Proposal of a voting system that attempts to be sufficiently secure when the stakes are low

Year: 2024
Author: Stavros Kalognomos

1. Introduction

Online voting systems are increasingly adopted as a solution for decision making. On the other hand, their high vulnerability on manipulating the results, make them totally unsuitable for decision making when the stakes are high, e.g. for national elections. There are questions whether and in which conditions they can become suitable for decision making in smaller groups like professional chambers, unions, movements, groups of activists etc. This article proposes a voting system that attempts to be sufficiently secure when the stakes are low.

2. Vulnerabilities in online voting systems

Online voting systems suffer from serious vulnerabilities [1]. It is very easy to alter massively the results with a few lines of code. These changes can become non detectable. Furthermore, the anonymity of the voters can also be jeopardized. Even block-chain technology that it is falsely assumed to be more suitable for online voting, makes things more complicated and less safe. In reality, these online systems do not interact with voters, but with devices that it is assumed that are used by legitimate and free to decide voters. This assumption can become not valid.

3. The proposed voting system

For a voting system to be fair, equal opportunities and time should be given to every candidate, in order to prepare and expose his/her ideas. Equal and adequate opportunities should also be given to every member to explore and evaluate these ideas.

A voting system, must also check that each member has only one account (e.g. that each member corresponds to one valid email) and that he/she can vote only once at a specific voting occasion {1}.

The votes cannot be secret from the side of the server, in an online voting system. However, in order to keep the votes secret from the side of the devices of the members, an alias (e.g. a random eight digit number) can be returned to the member who just voted, in order to be used as his/her identity. In this proposed voting system, a secret code is also sent to him/her that will be used later. This is the step 1 of the proposed procedure.

Step 1: When a member votes successfully, an alias and a secret code is sent to him/her {2}.

Step 2: After the closing of the "ballot box", a pdf with the temporary results is published {3}, that includes two tables:

- The first table has a list that each row corresponds to a member that has voted, displaying the following information:
  - The alias of the member (the username is not displayed, in order the secrecy of the vote to be kept).
  - The vote that corresponds to this alias.
  - The hash number that corresponds to the string of the alias concatenated with the string of the secret code (that was sent with this alias).
- The second table has a list with the usernames of the members that have voted, in a random order (so that there is not a correlation of any member with a specific vote).

A team of randomly selected members should verify that each username, alias, hash-number is unique in the tables {4}. If this is not the case then these rows should be highlighted as invalid. It should also check that the number of rows of the first table equals to the number of rows of the second table. If this is not the case, then a serious mistake has occurred in the process, and if it is not correctable, the voting event is needed to be repeated (perhaps with another method that might be more costly/cumbersome, but also will be more robust, e.g. postal voting).

For a certain period after publishing the temporary results, every member is invited to review the temporary results and post a complaint to a special area in the forum {5}, if his/her vote is not cast correctly.
The following cases might appear as a complaint:

A. The username of the member does not appear in the second table.

B. The username of the member appears in the second table, but one of the following errors appear to the first table:

1. The alias that was sent to the member does not appear to the table.
2. No alias or/and secret code was sent to the member when he/she voted.
3. The member lost/forgot the alias or/and secret code that was sent to him/her.
4. The alias combined with the secret code does not provide the hash number that appears in this row of the table (either the secret code or the hash number is wrong).
5. The vote is not cast correctly (or the member claims that his/her vote is not cast correctly).
6. The member claims that he/she did not actually voted (either by choosing not to vote or that he/she was not allowed to vote, despite his/her username appears in the second table).

In this proposed voting system, each of the above cases is treated differently. In case A, the member is allowed to cast his/her vote, eponymously, via a post (knowing that his/her username will be displayed in the post) {2}.
In case B, the administrator reveals, via a post, the row of the first table that corresponds to the username in the second table of the member that made the complaint. After the period for submitting the complaints ends, a second period starts that no more complaints are taken into account. During the second period, the members are invited to review the rows that the administrator revealed by revealing the usernames {6}.

For each challenged row (that the username is revealed), one of the following cases might appear:

I. No member provides the secret code, which combined with the alias gives the hash-number that appears in this row. Perhaps one of the cases B1 to B4 or case B6 has occurred.

II. A member provides the secret code which combined with the alias gives the hash-number that appears in this row. This member might:
   a. be the member that made the complaint. Perhaps it is the case of B5.
   b. be a member that his/her username appears on the second table and the row that contains his/her alias is the challenged row of the first table.
   c. be a member that his/her username does not appear on the second table (but somehow knows the secret code).
   d. It might be more than one members that provide the correct secret code.

