Cross–Platforming Web–Application of Electronic On–line Voting System on the Elections of Any Level

Evgeniy V. Palekha
Information System Cybersecurity Chair
Don State Technical University
Rostov–on–Don, Russia
palekha1994@mail.ru

Irina S. Trubchik
Mathematics and Computer Sciences Chair
Don State Technical University
Rostov–on–Don, Russia
trubchik@mail.ru

Olga N. Manaenkova
Mathematics and Computer Sciences Cair
Don State Technical University
Rostov–on–Don, Russia
manaenkova_o@mail.ru

Olga A. Safaryan
Information System Cybersecurity Chair
Don State Technical University
Rostov–on–Don, Russia
safari_2006@mail.ru

Vitaliy M. Porksheyan
Applied Mathematics Chair
Don State Technical University
Rostov–on–Don, Russia
spu–46@donstu.ru

Sergey A. Morozov
Information Systems and Radioengineering
Don State Technical University
Rostov-on-Don, Russia
andrey@sssu.ru

Larissa V. Cherckesova
Mathematics and Computer Sciences Cair
Don State Technical University
Rostov–on–Don, Russia
chia2002@inbox.ru
ORCID 0000–0002–9392–3140

Boris A. Akishin
Mathematics and Computer Sciences Cair
Don State Technical University
Rostov–on–Don, Russia
akiboralex@mail.ru

Abstract — this paper presents the practical implementation of the electronic voting Web–system. Analysis of the developed algorithm was carried out, which was taken as basis in practical implementation of software product. Developed software product can be used for electronic voting at the elections of any level.

Keywords: electronic voting, elections of any level, software product, web – system, program application, El–Gamal encryption system, digital signature.

I. INTRODUCTION

Currently, the conduct of voting by hand counting ballots is a rather laborious, resource-intensive and scrupulous process. The organization of elections at polling stations is time-consuming and may not always ensure the transparency and integrity of elections. In addition, some citizens, due to compelling circumstances, cannot arrive at the place of voting.

At this time, computers are used everywhere, so why not apply them to such an important event as the holding of elections? Moreover, usage of computer systems will not only make the voting process more convenient, but will also be able to protect it from falsification of results. Consider the system of electronic voting.

This system involves three parties: voters, moderators and administrator. None of the moderators is not able to replace the voice of a voter. It is possible to prevent change of result of vote by use of the strengthened version of cryptosystem El Gamal, and also by a ban of access of the administrator and moderators to this or that table of a database. Thus, it is possible to reduce the probability of vote rigging [1].

To spoof the results, the attackers will not only need to access the database, but also to hack into the cryptosystem used.

II. TASK DEFINITION

The application has been developed, which organizes the voting process. In order for the application to function correctly on all popular operating systems, it was decided to develop a web application, as it will provide maximum availability and eliminate the difficulties in the development for each individual operating system, whether Windows, Linux, MacOS, Android, etc. The purpose of this application is to ensure the voting process and control over the voting process.

III. THEORETICAL BASE

Cryptosystem, which is used in this application, is based on an enhanced version of the scheme El–Gamal. The substantiation of the scheme complexity was given in the article by A.S. Mazurenko and N.S. Arkhangel'skaya [1].

Some parameters are generated according to this scheme, the rest parameters are generated by the administrator, except in specified cases. Inspectors are divided into teams t of people, considering that $n=kt$, where $n \in N$ is some natural number, $k \in Z_+$ – positive integer. Decryption requires the participation of t inspectors, where $2 \le t \le n$.

X secret key generation occurs. Next, divide x into secret shares, which will receive each of the results of the vote. The administrator then publishes the second part of the public key as $k_0=(p, g, y=g^x)$, which will be used by voters to encrypt their vote [2].

Some number of candidate ($v \in N$) candidates participate in the elections, voters can vote only for one of the candidates. The voter votes, his vote is encrypted, and the resulting cipher text is sent to the people checking the election results. After the voting, all the inspectors, who received all the cipher texts-voices, restore the secret key and decrypt the vectors – voices. Then all the vectors are summed up and the resulting vector is declared as the result of voting.

The complexity of hacking the constructed cryptosystem is equivalent to the complexity of solving the universally recognized difficult problem of Diffie–Hellman decision – making in the group G.

Although this cryptosystem is reliable, however, its practical implementation is extremely labor–intensive. Accordingly, only part of the scheme, namely the digital signature of El–Gamal, was used in the development of the web application. The third party participating in the scheme developed by A. Mazurenko and N. Arkhangel'skaya were inspectors, at practical implementation they were replaced

978-1-7281-1003-5/19/$31.00 ©2019 IEEE

on moderators. El–Gamal's digital signature is applied by administrator to voter's unique identifier and voice.

