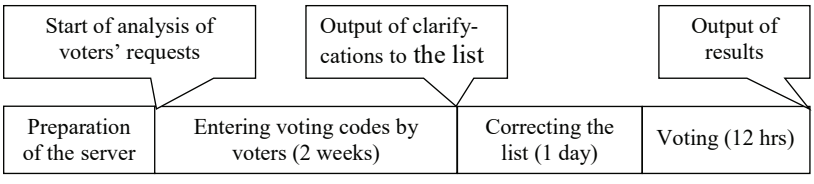

Fig. 8. Operating schedule of the voting server

No one, except the voter himself, should know the voting code. Therefore, it is necessary for the voter to invent this code on his own and not forget it until the moment of voting. Knowledge of this code protects the voter's vote from outside influence since it has the ability to mislead attackers by multiple fictitious voting with an incorrect code.

Additional authentication is required to prevent the transfer of voting rights to another person because this is prohibited by the electoral law. Any identity confirmation signs, e.g. biological ones, can be used on the authentication server. This server is not subject to voter audits as it does not contain information related to the issues of distrust. Therefore, proprietary software can be used here. This server's task consists in responding to requests from the voting server. The requests contain only the voter's ID while responses contain permission or prohibition. In the simplest case, a person can use traditional authentication technologies in the form of face-to-face verification showing his passport at a special point where conditions should be created similar to the voting booths. At the same time, voters can be authenticated at any time convenient for them within two weeks, and they will not need to come to the polling station on the election day. It should be noted that a time limit (15 minutes) has been adopted for entering the voting code after passing the authentication. In this case, the number of attempts to enter the code is limitless and the last code received by the server is considered valid. After completion of the period of entering the voting codes by voters, the server will provide access to the data of the voters who entered the codes. This data is used at voting stations to prohibit the issuance of paper ballots. Voting codes are stored exclusively in the server's RAM to which only the application program has access. The form of the voter's dialogue with the server during voting is shown in Fig. 10.

The server counts the voter's vote only if the code is correct but the message about the vote receipt is sent if only the first two characters of the code are correct and the message has the form shown in Fig. 11.

Thus, the voter is given the opportunity to mislead attackers since it is possible to vote fictitiously with a wrong code as many times as you like. At the same time, it is impossible to distinguish a real vote from a fictitious one by their external signs.

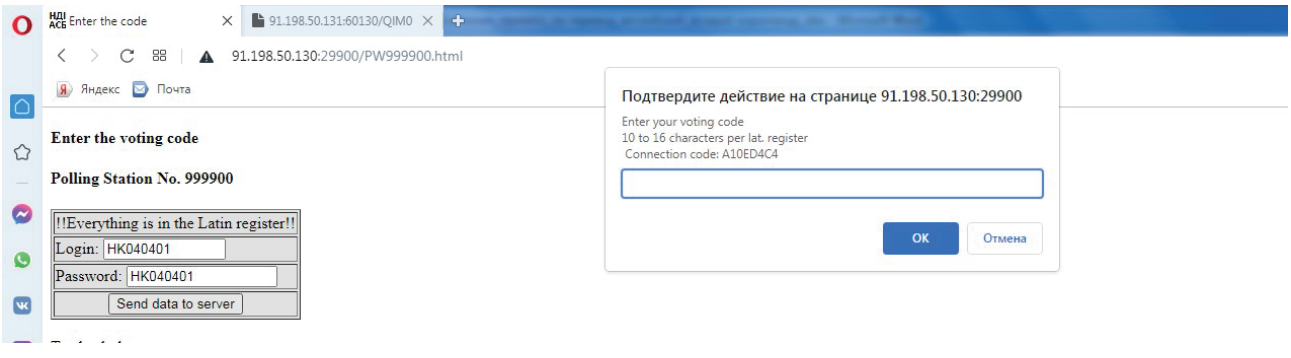

Fig. 9. The form of a dialogue between the voter and the voting server when entering the code

