COS 433 — Cryptography — Homework 8.

Boaz Barak

Total of 130 points. Due November 15th, 2007.

Exercise 1 (Interaction is necessary, 15 points)**.** Let L be a language that is not decidable in polynomial time (that is, there is no efficient (possibly probabilistic) algorithm that on input x outputs 1 if $x \in L$ and 0 otherwise). Show that there is no *non-interactive* zero knowledge proof system for L. That is, show that if a language L has a proof system that consists of a single message from the prover to the verifier then L is decidable by a polynomial-time algorithm.

Exercise 2 (Randomness is necessary, 15 points)**.** Let L be a language that is not decidable in polynomial-time. Show that there is no *deterministic* zero knowledge proof system for L. That is, show that if a language L has a proof system where the verifier is deterministic then L is decidable by a polynomial-sized algorithm.

Exercise 3 (35 points)**.** Consider the following more sophisticated identification problem. We wish to restrict access of people to building in the university, but different people (students, faculty, maintenance workers) may be authorized to enter the building at *different times*. We want to design a "box" that would be on the door and protect the access to the building. In addition we give every user a smart card that contains some secret information and can interact with the box. However, we assume that each user can open up their own smart card and see all the data there, and in particular people attacking the system have access to one more more such smart cards.

Design and prove security for an identification protocol with the following properties: (If you design a correct protocol but can't or don't have time to prove everything, you will still get partial credit)

- Assume that each person is authorized for a particular range of hours every day. That is, the range of each person is two numbers $t_1 \leq t_2$ between 0 and 24. For simplicity assume that every one has access to perfectly synchronized clocks.
- There is some central trusted algorithm that provides each person with some secret data corresponding to her authorized range. She can use this secret data when interacting with the box. This algorithm also provides the public information for the box.
- The box contains only a clock and some public information. The box does *not* contain the list of authorized people and their authorized time periods. We assume that it may be possible for an attacker to "open up" the box and see all of its contents.
- Even if an attacker constructs his own "fake box" that interacts with a person authorized to enter at a particular time, it should not learn anything about this person's secret information. It should not even learn anything about the range of that person (other than that this range contains that particular time). In fact, can you ensure that it doesn't even learn the person's name?

- A person that has authorization for a particular time period can not (with non-negligible probability) convince an honest box to allow her to enter in another time period. This holds even if that person is given the contents of data in the box and even in previous days managed to install a cheating box that interacted with other people that are authorized for that period.

See footnote for hint[1]

Clarification: You should try to design your system to achieve all the above security goals. Moreover, for whatever goal you believe your protocol achieves, you should argue informally why this is the case. However, to keep things simple when *proving security*, you only need to provide a full formal proof of security for the following simplified attack scenario: we have Alice— a user that is authorized to enter the building at all hours (e.g., range 0 to 23) and Bob— a user that is authorized to enter the building only at specified times (say between 1 and 2). We think of the following attack: Bob gets all the information contents of the valid box and his own smartcard. Then he has Alice interact once with a fake box of its own design. Then, he needs to interact with the real box and convince it to let him in at a time that is not in his range. We say the system is secure of the probability of Bob's success is at most $1/n^{\omega(1)}$.

For 15 points bonus: How would you change the protocol to protect against students giving their friends their smart cards? Assume that students have some secret information (e.g., social security number, middle name) that they'd not want to share with their friends.

Exercise 4 (35 points)**.** The notion of forward security has a natural generalization to *signature schemes.*

1. State informally the goals for a forward-secure digital signature scheme, and provide a formal definition for this notion.

2. Suppose that $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ is a construction for standard (not forward-secure) chosen-message attack secure signature schemes, where the length of the private and public keys, and also the length of the signatures is n (and the length of messages to be signed can be arbitrary, i.e., works for any message $\{0,1\}^*$). Construct a forward-secure signature scheme with lengths of private and public keys is $O(n)$, and the length of each signature is at most $O(Tn)$, where T is the maximum number of time periods allows. (If it makes things simpler for you, the signature scheme you construct can be defined only for messages of length n.)

3. Under the same assumption, construct a signature scheme where the length of the public key and signatures is $O(n)$, but the length of the private key can be $O(T \cdot n)$.

4. For **15 points bonus**, can you construct a forward secure signature scheme with better parameters? (Ideally, we'd like private key, public key, and signature length to be at most $O(n \log T)$ or perhaps even $O(n + \log T)$).

[1] **Hint:** For starters you might want to think how you would do that if you could put secret information in the box, but did not want to store there the long list of all people and their time ranges. Use the following components: zero-knowledge proofs for NP, commitment schemes, message authentication codes. Note that if you have some secret information, you can make a commitment to that secret public.