ATLANTIS PRESS

Advances in Computer Science Research (ACRS), volume 54

International Conference on Computer Networks and Communication Technology (CNCT2016)

Authentication of Voters using the Domain Driven Design (DDD) Architecture for Electronic Voting Systems

Md. Abdul Based[1,*], Md. Mahabub Al Hasan[2], and Md. Mazidul Islam[3]

[1]Flat E-8, Rahman Tropical Tower, 52, 52/1-52/5, Puarana Paltan, Dhaka-1000, Bangladesh

[2]66 Green Road, Dhaka 1205, Bangladesh.

[3]66 Green Road, Dhaka 1205, Bangladesh

*Corresponding Author

Keywords: E-voting, Domain driven design architecture, Voter authentication.

Abstract. The primary goal of each voting system is to ensure that only the eligible voters should be able to cast their ballots to the respective candidates. Therefore, electronic voting systems need to provide a transparent and trusted authentication process. Traditional paper-based voting schemes involve manual authentication of voters and use paper ballots to cast by the voters. This manual procedure is susceptible to time wasting, ballot snatching, lacks voter privacy, and raises questions about the integrity of fair election. Thus, the electronic voting systems are becoming more demanding to overcome these flaws of paper-based voting schemes. This paper works on the authentication of the voter and proposes a new authentication process that is suitable for simultaneous access of millions of voters. In this authentication process, many voters can have simultaneous access to the voting web page without interruption while providing multifactor authentication. We particularly consider general elections in the populated countries like Bangladesh, China, and India where the number of voters is extremely high. The authentication process that is presented in this paper uses the Domain Driven Design (DDD) architecture. The voters' mobile number, National Identification (NID) number and biometric serial number are verified during the authentication process. The voting web page is designed using the Active Server Page (ASP.NET). The ASP page is then connected to the Microsoft Structured Query Language (SQL) server database. The ASP page is served from an Internet Information Services (IIS) server.

Introduction

Electronic voting (e-voting) has become more popular day-by-day to overcome the flaws of existing paper-based voting schemes. Authentication of voter is the first requirement for conducting an election either for e-voting or for paper-based voting. This process should be fair and secure enough so that only the eligible voters have access to cast their ballots. In case of e-voting, the voting web pages should be running properly during the voting period when multiple voters may access the voting web pages simultaneously. In the over populated countries like Bangladesh, China, India, it is experienced that the web pages do not work properly on some occasions when many users try to access the web pages simultaneously. For example, this particularly happens when millions of students try to get their results online and when millions of candidates apply for some public examinations online.

This paper ensures that only the eligible voters will be able to cast their ballots. In addition, this authentication process also provides simultaneous access of the millions of voters without interruption of the voting web pages.

High level security in the authentication process is provided using the Domain Driven Design (DDD) architecture. The voters' National Identification (NID) number, mobile number, and biometric serial number are verified during the authentication process. The implementation of the voting web site is done using the Active Server Page (ASP.NET). The ASP page is then connected to the Microsoft SQL Server database. The ASP page is served from an Internet Information Services (IIS) server.

Organization of the Paper

We describe the related works, architectures, technologies used, summary of design and development in the following sections. Then we show the results of our work, analyze the findings, write down the conclusions and recommend the further works. This paper concludes with the reference lists.

Related Works

The FOO92 [1] protocol is a well known protocol for e-voting. This protocol is based on blind signature scheme for voter authentication. Smartcard with biometric identification (fingerprint) is proposed for the authentication of voters in [2]. Pret-e-Voter [3], Punchscan [4], Helios [5], Votebox [6], Civitas [7] are various examples of e-voting systems. Almost all of these systems focus on the security of the authentication of the voter and the security of the voting process. If the voter number is very high, there is no discussion on that. However, in this paper, we primarily focus on the security of the authentication process and then consider the simultaneous access of millions of voters without interruption during the voting period.

Architectures

We use DDD architecture [8] in the authentication procedure of the voters. The DDD architecture is a layered architecture that can meet the complex needs of software development (particularly web development). It can connect the implementations to an evolving model. This architecture is also suitable for agile processes that require iterations, working closer with business partners, applying continuous integration, and working in a high-communication environment.

Technologies Used

We use ASP.NET Model View Controller (MVC) to build the voting web site. The MVC is a composition of three roles: model, view and controller. The MVC works with three logic layers namely model (business layer), view (display layer), and controller (input control). We have used SQL Server 2008 for storing data in the server. This particularly stores the information of the eligible voters prepared by the election commission before conducting the elections.

Design and Development

Domain Model Design. We consider **t**wo actors: voter and candidate. Many voters may cast ballots for a single candidate. Many candidates may have one flag. We assume that voters and candidates have one image. We also consider that candidates may have

multiple locations and many candidates may live in the same city. In addition, a city may have many locations.

Sequence Diagram Design. First, the voters need to login in order to get access to the voting web site. For this purpose, a voter should supply his/her mobile number, National Identification (NID) number, and bio-metric serial number. If these data match with the data previously stored in the database of the election commission (who controls the election), then a Short Message Service (SMS) will be sent to the voters including an auto generated code. A voter needs to use this code as a password. After that the voter will get access to the voting web page. The NID is unique. So a voter will have a single chance to get access on the web.