In case I, the challenged row becomes invalid, and the vote in it is removed. The member that made the complaint is allowed to vote eponymously, via a post (during a specific period). It might be the case that he/she did not want to vote and that he/she still does not want (sub-case of B6).

In case II.a too, the challenged row becomes invalid, and the member that made the complaint is

allowed to vote eponymously via a post.

In cases II.c and II.d too, the challenged row becomes invalid, and the vote in it is removed (either the error was from the side of the server or the secret code was leaked).

In case II.b the challenged row is apparently validated by a member that his/her username appears in the second table (the row should not become invalid, since it is validated by only one member, that appears in the second table, and he/she has not filed a complaint that the information in this row is wrong). In this case, apparently the claims of the complaint that challenged this row are not valid.

It should become clear that every member who has voted, should observe the phase of step 2. His/her vote might not be cast if someone else claims his/her alias, and he/she does not provide the secret code that will prove whose this alias belongs to. In addition, members that have not participated in this voting occasion should at least check the second table to ensure that their username is not in it (the server could easily add fake rows by using usernames of inactive members, if it is affected by a malicious software) {7}.

Step 3:
After the period that allows the members who complained to cast eponymously their vote via a post, a team reads these posts and prepares the pdf with the final results. In this report, the following tables should appear:

- The first table (with the rows that became invalid, clearly marked, and the links of the posts that resulted to the invalidation of the row, displayed as information inside the row).
- The second table (with the usernames that correspond to the rows of the first table that became invalid, clearly marked).
- A table with the members that voted eponymously via posts (and are considered valid, concerning the procedure of step 2). This table should have in each row the following information:
  - The username of the member that voted.
  - His/her vote.
- A table with the summary of the results that will display at least:
  - The number of the members that have voted.
  - The number of the members that have voted eponymously.
  - The sum of the votes for each option that the voters had.

This pdf should be sent to a team of randomly selected auditors, so that the auditors can check the following:

- Each alias, username, secret code, hash-number is unique.
- When excluding the rows that have become invalid, the number of rows of the first table equals to the number of row of the second table.
- That the procedure of step 2 has been followed correctly and that every row from the first table that has become invalid, has become so according to the procedure.
- That the sum of the votes is the correct sum of the votes of the valid rows plus the votes of the third table (eponymous votes).

If that is not the case, then the pdf should be returned to the team that created it, for corrections.

If the pdf has passed these checks, then it should be published as the final results.

{1}. The authentication of each member can be done at a local office (e.g. via showing his/her ID card) or using other credible methods.

{2}. This proposed voting system does not protect the secrecy of those votes that mistakes have been detected after their casting. It is assumed that when the number of mistakes is small and the mistakes can be corrected, then the voting event can be considered credible and that due to the low-stakes of the voting occasion, no coercion or vote-buying is expected for such small disclosure of votes. An alternative method that protects better the secrecy of each vote, is proposed in chapter 5.

{3}. The file of the results should be available to be downloaded by any member. This file should not be generated by the server, every time a team member requests it (because it would be easily subjected changes by a malicious software). It should rather be one file (e.g. a pdf), and available to be downloaded, preferably from a separate platform than the one of the voting system.

{4}. Three independent teams are proposed to be formed for this voting system. The first one is responsible for running the voting event and creating the files with the temporary and final results. The second one is responsible for auditing the results of the first team before they are published. The third one is responsible for auditing the hardware and software which is used in the voting process. In order for these teams to be truly independent, it is essential that their members are randomly selected from a pool of well-trained members for these tasks, (and not just appointed by an authority).

{5}. Administrators should redirect complaints that were posted in other areas of the forum by mistake, to the area of the complaints for the specific voting occasion. Members should file their complaints during a specific period (complaints that have been posted much later than the voting event and its audit phases, cannot become useful).

{6}. It is essential to be explicitly explained to all members that they should follow the whole voting procedure in order to be sure that their votes are cast properly.

{7}. Each member should be encouraged to look on the results of each voting event in which he/she did not participate, to check that his/her username does not appear on the second table of the results (i.e. to check that he/she does not appear to have voted). It is assumed that this will not be considered as a time-consuming task (since it is only a search in these tables).

4. Online voting systems are highly insecure and unsuitable for voting occasions that the stakes are high

It is important to mention again that online voting systems cannot become adequately secure for elections that the stakes are high. There must always be a procedure, in case that a specific online low-stakes voting occasion is not considered to be credible/valid. This procedure might be, for example, to repeat the online voting event (after a thorough auditing and corrective actions, such as installing again the software of the online voting system). It could be, to repeat the elections via postal voting this time. Members should be trained that this incident can occur, and that mistakes should not be hidden. They should also be trained on how to use the alternative voting method.

