CSEP 590E: Practical Aspects of Modern Cryptography

Homework #7 Due: 6:30pm Nov 29, 2016

1. In even the best of elections, privacy is not absolute. If I vote in an election, and the result is unanimous, everyone knows how I voted. If the result in a local election is 99 votes for candidate A and 1 vote for candidate B, and I cast my vote for candidate B, then I (and I alone) know exactly how everyone voted.
   Suppose Alice, Bob, and Carol are the only voters in a very small election in which voters each prepare and submit ElGamal encryptions of their own votes and that each encryption is made public as soon as it is submitted. Suppose that Bob doesn't care about who wins the election, but really, *really* wants to know how Alice voted. Describe how he can (surreptitiously) accomplish this in the scenario described.

2. Suppose that I run a company in a small town in which I am running for mayor. In addition to the mayoral race, there are lots of "minor" contests on the same ballot. (Let's say there are 10 voter initiatives on which voters can vote yes/no.) The voting is done is person using ballot-marking devices which produce neat paper ballots with uniform marks. There are no write-ins allowed in this election. At the conclusion of voting, the paper ballots are thoroughly anonymized and then displayed publicly so that anyone can scrutinize and count them.
   My company has 1,000 employees (each has a unique employee number in the range 1 through 1,000). I want them all to vote for me as mayor, and they all want to keep their jobs.
   Describe how I can coerce all of my employees into voting for me.

3. The coercion attack of the previous problem works just as well if voting is done digitally with an end-to-end verifiable voting system using a MixNet to dissociate ballots from voters and then publicly disclosing the ballots. Describe how to modify the MixNet system to protect the privacy of voters and even enable write-ins. (Do not simply switch to homomorphic tallying. This makes write-ins very difficult.)

4. *Group ElGamal* allows multiple individuals to each form a public-private key pair and any individual to send a single encrypted message to a group which can only be decrypted if all group members cooperate.
   An individual's private key is a_i and the corresponding public key is $A_i = g^{a_i} \bmod p$ for some publicly agreed upon parameters p and g. To encrypt a message for a group, one uses ordinary ElGamal encryption with the group public key $A = \prod_i A_i$ where i ranges over all of the group members. To decrypt a message (r, s), each group member performs a partial decryption by computing and sharing $S_i = s^{a_i} \bmod p$. The decrypted message is simply $\frac{r}{\prod_i S_i}$. (Note that it is not necessary for the group members to disclose or share their individual keys to decrypt the message for the group.)
   Assuming that we use the familiar parameters $p = 11$ and $g = 2$ and that Alice, Bob, and Carol have private keys of 3, 4, and 9 respectively, what are their individual public keys and what is the group public key? Suppose that the group receives the encrypted message (1,7). What are the partial decryptions computed by Alice, Bob, and Carol? What is the full decryption of the message (1,7)? (As before, please note that since we are working with the very small toy modulus of $p = 11$, there will unavoidably be values that repeat.)

5. In the voter-initiated "cast or spoil" audit process, a malicious device might incorrectly encrypt some votes in hopes of not getting caught. The device's cheating will be detected if a voter ever challenges (spoils) an incorrect encryption. If a device incorrectly encrypts 1 out of every n votes (on average), the probability that this goes undetected depends on the number of challenges. If there are m challenges over the course of an election (issued at random), the probability of the malicious device going undetected is approximately $e^{-\frac{m}{n}}$, where e is the base of the natural logarithm, $e \approx 2.71828$.
   Assuming that in a large national election with 100,000,000 votes, a malicious entity wants to change the outcome by altering 1% of the votes, what percentage of voters would have to issue challenges to detect this cheating with probability at least 50%? With probability at least 99%?