Database Design. There are eight tables in the database. Table 1 contains voter information like mobile no, NID, bio-metric data. Table 2 is designed for storing detail address of the voters. Table 3 is for storing location, Table 4 is for storing city, Table 5 is for storing candidate information, Table 6 is for image, Table 7 is for Flag and Table 8 is for result.

Voter Interface Design. Voter Registration Form - A voter can register through User Interface (UI) to fill the voter registration form. The voter should supply mobile number, NID, address, location, and biometric serial number. Then the voter needs to press the submit button.

Voter Verification Form – This form will verify the authenticity of the voter.

We deploy the web page in windows azure.

Result Analysis

This paper provides authentication of voter by using some credentials (mobile number, NID, biometric serial number) in the DDD architecture. This architecture encrypts the data. Thus no one can manipulate these data. We use windows azure (cloud server) which can handle maximum simultaneous user requests on the web site. This ensures that the web site will not be hanged or interrupted during the voting period.

If a voter tries to access the web twice or more, he/she will not be able to succeed because the voter's NID number, biometric serial number, and mobile number are stored in the database. These records get verified and prevent the voter to get authentication more than one time even though a voter may have more than one mobile number and more than one biometric serial number.

If a hacker makes an attempt to make the voting web page busy, the attacker will not succeed. The windows azure server prevents Denial of Service (DOS) attack and can handle maximum user requests. This can get rid of the web page down. Even if an attacker tries to attack the client side script, he/she will not succeed since the verification takes place in the server side. In addition, the hacker will not be able to implement SQL injection as the voting web site uses entity framework.

Limitations

The paper is written based on the following assumptions:

- We assume that alll voters must have their own mobile phones. Without mobile phones, voters will not be able to get access on the voting web site.
- We assume free internet access on the voting page during the voting period.

- The DDD architecture is not suitable for limited functioning applications since the architecture is developed for longterm applications. So we consider large scale elections where the number of eligible voters is extremely high.

Conclusions

In many countries, for example in Bangladesh, the government is spending a huge amount of money for conducting national and regional elections. A major part of the expenditure involves the authentication process. However, the average percentage of participation of voters is usually 60%-70%. Moreover, the opposition parties always question the fairness of the authentication process. The authntication process presented in this paper will overcome these limitations and the cost for conducting the election will be reduced. This will additionally provide an opportunity for those voters who are physically disable to come to the polling booth for casting ballots.

The authentication process in this paper allows only the eligible voters to get access on the voting web pages. A voter will not be able to get access on the web twice or more. More importantly, millions of voters will be able to access the voting web pages simultaneously since the web site will not be inturrupted.

Future Works

Electronic voting as a whole is a huge task. We have focused only on the voters‘ authentication part. Generation of electronic ballots and working on the privacy, receipt-freeness, and coercion-resistance issues will be done in future. Implementation of a comprehensive e-voting system followed by performance analysis will be useful future works.

References

[1] A. Fujioka, T. Okamoto, and K. Ohta. A Practical Secret Voting Scheme for Large Scale Elections. In J. Seberry and Y. Zheng, editors, Advances in Cryptology – AUSCRYPT ’92, Lecture Notes in Computer Science, vol. 718, pp. 244–251, Springer-Verlag, Berlin, 1992.

[2] M. A. Based, J. K. Tsay, and S. F. Mjølsnes. PEVS: A Secure Electronic Voting Scheme using Polling Booths. Y. Xiang et al. (Eds.): ICDKE 2012 (Colocated with NSS 2012), LNCS 7696, pp. 189–205. Springer-Verlag, Berlin, 2012.

[3] P. Y. A. Ryan and S. A. Schneider. Pret-e-Voter with Re-encryption Mixes. In Proceedings of European Symposium on Research in Computer Security, Sept. 2006.

[4] S. Popoveniuc and B. Hosp. An Introduction to Punchscan. In Proceedings of Workshop on Trustworthy Elections, June 2006.

[5] B. Adida. Helios: Web-based Open-Audit Voting. In Proceedings of the Seventeenth Usenix Security Symposium, pp. 335–348, USENIX Association, 2008.

[6] D. Sandler, K. Derr, and D. S. Wallach. Votebox. A Tamper-Evident, Verifiable Electronic Voting System. In Proceedings of the 17th Conference on Security Symposium (SS’08), pp. 349–364, Berkeley, CA, USA: USENIX Association, 2008.

[7] M. R. Clarkson, S. Chong, and A. C. Myers. Civitas: Toward a Secure Voting System. In S&P'08: Proceedings of the 2008 IEEE Symposium on Security and Privacy, pp. 354–368, Washington, DC, USA, 2008. IEEE Computer Society, 2008.

[8] Domain Driven Design Quickly. Accessed on May 2016. Web: http://www.infoq.com/minibooks/domain-driven-design-quickly.