It must be highlighted that if the number of invalid rows and the number of the members that voted eponymously are high, then the elections should be repeated. Whether it is because the hardware/software produced errors, the system was hacked, many members acted maliciously and posted fault complaints, or simply many members saw it as an opportunity to change their minds and vote again for something different, it does not really matter. For the shake of implementing a truly democratic procedure, the elections should be repeated, independently of how much inconvenience this will bring. Even if this decision makes some members to question the validity of

the procedure, this is better than making them think that online low-stakes voting systems are always secure.

5. Possible improvements on the above suggested voting method:

1. In order to enhance the secrecy of the votes, the complaint-posts can be eponymous (by revealing the username) only to the audit team and the team that prepares the pdfs of the temporary and final results. These posts could appear to the rest of the members as anonymous_1, anonymous_2 e.t.c. without revealing the real usernames.

2. The second suggestion tries to reduce possible coercion, by enabling each voter to keep secret when he/she actually voted. When a member tries to vote for a second time, the voting system can pretend that it accepts these entries by providing an alias and a secret code that will not be used. The system will only count the entries that the member sent when he/she voted for the first time for this voting event. In practice, this can become very confusing, and could only work if the members are well trained to keep only the first alias and secret code that were sent to them. On the other hand, when combined with improvement 1, it can reduce coercion, since no one can really know when a member actually votes (except from the teams that run/audit the voting event), making pressure at a great number of members on what to vote, impractical.

3. The third improvement tries to handle the following situation. A lot of members might choose to vote eponymously (by filing fault complaints that their votes were not cast correctly), aiming only to repeat the elections. For example, we can consider the case that the voting occasion concerns a "yes"/"no" decision, and the criterion to repeat the voting event is how many members voted eponymously compared to the difference between the votes for "yes" and the votes for "no". In this case, the members that voted for the option which lost in this voting event, might have the incentive to vote again eponymously in order to force the repetition of this voting event. This could be avoided if the members that voted eponymously for the option that lost in the elections are not considered concerning this criterion. For instance, (in the case that the "yes" option wins, and only eponymous votes for "yes" option are considered concerning the criterion for repeating elections), if the members that have voted for "no" choose to vote "yes" eponymously (by filing a complaint), the difference between the votes for "yes" and the votes for "no" will become greater, and thus it will become more difficult for the criterion for repeating the elections to be triggered. More simply, in one case, the criterion is: eponymous votes > |votes for "yes" - votes for "no"| * ratio_factor, where in the other case, the criterion is: eponymous votes for the winning option > |votes for "yes" - votes for "no"| * ratio_factor.

4. It may be meaningful, the voting system platform and platform for the forum (that the complaints are posted) to be in different servers, and administrated by different teams. Separating the systems makes it harder implementing hacks/frauds to/from the side of the servers.

5. There should be an effort to engage members in the discussion that takes place before the voting event, especially those that tend to only read and vote, without contributing to the discussion. If there is a high percentage of members that only vote, then it is difficult to monitor how many members are permanently inactive, and appear to vote due to a malicious code in the server.

6. For each long proposal, there should be an effort to be broken down in smaller parts, instead of giving to the members the only options to "take it or leave it". This might not concern the security of the voting system, however, it is essential in order the whole procedure to be meaningful. It is equally essential that each phase of the procedure lasts a sufficient time.

7. For all complaints, it should be clearly and publicly stated, how each one of them was handled and why. A registry number assigned to each complaint, could make this procedure easier.

8. After a period of publishing the final results, the data in the database for this voting event should be deleted.

9. After each voting event, the audit team should have a meeting with the people that voted eponymously, in order to understand the true reasons behind this. This dialogue can contribute to the improvement of the procedures of the voting system. It can also help these members to understand that the difficulties they might had with the voting system are heard. It can help them comprehend better the procedures, and make them more confident about the operation of the voting system.

10. A statistical approach can also be used to enhance some aspects of this voting system [2].

11. Zero knowledge proof might come in use in voting systems in the future [3]. However, a voting system that is not easily understood by its voters, can easily produce false assumptions about its security.

[1]. https://dci.mit.edu/voting-on-the-blockchain
Going from bad to worse: from Internet voting to blockchain voting. Journal of Cybersecurity, 2021, 1–15

[2]. https://en.wikipedia.org/wiki/Hypergeometric_distribution#Application_to_auditing_elections

[3]. https://en.wikipedia.org/wiki/Zero-knowledge_proof