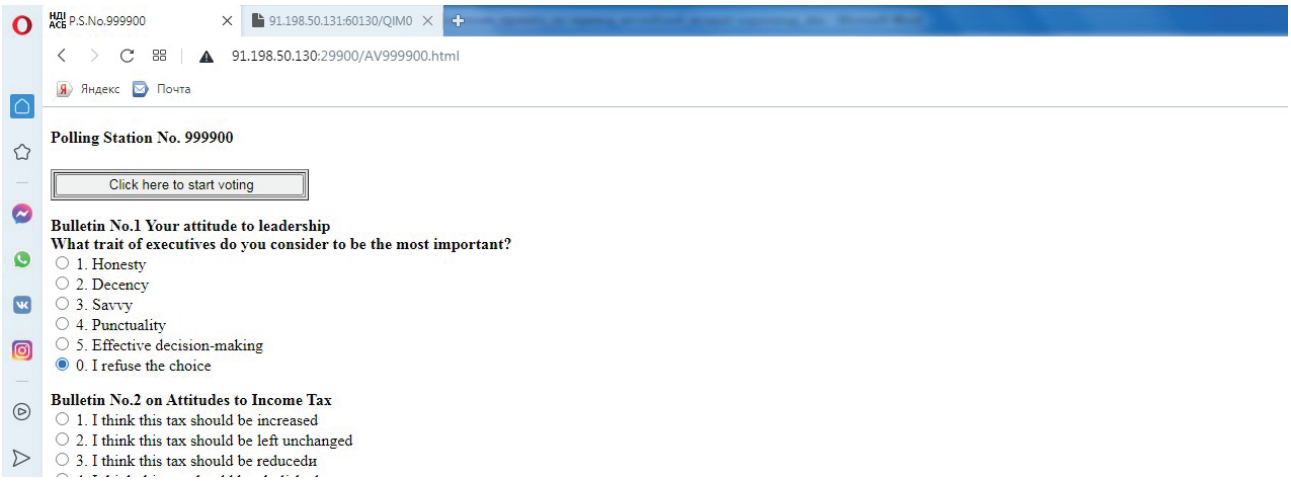

Fig. 10. The form of a dialogue between the voter and the server during voting

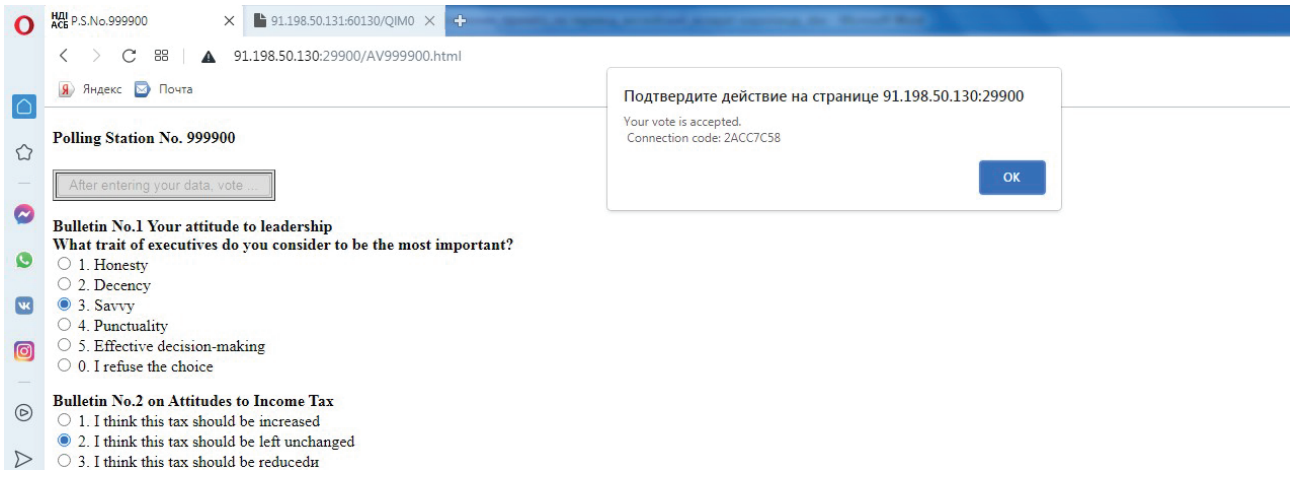

Fig. 11. Message to the voter on his vote receipt by the server

6. Discussion of results of the development of principles and study of audit tools and methods of data protection in electronic voting systems

Students who created the voting system that could be trusted by voters were dependent only on their own initiative. At the same time, everyone could make proposals, implement them and act as an expert. The best technical solutions were selected through repeated checks with the participation of hundreds of voters. Thanks to seven years of work, it was possible to obtain an *online* voting system, operation honesty of which leaves no room for doubt. Anything that may be in doubt is audited in this system. For a complete check of the application software the capacity of which was minimized, no high qualifications are required but an average level of knowledge of HTML and JavaScript is sufficient.

