M2: Requirements - Classroom Voter

1. Personnel.

Harris McCullers - harrismccullers@gmail.com

Ishaan Gandhi - ishaangandhi@gmail.com

Douglas Webster - dwebster@students.pitzer.edu

Jay Rodolitz - jbra2017@pomona.edu

Goals for this week:

- Professor program should be able to announce a poll
- Server should be able to listen to announcements.
- Any students connecting to the server should be able to see the poll, and send a response back
- At the end of some time period, the server sends the aggregation back to the professor

2. Documentation

Documentation for the project codebase is available here:
https://harrismcc.github.io/classroom-voter/

3. System Purpose

Classroom Voter is a secure LAN based polling system, similar to iClicker, which allows teachers to administer live polls. To start a poll, the teacher simply has to run the host program on their local machine. To join, students run the client program and enter the teachers IP address. Students can respond to questions via clients on their machines.

Traffic is sent on the local network. The system provides instructors with the ability to take polls while providing controllable levels of anonymity for students. For example, a teacher might make one question anonymized so that their view of the poll results just shows the number of students who voted for each option. Another poll might be transparent, and the instructor can see which participant voted for which option. The

results of the poll are persisted on the server so the professor can access them as needed.

4. System Backlog

Completed Tasks

User Type	Assets	Importance	User Story
professor	Poll creation	M	As a professor, I can create a new poll specifying the question and answer
professor	Poll sharing	M	As a professor, I can share a poll I've created with my students
student	Poll access	M	As a student, I can enter in the professor's IP and connect to the running poll
student	Poll read	M	As a student, I can see the poll that the teacher has just shared
student	Poll answering	M	As a student, I can submit my answer to a poll
professor	Poll aggregation	M	As a professor, I can see the results of the poll

Todos

User Type	Assets	Importance	User Story
professor	Create Multiple Choice polls	M	As a professor, I can present students with multiple choice polls, specifying the possible answers
professor	Poll distribution	M	As a professor I can share polls that are only accessible by a certain "class" of mine
professor	Create classes	M	The professor can designate a set of students as a "class" of users who can receive polls
student	Poll check	S	As a student, I receive confirmation that my answer has been submitted to a poll

professor	Anonymous Polls	W	As a professor, I can create polls of varying levels of anonymity, so that I can control when I know who gives what answer
student	Edit answer	S	As a student, you can update your poll answer as long as polling has not been closed by the professor.
professor	View past polls	C	The professor is able to access past polls from the server and see student answers
professor	Set correct answers	W	The professor is able to set a correct answer for a poll, and the program will tabulate the students' answers
everyone	Poll Access	M	Log in (connect) and log out (disconnect) from server
professor	Share class	W	The professor can designate other users as professors and allow them to submit polls for a given class
System administrator	Logs	W	The system administrator can view logs saying which IP address tried to authenticate as which user, and whether they were successful
Students		S	Should be able to audit that their vote was accounted for in the professor's aggregation

5. Threat Model.

- Attackers are probably students in the class (or other sections of the class), who are trying to cheat on quizzes/seem more in attendance than they are. Some attackers (if implemented in a Systems class) may be attempting to use their new hacking skills on their fellow students
- Attackers can listen to all traffic over the network, and send packets to any device on the network
- Attackers seek to find out the result of the poll, or any vote cast in a poll
- Attackers seek to falsify the results of the poll, so teachers see the wrong results
- Attackers seek to prevent students from viewing poll questions they were intended to see
- Attackers may seek access to the machines of other users

- Non-threat: Attackers from outside the local network have access to the local network
- Non-threat: Attackers do not have the ability to drop packets in transit

6. Security Goals

The system shall:

- Confidentiality: Prevent attackers from being able to intercept responses
- Confidentiality: Prevent attackers from finding out the result of the poll
- Confidentiality: Prevent attackers from finding out the result of any individual vote of poll
- Confidentiality: Prevent attackers from finding out whether any individual voted in a specific poll
- Integrity: Prevent attackers from modifying the results of the poll, or any vote submitted
- Integrity: Ensure that polls created by the professor is the exact poll that students receive (i.e., the question and/or multiple choice answers are transmitted faithfully)
- Availability: Ensure that answers submitted to polls are received, and both students and professors are aware of that.
- Availability: Ensure any poll created by a professor is received by all students
- Availability: Ensure that all student responses are received and recorded

7. Essential Security Elements

Authentication: Students need to be able to verify they are who they claim to be, otherwise students could submit answers as/for each other. This would clearly be a problem.
Authorization: Professors need to be able to choose which set of students get a poll. If all students received all poll questions, this would be inappropriate for systems where the poll is sensitive in nature, or where only the results of *some* students are important. Additionally, the only individual able to create a poll and view poll results is the professor.

Audit: Students need to be able to verify that the professor recorded their answer as intended. This is an auditing of the poll results, distributed among the students.
Confidentiality: Without confidentiality, other students could potentially view what answers are being submitted. If the poll were to be used as a quiz, this would enable students to effectively cheat on the quiz.
Integrity: Without integrity, attackers would be able to modify the results of the poll, which would defeat the purpose of Classroom Voter, which is to let professors understand what students voted for in polls.