IV. Practical Implementation

Web application is a client – server application in which a client accesses a web server through a browser and receives a response in the form of an HTML page [3].

Data is primarily stored on the server, and information is exchanged over the network. The developed application is based on the Model –View – Controller concept [4].

The General scheme of MVC is shown in the Figure 1. This is scheme for dividing the application data, user interface and logic of his work on three components:

- Model provides data and responds to controller commands by changing its state;
- View displays the resulting data from the model to the user; react to changing the model;
- Controller is a link between model and view, notifies model of user's response.

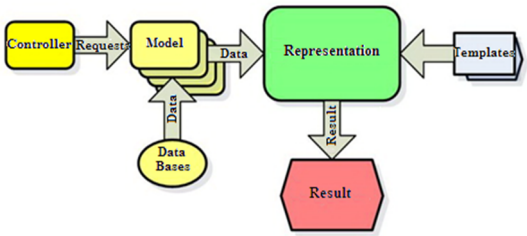

Fig.1. General Scheme of the MVC Concept.

This application uses one of the concept modifications – hierarchical, namely HMVC [4]. It is typically used with object – oriented language tools. Its essence lies in the fact that for each of the concepts three main classes are described: a common model class, a common controller class and a common presentation class [5].

Next, the development describes the new models that inherit the properties and methods of the General class of the model. Similar actions apply to new model and view classes. Consider the root directory of the application that contains the project, it is shown in the Figure 2.

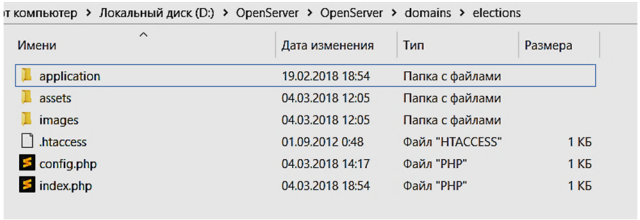

Fig. 2. Project root directory.

The MVC concept implies one entry point – the index.php. This file is the backbone of the project, through this script will pass all requests to the web application and all the logic of the project. In order to implement this approach, you must configure the server. It is assumed that the site runs on the apache server, this requires a file.ht access, which contains URL routing rules.

Routing will also allow you to create human – friendly URLS, so that the address of the pages will look like: project.ru/view/action Oh. This example calls the action method of the view controller class.

Config.php is a project configuration file that contains data, such as the full project path, as well as database connection constants (user, password, host, database name). The assets folder contains style files and js scripts, and the images folder obviously contains images for the project.

The core of the project is the application folder – it contains the whole MVC concept structure, the contents of the application folder are shown in the Figure 3.

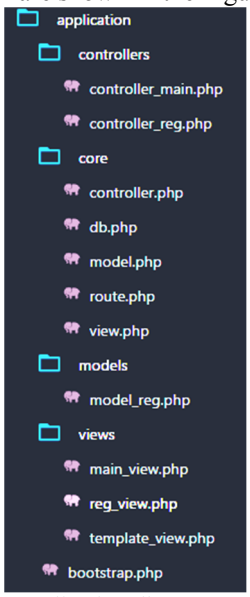

Fig.3. Application directory content

This folder contains the bootstrap.php. It collects all the files from the project core. All files from the folder core is connected to the bootstrap file.php and called the router, which will guide the application. The router class is described in the route.php and shown in the Figure 4.

The task of the router is to select the desired controller from the address bar, call the specified method of this controller and provide to user the output result using the specified view [6].

```
//  контроллер и действие по умолчанию
$controller_name = 'Main';
$action_name = 'index';
$routes = explode('/', $_SERVER['REQUEST_URI']);

// получаем имя контроллера
if(!empty($routes[1]))
{
    $controller_name = $routes[1];
}
if(!empty($routes[2]))
{
    $action_name = $routes[2];
}
//  добавляем префиксы
$model_name = 'Model_'.$controller_name;
$controller_name = 'Controller_'.$controller_name;
$action_name = 'action_'.$action_name;
//  подцепляем файл с классом модели (файла модели может и не быть)
$model_file = strtolower($model_name).'.php';
$model_path = 'application/models/'.$model_file;
if(file_exists($model_path)){
    include 'application/models/'.$model_file;
}
$controller_file = strtolower($controller_name).'.php';
$controller_path = 'application/controllers/'.$controller_file;
if(file_exists($controller_path))
{
    include 'application/controllers/'.$controller_file;
}
else
{
    exit('Файл контроллера не найден');
}
//  создаем контроллер
$controller = new $controller_name;
$action = $action_name;
if(method_exists($controller, $action))
{
    //  вызываем действия контроллера
    $controller->$action();
```

Fig. 4. Route class code.

