SILESIAN UNIVERSITY OF TECHNOLOGY

FACULTY OF AUTOMATIC CONTROL, ELECTRONICS AND COMPUTER SCIENCE

Engineer thesis

Central online voting system

author: Wojciech Drzewiecki

supervisor: Krzysztof Simiński, PhD DSc

Gliwice, January 2020

załącznik nr 2 do zarz. nr 97/08/09

Oświadczenie

Wyrażam zgodę / Nie wyrażam zgody* na udostępnienie mojej pracy dyplomowej / rozprawy doktorskiej*.

Gliwice, dnia 20 stycznia 2020

.....
(podpis)

.....
(poświadczenie wiarygodności
podpisu przez Dziekanat)

* podkreślić właściwe

Oświadczenie promotora

Oświadczam, że praca „Central online voting system” spełnia wymagania formalne pracy dyplomowej inżynierskiej.

Gliwice, dnia 20 stycznia 2020

.....

(podpis promotora)

Contents

Chapter 1

Introduction

Chapter 2

[Problem analysis]

2.1 Introduction

Ever since internet was introduced in our lives, online election voting was a contentious issue.

First of all, online voting would be convenient - right now voting is associated with going to a nearby ward, waiting in a queue, complicated procedure of receiving a ballot and filling a ballot. Moreover, waiting for results could be decreased - now we have to wait for all the votes to be counted, protocols sent to the commissions, summed up, and determination of the winners. All this can take over few dozen hours. Online voting would decrease that time hugely, at the same time being much more precise than counting by a person.

On the other hand, there is significant trust issue. There will always be a person which have access to data, and with such access that person could change elections results in a matter of seconds. Following this lead, even if nobody has access to data, we will never be sure that vote we cast is not changed somewhere in the logic of the application. We can eliminate that by developing open source application, which can be verified that it does not change something in the logic. This approach would create many more problems, of which two are: firstly, open source application can be investigated by everyone, and with given source it is much easier to find a hole in the logic from which we can benefit. Secondly, even if we see view of that open source application, we can never be sure that logic

behind it is not swapped.

You can see a pattern here - for every argument there is a counter argument, which generates another problems. There is no right answer in this debate.

2.2 Ideal voting system

So what would be features of the perfect voting system? Let's investigate it along with flow of online voting.

Before election, system administrator has to enter committees and candidates data into the system, register them for polls predicted for certain date, which he have entered at the beginning. Although he could handle registering candidates to certain committees, he cannot handle scope of managing whole committee during election. It's best if administrator granted some chosen user privileges to manage certain committee inside the system - let's call him committee administrator. It's useful for example in case of allocation seats via D'Hondt method - it's committee itself that wants to decide on order in closed list, there should be no third party involved.

First, in order to cast a vote through internet, a citizen would have to register. What online voting system wants to achieve, is for citizens to be able to cast a vote without leaving home. It makes sense that citizens would register also through the internet - for example in case of some injury, if someone could go out of home, they could also vote in a ward. It creates a problem - how do we know that person on the other side of the screen is who he says he is. Solution is two step authentication process - citizen registers with citizen id (for example PESEL in Poland), he is sent a text message or an email, but also receives registered letter of registration, both with activation codes, that only together allows him to register properly. Registered letter allows to verify citizen willing to vote through internet without need of leaving the house. Because email can be generated without interference of any third party person, code sent this way secures citizen from someone taking advantage of code in letter. Those two ways combined secure citizens account from being taken advantage of.

Let's say a citizen registered successfully. On election day, he should be able to vote on exactly the same polls as he would personally, in ward. Citizen should be

able to cast a vote only once, and his vote should be detached from his account - in no way should anybody be able to tell which vote belongs to which citizen. Although registered for online voting, a citizen should still be able to vote personally in ward. However, once he voted in any of two ways, the second one should be automatically and immediately blocked. This requires for the system to work also in wards. Once user votes online or receives a ballot from commission in a ward, he should be immediately blocked for second type of voting.

Citizen easily and securely voted either through internet or personally. Now his journey ends, he can go back to living his life. What's left to do is to calculate results of election. However, system also has to take into account all votes casted in wards. Results can be calculated if and only if all wards have entered their protocols into the system. To do so, a member of commission should be assigned with access to send the protocol to the system - let's call him ward administrator. Once that member is a registered user, he should be able to be assigned as particular ward admin by the system administrator. Then, he should be able to file in the protocol, once the election is closed for voting.

After closing the election and collecting protocols from all wards, results of the election can be calculated. This information can be public and delivered to public on the same site that they voted on, as well as on official government electoral commission website.

2.3 Problem statement

2.3.1 Overview

Let's put above functionalities into more technical perspective. In order to secure application from unauthorized access, it is best if we divide application into four main cockpits:

- Voter cockpit
- Administrator cockpit
- Ward administrator cockpit

- Committee administrator cockpit

Each cockpit will have certain role that is necessary to even display available actions. In order to have access to anything, we have to be authenticated (active, successfully registered) user, with assigned role that gives us permission for certain cockpits. A guest - not registered user - is only available to register or login.

2.3.2 Voter cockpit

User with role of a voter should be able to vote for election, and see election results, after it's closed. In case of voting, user must be able to vote on a poll only once, without any exception.

2.3.3 Administrator cockpit

User with role of an administrator should be able to create essential data - wards, polls, committees etc. as well as granting certain roles to certain users. Administrator should not be able to interfere in ward or committee administration, other than choosing its administrator. Moreover, administrator should be the one to trigger polls results calculation.

2.3.4 Ward administrator cockpit

User with role of ward administrator should be able to manage only his assigned ward. Under no circumstances should he be able to manage other wards. In ward administrator cockpit he should be able to enter protocol from ward for each poll during an election.

2.3.5 Committee administrator cockpit

User with role of committee administrator should be able to manage only his assigned committee. Under no circumstances should he be able to manage other committees. In committee administrator cockpit he should be able to choose order of committee candidates in closed list.

2.4 Overview of similar solutions

2.4.1 Electronic voting in Estonia

In Estonia, voting is based on electronic citizen ID. It's mandatory and sufficient national identity document, which allows for secure remote authentication. To cast a vote, an Estonian needs a computer connected to the internet and equipped with card reader. They can authenticate using digital certificate included in their citizen card, and cast a vote on the internet.[1]

This solution is problematic for several reasons:

- It's impossible to be easily implemented in countries without electronic citizen ID. Providing such documents for most of society of the country is a long time process, measured in years rather than months.
- Card reader capable of reading such card is not a common device - a citizen has to go somewhere with a card reader, if he wants to cast a vote.
- Electronic citizen ID is not an intellectual knowledge like password or token. It's a physical device that can be easily stolen - and of course blocked in some department - but it may be too late and someone may have already taken advantage from owning someone else's citizen ID.

Chapter 3

Requirements and tools

Chapter 4

External specification

-

Chapter 5

Internal specification

-

Chapter 6

Verification and validation

-

Chapter 7

Conclusions

-

Bibliography

[1] Jan Willemson Sven Heiberg. Verifiable internet voting in estonia. https://research.cyber.ee/~janwil/publ/mobileverification-ieee.pdf. [access date: 2020-01-20].

Appendices

List of abbreviations and symbols

-

Listings

Contents of attached CD

The thesis is accompanied by a CD containing:

- thesis (LaTeX source files and final `pdf` file),
- source code of the application,
- test data.

List of Figures

List of Tables