**CSCI 531 Semester Project**

**Designing a Secure Electronic Voting System**

1. **The** **nature of the assignment**

The semester project gives each student the opportunity to use and illustrate the concepts from the course in an applied manner. Based on information you gather and review, you are to report your design, analysis, and prototype of a secure electronic voting system.

In this project, you will have an option to either concentrate more on the on the detailed design of the system or on the implementation of the system prototype.

In not less than 7 and no more than 10 pages prepare a report in PDF format with a font size between 10 and 12. Figures, tables, and the like are not included in the 10-page maximum page count, but text beyond the 10-page limit will not be considered in grading. There are no specific requirements related to the formatting of your report. Submit the report and your code in electronic form on DEN D2L.

1. **Project description**

In democratic countries the process of voting is a very important event where people choose new government to rule their country. Electronic voting makes the voting process simpler for the government and for the citizens.

**Simplified voting process steps:**

1. Registration is the first step to disallow unregistered people to participate in the vote. During registration, each potential voter will be assigned an identification number for authentication during the voting process.
2. On election day, a voter authenticates to the system and receives a prompt to vote (a unique ballot).
3. After the voter has chosen the candidate, the voter will send the ballot to a server. The server will validate the ballot and acknowledge that the ballot was received.
4. The final stage of the voting protocol is the vote count and the audit which takes place after the count to review the election process and ensure that the integrity of the election has not been compromised.

**Your job is to produce design and prototype a system that meets the following security goals:**

* *Privacy*. Voters privacy should be maintained: no one should know how a particular person voted. The tallying computation should not reveal any information about the individual votes, but it should allow one to prove that it tallied the votes accurately. You may assume that votes are cast in a private environment and the voter's interaction with the system is not observed by anyone else.
* *Eligibility*. Only registered and authorized voters should be able to vote. No voter should be able to cast two or more (possibly contradictory) ballots. The repeated votes must be ruled, either by prevention or by detection.
* *Verifiability*. Everyone involved in the voting process should be able to verify the results. This brings transparency and trust in the election. Voters should be able to verify that their votes were received and tallied.
* *Immutability*. No one should be able to change the vote of any voter. All the records should be immutable.

The description of the system is deliberately underspecified so that you have the intellectual freedom to consider various possibilities for how such a system should operate.

Some useful references are listed on the last page. You may use web sites and other aids to help you with this project, be sure to list your references in your report.

1. **Deliverables**

Based on information you gather and review, you are to report the design and prototype of a secure electronic voting system.

You must provide a written description of the design of your system. The report should include the following:

* **System architecture**. Describe the system components (e.g., polling server, audit server, etc.), their functionality, and communication patterns.
* **Cryptographic components**: discuss appropriate choice of specific cryptographic primitives to ensure the system supports the goals outlined above.
  + Describe the concrete encryption schemes and key management approaches to be used in your system.

You must provide a written description of the prototype of your system.

This project will let you explore the implementation of software for a simplified electronic voting system that allows users to cast votes and counts the number of votes for each candidate. Limit your system to just two election candidates, i.e. each voter can choose only between two mutually-exclusive votes.

Implementation of a practical voting system should be distributed and run over the network. However, we do not expect everyone interested in applied cryptography to have network programming experience. For that reason, your implementation can run on a single machine.

Prototype some system components discussed in your system design. You may reuse the code you developed for projects 1 and 2.

Required prototype implementation:

* Support for confidentiality and integrity protection of messages exchanged between components Implement routines to read and write sensitive data (e.g., vote information, receipts) to an encrypted file.
* Implement routines for users to check if their vote was counted.
* Implement routines for users to verify that the votes were tallied accurately while preserving voters’ anonymity.

You can choose one of the two options to complete the project:

1. Explore applicability of various cryptographic schemes (e.g., blinded signatures, zero knowledge proofs, etc.) to provide a more detailed design of your system. You must state why simpler techniques will not work.
2. Implement extended prototype. In particular:

* Implement routines to support user registration so that only registered voters may cast a ballot.
* Implement routines to validate that one may have more than one vote tallied. (You may solve this either by making multiple votes impossible, or by detecting multiple votes during the vote-counting process and throwing out all but one.)

You must provide a written description of the design of your programs and a screen capture of a session demonstrating that your programs work. Clearly explain how your programs can be executed, describe expected inputs and outputs.

1. **Due date and grading**

The project is due 11:59pm April 29, 2020. The semester project will account for 15% of the grade for the course. As detailed in class, there is a substantial grade penalty for late submission of a cumulative of 10% times number of days late, i.e., 1 day late loses 10%, 2 days 30%, 3 days 60%, greater than 4 days late not accepted.

You will be graded on completeness: you need to demonstrate that your proposed system meets the outlined security requirements. Clearly state your assumptions and discuss limitations of the proposed system. We will not require rigorous proofs of correctness.

The total of 100 points for the project will be allocated to four areas as follows:

1. [30 points] System architecture.
2. [20 points] Discussion of cryptographic components.
3. [40 points] Prototype implementation.
4. [10 points] Discussion of assumptions and system limitations.

Completion of the semester project is to be an independent, individual effort for each student. Communication with fellow students for this assignment, attempting to benefit from work of another student, past or present and similar behavior that defeats the intent of an assignment is unacceptable to the University. Such behavior will be treated as a violation of USC academic integrity standards, which are summarized in the on-line tutorial available at <http://www.usc.edu/libraries/about/reference/tutorials/academic_integrity/index.php>

**References**

Trustworthy Electronic Voting Using Adjusted Blockchain Technology, Basit Shahzad, Jon Crowcroft

IEEE Access, 2019.

Blockchain-Based E-Voting System, Friorik P. Hjalmarsson, Gunnlaugur K. Hreioarsson , Mohammad Hamdaqa

Gisli Hjalmtysson, IEEE 11th International Conference on Cloud Computing (CLOUD) 2018.