To simplify code development and readability, controller, model, and view names have the same names, except for prefixes. It is also taken into account that the address can be deliberately changed and the desired controller does not exist, it uses the default value, or there is no specified method, then an error message is displayed. It is worth noting that the model file may not be, and is required only the presence of the controller.

The Controller General class contains a constructor that declares the view class. It also describes the action index () abstract method [6], which is required for each individual controller to perform its tasks during inheritance.

The description of the General view class contains only the generate method ($content_view, $template_view, $data = null). This method connects the $template_view page template, the content of which is filled with the $content_viewpage, and the $data parameter is an array that contains the data retrieved from the model [6]. It is worth noting that the default page template is template_view.php, the code of which is described in Figure 5.

```
<!DOCTYPE HTML>
<html>
    <head>
        <title>Выборы лучшего кандидата</title>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1" />
        <!--[if lte IE 8]><script src="<?=SITE_PATH?>assets/js/ie/html5shiv.js"></script><![endif]-->
        <link rel="stylesheet" href="<?=SITE_PATH?>assets/css/main.css" />
        <!--[if lte IE 8]><link rel="stylesheet" href="<?=SITE_PATH?>assets/css/ie8.css" /><![endif]-->
    </head>
    <body>

        <?php include 'application/views/'.$content_view; ?>

    <!-- Scripts -->
            <script src="<?=SITE_PATH?>assets/js/jquery.min.js"></script>
            <script src="<?=SITE_PATH?>assets/js/jquery.scrolly.min.js"></script>
            <script src="<?=SITE_PATH?>assets/js/jquery.poptrox.min.js"></script>
            <script src="<?=SITE_PATH?>assets/js/skel.min.js"></script>
            <script src="<?=SITE_PATH?>assets/js/util.js"></script>
            <!--[if lte IE 8]><script src="<?=SITE_PATH?>assets/js/ie/respond.min.js"></script><![endif]-->
            <script src="<?=SITE_PATH?>assets/js/main.js"></script>

    </body>
</html>
```

Fig. 5. Code of the default template.

As you can see from Figure 5, the page template contains only the General structure of the HTML document, as well as the connection of the style file and js scripts. Also connects the contents of the required page from $content_view. A generic model class consists only of the get_data method, which is an abstract method. It also describes a single General method, which can be attributed to the model, so it is in it that the database is accessed. The db.php contains a description of this class, DB class is abstract, but it contains a lot of methods that will be useful to all classes that inherit these methods.

Using these methods, you can perform various database manipulations without using SQL in its pure form [7]. It is enough to describe a new class, the name of which will tell the General DB class which table from the database to work with. You also need to specify what conditions are needed to extract data from the database. For example, on which of the columns to conduct a search, or you can specify a limit on the issuance results, etc.

Therefore, on the main page of the application contains a list of candidates that voters can choose. This page is the default page whose controller is described in the controller_mainfile.php (this controller does not have its own model works independently). This file describes the Controller_main class, whose code is shown in Figure 6.

```
<?php
class Controller_main extends Controller
{
    function action_index()
    {
        $model = new DB_Candidates(); // создаем объект модели
        $usersInfo = $model->getAllRows(); // получаем все строки
        $this->view->generate('main_view.php','template_view.php', $usersInfo);
    }
}
```

Fig. 6. Class description Controller_main.

Here selects all the candidates in the voting from the database and calls the main_viewview.php. This file displays information about candidates and their brief description.

To participate in the voting, the voter must register. To do this, click on the link "Register". During the transition, there is a change of controller on Controller_Reg, which causes the registration form. When registering, you must specify the full name, date of birth, passport details, date of issue and TIN.

After filling in the data and sending the form, there is a void check, as well as checking the correctness of the entered dates, as well as series and passport number.

The passport series consists of four digits, the first of which contains information about the region in which the passport was issued. The region code is checked by the list, for example, Moscow has the code 45, and Rostov region corresponds to the code 60. This check is necessary in order to avoid a possible attempt to falsify the voting results by introducing a non-existent region code. The second 2 digits of the passport series indicate the year of issue of the document. This value can not be more than 18 or less than 97, as it was October 1, 1997 passports of the USSR began to replace on passport of the Russian Federation.

The passport number is 6 digits, and paired with a series is a unique identifier of the voter, matches in the database with the same series and the passport number is impossible, so voters are tested for matches to ensure that the voter could not vote more than once. The TIN must also be filled in according to the rules and consist of 12 digits.

Also, when registering, you must choose, in fact, your candidate. It uses a separate table in the database, which contains two columns: voter id, candidate id. They point to each other's conformity. It is worth noting that none of the moderators or even the administrator will not be able to change the values of this table, because they do not have privileges to modify the data in this table, only to add data [8].