Note that according to the plan of the government of Ukraine dated June 12, 2019, No. 450-r [23], the implementation of electronic voting is primarily entrusted to higher educational institutions. This approach differs significantly from the one that existed in other countries which relied entirely on professional developers. As a result, they received a stream of criticism because of the impossibility of ensuring the trust of citizens as described in studies [1, 11, 12]. Thanks to this approach, the idea expressed in [24] was implemented where it was recommended to follow the path of simplifying the new voting systems instead of complicating them. In fact, this study was based on the idea of creating simple and auditable electronic voting systems that was not supported by professional developers. The idea of simplification is not attractive for professionals as it can negatively affect their funding. As analysis of recent electronic voting systems shows, they continue to be complicated [14]. But the idea of simplification was supported in the student study [25] where openness and simplicity are promoted to the fore in order to ensure voter trust. Voters will be able to get rid of distrust only if the audit is widely available from the beginning of voting to the completion of the vote count. Auditing the electronic voting systems requires knowledge acquired in educational institutions. In addition, if students are engaged in development and operation in the learning process, then the software and hardware solutions will be simple and understandable. Students have been using this system to elect representatives to their student government bodies for more than three years at Kyiv National University of Construction and Architecture. Access to the system is open through the site http://vybir.knuba.edu.ua/.

The general view of servers on the provider's platform is shown in Fig. 12.

This system is capable to conduct experimental voting in Ukrainian, English, and Russian. Since October 2020, the Academic Council of the University has been using this system to conduct secret polls [26]. Using this system, the Ukrainian Red Cross Society has successfully held elections to the governing bodies on December 4, 2020. Voters voted without leaving their cities all over Ukraine.

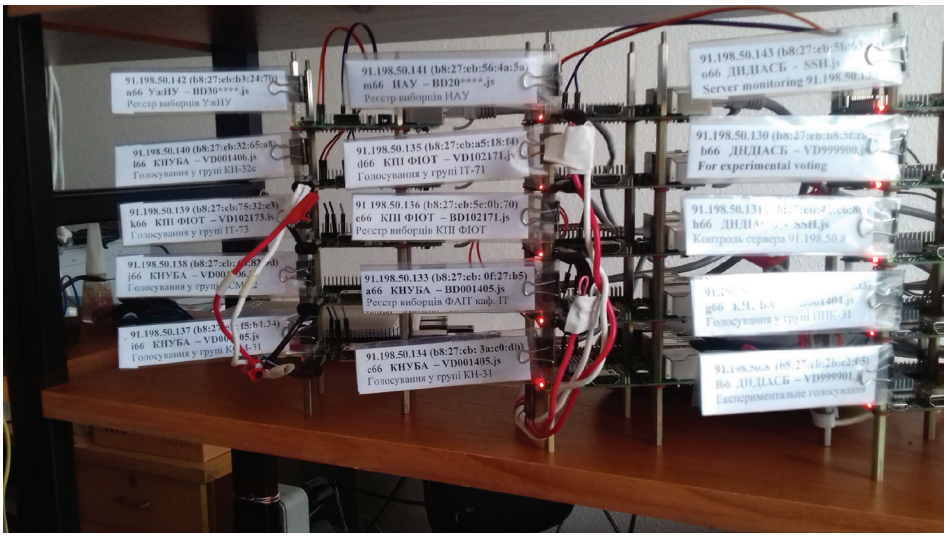

Fig. 12. General view of voting servers on the provider's platform

The need for knowledge in the IT field should be attributed to the limitations of the audit concept proposed in this study. However, this knowledge is rapidly spreading in recent years. Even if 1–2 % of voters can become full-fledged auditors, this may be enough to earn voter confidence in the voting system.

As a disadvantage of the system, it should be pointed out that the voting results in a form of readings of the vote counters are stored only in RAM. This is good from the point of view of the impossibility of disclosing information about who and how voted, however, it does not enable restoration of voting results, e.g. in a case of failure of the server power supply. At the same time, the only way out of this situation consists in conducting a repeat voting. It is possible to combat this drawback by saving the RAM image on another computer but this will complicate the system which in its turn will complicate the audit procedure.

To further improve the proposed technology, it would be advisable to create a specialized OS with two types of users for management and auditing. In doing so, all control commands are recorded and audited at the OS level. In addition, it is possible to remove a lot of unnecessary things keeping only what is necessary for the application program to work. Then it would be possible to simplify the audit procedure and get rid of all fears associated with accidents and emergencies.

7. Conclusions

1. Principles of a widely available audit of all those software and hardware means and processes of the online voting system which can cause voter distrust have been developed. This audit is carried out using a dedicated server open to voters and their fiduciaries. This server provides continuous monitoring of actions of the service personnel in terms of possible emergency interference in the operation of the voting system. Also, due to this server, auditors receive data on the integrity of the voting system hardware and software including its audit tools and an alarm signal in the event of a threat.

