Introduction to Cryptography: Homework 10

December 1, 2021

Requirements about the delivery of this assignment:

- Submit a pdf-document via Brightspace;
- Make sure that you write both name and student number on all documents (not only in the file name).

Deadline: Monday, December 13, 17:00 sharp!

Grading: You can score a total of 100 points for the hand-in assignments. To get full points, please **explain all answers clearly**.

Exercises:

1. **Soundness of the Schnorr authentication protocol.** In the lecture, we have seen that the Schnorr authentication protocol is sound, because the best that any cheating Eve can do is either solve the discrete log problem, or guess the challenge, and generate a valid transcript based on her guess. In fact, we can even prove that this is the best that she can do. That is, for any commitment, there is at most one challenge for which she can generate a valid transcript. We can prove this by showing that, if any prover can generate a valid transcript for two or more challenges, then she must know the secret. To prove this, we consider two valid transcripts $(V; c; r)$ and $(V; c'; r')$ on the same commitment V, with $c \neq c'$. Show how you can recover secret a from this.

2. **The Fiat-Shamir transform.** The Fiat-Shamir transform is not only useful because it can make an interactive protocol non-interactive. It is also useful, because it automatically introduces an honest verifier. Recall that, in the lecture, we mentioned that in honest-verifier zero-knowledge protocols, we require that the challenges are generated randomly.

   (a) Explain why the Fiat-Shamir transform introduces an honest verifier, when the hash function behaves like a random oracle.

   (b) Explain why the problem in Hand-in Exercise 1(e) of this week may not be much of a problem if the Fiat-Shamir transform is used, i.e., $c \xleftarrow{\$} \mathbb{Z}/q\mathbb{Z}$ is replaced by $c \leftarrow \mathrm{h}(p; g; A; V)$.

3. **Forgery of Schnorr signatures.** In this exercise, we consider the security of Schnorr signatures against forgery.

   (a) Explain, in steps, how Eve, who does not know private key a, can try to create a forgery on some specific message m for Alice's private key. [Hint: Consider the soundness of Schnorr's interactive authentication protocol, i.e., which considers Eve as cheating prover (see slide 17 of slides_10_Schnorr).]

   (b) Give the probability that Eve succeeds in one try, using your approach in (a).

   We assume that p is a prime number of 3072 bits and q is a prime number of 256 bits, and that the workload N needed to compute one Schnorr signature is 1.

   (c) Give the security strength of Schnorr's signature scheme with respect to the forgery of signatures on some specific message m, by only considering the attack in (a).

   (d) Give the security strength of Schnorr's signature scheme if you take into account that you can also find Alice's private key a to forge signatures. [Hint: Check slide 25 of slides_9_DH.]

   (e) Suppose we select the challenge from a smaller set $\{0, 1\}^n$ (with $n \in \mathbb{N}$), i.e., the hash digests are n bits long. Give the security strength against your attack in (a).

   (f) Give the smallest n for which Schnorr's signature schemehas at least the same security strength as your answer in (d).

Hand-in assignments:

1. **(50 points) Another Σ-protocol.** In this exercise, we consider the difference between zero-knowledgeness and honest-verifier zero-knowledgeness. In particular, we show that protocols such as that in Table 1 exist that are complete, sound and honest-verifier zero-knowledge (like Schnorr's protocol), but are decidedly not secure against a cheating verifier, i.e., a verifier that does not necessarily choose the challenge c uniformly at random (and therefore honestly), but rather chooses it cleverly.

Alice		Bob
p, g, q, A, a		p, g, q (Alice: A)
$v \overset{\$}{\leftarrow} \mathbb{Z}/q\mathbb{Z}$		
$V \leftarrow g^v$	$\xrightarrow{\text{Alice}, V}$	
	$\xleftarrow{c}$	$c \overset{\$}{\leftarrow} \mathbb{Z}/q\mathbb{Z}$
$r \leftarrow cv - a$	$\xrightarrow{r}$	$V^c \overset{?}{=} g^r A$

Table 1: Variation on Schnorr's protocol

(a) What is the (main) difference between this and Schnorr's protocol? 7 pt

(b) Show that the protocol is complete. 7 pt

(c) Show that the protocol is special sound, i.e., given two transcripts $(V; c; r)$ and $(V; c'; r')$ on the same commitment such that $c \neq c'$, we can extract the secret a. 9 pt

The simulator generates transcripts $(V; c; r)$ for some challenge c by choosing a response $r \overset{\$}{\leftarrow} \mathbb{Z}/q\mathbb{Z}$ and computing the commitment as $V \leftarrow (g^r A)^{c^{-1}}$.

(d) Show that the protocol is honest-verifier zero-knowledge, i.e., for some given challenge c, show that both the protocol and the simulator generate a specific transcript $(V; c; r)$ with probability $1/q$. 20 pt

(e) Show how a cheating verifier Bob can recover Alice's private key a after one correct run of the protocol. 7 pt

2. **(50 points) Schnorr signature misuse.** Consider Schnorr signatures in a setting with domain parameters (p, g, q) and Alice's public key $A = g^a$.

(a) Alice sends you a message m with signature $(r, 1)$. Explain how you can determine a from this information. 8 pt

(b) Alice now sends you a message m with signature (r, g). Explain how you can determine a from this information. 8 pt

(c) Alice now sends you two messages $m \neq m'$ with signatures (r, V) and (r', V). Explain how you can determine a from this information. 18 pt

(d) What did Alice do wrong in her generation of the signatures in (c)? 8 pt

(e) Explain why the aforementioned issues are not likely to occur if Alice generates her commitments as specified in the protocol. 8 pt