For additional data protection, a special table is used, which also contains two columns: the voter id and the result of the El Gamal digital signature [9].

It is the administrator who signs the concatenation from the ID of the candidate and the series, and the passport number of the voter with his private key. This table has several purposes. The signature is used to verify the results of the vote, whether the voter actually chose this particular candidate. This check is necessary only when attempting to compromise the electronic voting system [10].

To do this, an attacker must have access to the root super user and replace the votes in the table. After the registration is complete, the voter goes to the voting results page. The results of the voting are presented in the form of a pie chart, which can be seen in Figure 7.

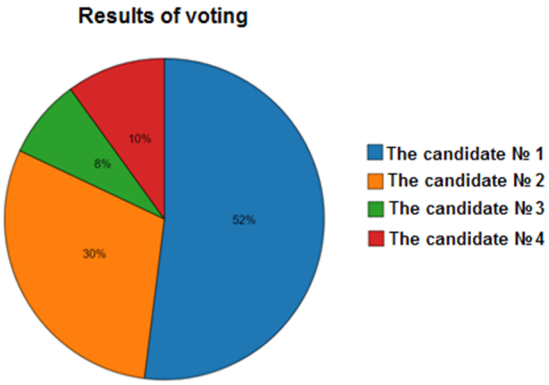

Fig. 7. Election Results.

Before counting votes, the system of voting checks whether the signatures match, and only after that the results of voting are displayed.

It should be noted that there may be cases when the voter accidentally entered his passport data incorrectly, thereby prohibiting the voter from voting, whose data he entered. This requires moderators who can change the data in the voter table. After changing the data on the series and passport number, the administrator again signs the concatenation of the id of the selected candidate and the series, and the passport number of the voter [11 – 13].

V. Conclusion

Thus, an enhanced version of the El Gamal cryptosystem was analyzed for the practical implementation of the electronic voting system. As a result, a web-based application for electronic voting has been developed, which ensures the simplicity, honesty and transparency of voting. The developed application functions correctly on all popular operating systems. In addition, this application contains protection against data spoofing, thus significantly reduces the likelihood that the system will be compromised [14], [15].

References

[1] A. Mazurenko, N. Arhangel'skaja, L. Cherckesova, O. Safaryan "Computational Complexity of Coding and Information Security System Based on Threshold Secret Sharing Scheme Used for Electronic Voting Systems". – Rostov–on–Don: DSTU, 2017. (In Russian).

[2] R. Barbulescu, P. Gaudry, A. Joux, E. Thom´se, "A heuristic quasi–polynomial algorithm for discrete logarithm in finite fields of small characteristic". In: P.Q. Nguyen, E. Oswald, (eds.) EUROCRYPT 2014. LNCS, Springer, Heidelberg (May 2014). Vol. 8441, Pp. 1–16.

[3] Free encyclopedia [Electronic resource]. – Access mode: https://ru.wikipedia.org/wiki (circulation date 25.02.2018).

[4] S. Rogachev, "A Generalized Model–View–Controller". – Moscow: K–Press, 2016. – Pp. 37–66. (In Russian).

[5] D. Kolisnichenko, "PHP and MySQL. Development of Web applications". – St. Petersburg: BHV–Petersburg, 2014. – 560 p. (In Russian).

[6] S. Prettyman, "Learn PHP 7: Object Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON, and MySQL". Apress; 2015. – 294 p.

[7] W. Stallings "Cryptography and Network Security", 7Th Edition". Pearson India; VII–th Edition, 2016.

[8] J.–P. Aumasson, "Serious Cryptography: Practical Introduction to Modern Encryption", 2017.–312 p.

[9] V. Stone, S. James, "Information Theory: A Tutorial Introduction". Sebtel Press, 2015. – 260 p.

[10] Ye. Lindell, "Introduction to Modern Cryptography. Chapman and Hall / CRC"; II edition (November 6, 2014). 603 p.

[11] S.A. Zheltov, "Effective Computing in the CUDA Architecture in Information Security Applications". PhD dis. / Zheltov S.A. – M: IINTB RSUH, 2014 – 145 p. (In Russian).

[12] P. Kocher, "Timing attacks on implementations of Diffie– Hellman, RSA, DSS, and other systems", Advances in Cryptology Crypto'96, Springer–Verlag, LNCS 1109, 1996, Pp.104–113.

[13] J. Sammons, M. Cross, "The Basics of Cyber Safety". Computer and Mobile Device Safety Made Easy". Syngress. I–st Edition. 2016. – 254 p.

[14] A.F. Chipiga, "Information Security of the Automated Systems". / A.F. Chipiga. – M.: Helios of ARV, 2010. – 336 p.

[15] T. Koppel "Lights Out: A Cyberattack, Nation Unprepared, Surviving the Aftermath". 2015. – 279 p.