2. In order to ensure a full audit, the following set of measures is proposed. First, voting servers are installed and launched under the supervision of voters or their fiduciaries during a period when there is still no critical information on the servers. Secondly, after launching the servers, voters continue the audit remotely using their specialized servers without losing information about interference with the system. Thirdly, all software and hardware solutions are simple and open which minimizes time for their full audit. As a result, causes of the manifestation of voter distrust were eliminated.

3. Absolute protection is provided during the transfer of confidential data via the Internet channels through the use of well-known methods of perfect encryption. Attempts to intermediary attacks are detected by voters when establishing communication with the server. This makes it possible to timely prevent attacks.

4. Average time for processing requests by the voting server is about two seconds. This means that in a case of simultaneous access of 50 voters to the server, waiting time in the queue will not exceed two minutes which fully meets the voting system requirements.

5. To eliminate illegal influence on voters by bribery or other methods of moral or forceful pressure, voting technology was used which allows the voters to conduct fictitious voting to deceive intruders. This technology does not allow anyone, except the voter himself, to know the actual result of his vote.

Acknowledgments

The authors express their deep gratitude to Bruce Schneier for the ideas that formed the basis of this work. Firstly, the fact that openness of the system contributes to the improvement of data protection, and secondly, that it is necessary to follow the path of simplifying the electronic voting systems and not complicating them.

The authors express their gratitude to Professor Denis Chernyshev, the first vice-rector of Kyiv National University of Construction and Architecture for a significant contribution to the development of the system at the implementation stage.

The authors express their gratitude to scientists of the National Aviation University of Ukraine for ideological support and assistance in the development of the system:

– Professor Alexander Korchenko, Head of Department of Information Technology Security;

– Vladimir Chuprin, Professor of Department of Telecommunication and Radioelectronic Systems.

The authors also express their gratitude to scientists of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" for participation in the creation and development of the system:

– Anton Silvestrov, Professor of Department of Theoretical Electrical Engineering;

– Vadim Poltorak, Associate Professor of Department of Automation and Control in Technical Systems.

References

1. Electronic Vote & Democracy. Available at: http://www.electronic-vote.org/
2. Prozori vybory u KNU. Kyivskyi natsionalniy universytet imeni Tarasa Shevchenka. Available at: http://univ.kiev.ua/news/8696
3. Devid Bismark: Elektronne holosuvannia bez obmanu. Available at: https://www.ted.com/talks/david_bismark_e_voting_without_fraud/transcript?language=uk
4. Ajish, S., Anil Kumar, K. S. (2020). Secure I-Voting System with Modified Voting and Verification Protocol. Advances in Electrical and Computer Technologies, 189–200. doi: https://doi.org/10.1007/978-981-15-5558-9_19
5. Solvak, M. (2020). Does Vote Verification Work: Usage and Impact of Confidence Building Technology in Internet Voting. Lecture Notes in Computer Science, 213–228. doi: https://doi.org/10.1007/978-3-030-60347-2_14
6. Bezpyatchuk, Zh. (2019). U Zelenskogo obeschayut onlayn-golosovanie: chem eto grozit? BBC News Ukraina. Available at: https://www.bbc.com/ukrainian/features-russian-49266210
7. Recommendation CM/Rec(2017)5[1] of the Committee of Ministers to member States on standards for e-voting. Available at: https://search.coe.int/cm/Pages/result_details.aspx?ObjectID=0900001680726f6f

8. Internet-holosuvannia: pytannia do rozghliadu. Zahalnyi ohliad dlia orhaniv administruvannia vyboriv. Bila knyha IFES. Available at: https://ifesukraine.org/wp-content/uploads/2020/04/IFES-White-Paper-Applegate-Chanussot-Basysty-%E2%80%98Considerations-on-Internet-Voting%E2%80%99-Mar-2020-Ukr-1.pdf
9. Ernest, A., Hourt, N., Larimer, D. (2016). Pat. No. US 2017/0109955 A1. Blockchain Electronic Voting System And Method. No. 15/298,177; declareted: 19.10.2016; published: 20.04.2017. Available at: https://patentimages.storage.googleapis.com/ba/6f/d2/1920626de10c1b/US20170109955A1.pdf
10. Ibrahim, M., Ravindran, K., Lee, H., Farooqui, O., Mahmoud, Q. H. (2021). ElectionBlock: An Electronic Voting System using Blockchain and Fingerprint Authentication. 2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C). doi: https://doi.org/10.1109/icsa-c52384.2021.00033
11. Alvi, S. T., Uddin, M. N., Islam, L., Ahamed, S. (2020). From Conventional Voting to Blockchain Voting: Categorization of Different Voting Mechanisms. 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI). doi: https://doi.org/10.1109/sti50764.2020.9350399
12. Fernandes, A., Garg, K., Agrawal, A., Bhatia, A. (2021). Decentralized Online Voting using Blockchain and Secret Contracts. 2021 International Conference on Information Networking (ICOIN). doi: https://doi.org/10.1109/icoin50884.2021.9333966
13. Golubitskiy, S. (2019). Mutnaya tekhnologiya. Uroki moskovskih vyborov na blokcheyne. Novaya gazeta, 111. Available at: https://novayagazeta.ru/articles/2019/09/30/82175-mutnaya-tehnologiya
14. Schneier, B. (2020). Voatz Internet Voting App Is Insecure. Schneier on Security. Available at: https://www.schneier.com/crypto-gram/archives/2020/0315.html#cg1
15. Vyshnyakov, V. M., Komarnitsky, O. A. (2019). Transparentnye sistemy elektronnoy demokratii. Ottawa: Accent Graphics Communications & Publishing, 96. Available at: http://www.asdev.com.ua/dndiasb/assets/files/Vyshnyakov/e-voting.pdf
16. Chupryn, V., Vyshniakov, V., Komarnitskiy, O. (2018). Method of counteraction of attackes of mediator in transparent system the internet voting. Ukrainian Information Security Research Journal, 20 (3), 180–187. doi: https://doi.org/10.18372/2410-7840.20.13079
17. Shannon, C. E. (1949). Communication Theory of Secrecy Systems. Bell System Technical Journal, 28 (4), 656–715. doi: https://doi.org/10.1002/j.1538-7305.1949.tb00928.x
18. Chupryn, V., Vyshniakov, V., Prygara, M. (2016). Method of generation of casual numbers on the basis of the use of apparatus of the computer plugged in the Internet. Ukrainian Information Security Research Journal, 18 (4), 323–335. doi: https://doi.org/10.18372/2410-7840.18.11085
19. Diffie, W., Hellman, M. (1976). New directions in cryptography. IEEE Transactions on Information Theory, 22 (6), 644–654. doi: https://doi.org/10.1109/tit.1976.1055638
20. Chupryn, V., Vyshniakov, V., Prygara, M. (2017). Method of combating illegal influence on voters in the Internet voting system. Ukrainian Scientific Journal of Information Security, 23 (1), 7–14. doi: https://doi.org/10.18372/2225-5036.23.11547
21. Khlaponin, Y., Khalifa, E. K., Khlaponin, D., Selyukov, A., Tolbatov, A., Tolbatov, V., Odarchenko, R. (2019). Method of Improving the Stability of Network Synchronization in Multiservice Macro Networks. Proceedings of the International Workshop on Cyber Hygiene (CybHyg-2019) co-located with 1st International Conference on Cyber Hygiene and Conflict Management in Global Information Networks (CyberConf 2019). Vol-2654. Kyiv, 786–797. Available at: http://ceur-ws.org/Vol-2654/paper61.pdf
22. Khlaponin, Y. I., Khoroshko, V. O., Khokhlacheva, Y. E., Gavrilko, E. V. (2017). Parametric monitoring of computing processes in information and computing systems. Selected Papers of the XVII International Scientific and Practical Conference on Information Technologies and Security (ITS 2017). Vol-2067. Kyiv, 125–131. Available at: http://ceur-ws.org/Vol-2067/paper18.pdf
23. Pro zatverdzhennia planu zakhodiv shchodo realizatsiyi Kontseptsiyi rozvytku elektronnoi demokratii v Ukraini na 2019-2020 roky. Vid 12 chervnia 2019 r. No. 405-r. Kyiv. Available at: https://zakon.rada.gov.ua/laws/show/405-2019-%D1%80/sp:max10#Text
24. Schneier, B. (2004). What's Wrong With Electronic Voting Machines? Schneier on Security. Available at: https://www.schneier.com/essays/archives/2004/11/whats_wrong_with_ele.html
25. Vyshniakov, V. M., Pryhara, M. P., Voronin, O. V. (2014). Vidkryta systema taiemnoho holosuvannia. Upravlinnia rozvytkom skladnykh system, 20, 110–115. Available at: http://urss.knuba.edu.ua/files/zbirnyk-20/22.pdf
26. Cgernyshev, D. O., Khlaponin, Y. I., Vyshniakov, V. M. (2020). Experience of introduction of electronic voting in higher education institutions. Zbirnyk naukovykh prats Viyskovoho instytutu Kyivskoho natsionalnoho universytetu imeni Tarasa Shevchenka, 68, 90–99. Available at: http://nbuv.gov.ua/UJRN/Znpviknu_2020_68_12