Cryptography and Security — Final Exam

Serge Vaudenay

22.1.2025

- duration: 3h00
- no documents allowed, except one 2-sided sheet of handwritten notes
- a pocket calculator is allowed
- communication devices are not allowed
- the exam invigilators will **not** answer any technical question during the exam
- readability and style of writing will be part of the grade
- answers should not be written with a pencil

1 Making Meaningful Collisions

We consider a hash function H following the Merkle-Damgård construction, to hash a file which is encoded like a programming language such as PDF or Postscript. We denote by $C(X, Y)$ the compression of a chaining value X with a message block Y to obtain the next chaining value.

Q.1 Describe an algorithm $\mathcal{A}(X) \to Y, Y'$ taking an initial chaining value X as input and producing two different message blocks Y and Y' such that $C(X, Y) = C(X, Y')$. Give its complexity in terms of number of compressions.
NOTE: full points only go to algorithms which are at least as good as the one in the solution.

Q.2 Give an algorithm using $\mathcal{A}$ as a subroutine and making a collision on H. The complexity of this algorithm must be equal to the one of $\mathcal{A}$ plus a constant overhead.

Q.3 By using a method seen in class, give an algorithm which takes two files F and F' as input and produce two files D and D' such that a document viewer would show D and F the same, it would show D' and F' the same, and we would have $H(D) = H(D')$.

Q.4 If now H follows the sponge construction, by defining a proper C function, show that the same method works.

Q.5 What is the problem with the complexity of the previous method?

2 Secure KEM

We recall how that a KEM scheme is defined by three efficient algorithms

- $\mathsf{Gen} \to (\mathsf{pk}, \mathsf{sk})$
- $\mathsf{Enc}(\mathsf{pk}) \to (k, \mathsf{ct})$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to k'$

Q.1 Define correctness for a KEM.

Q.2 Adapt the IND-CPA notion for public-key cryptosystems to define the appropriate game for KEM.

Q.3 Define IND security using the previous game.

3 Quantum Billionaires

In the bitcoin infrastructure, a miner who succeeds to make a new block in the blockchain inserts a special transaction (called the coinbase transaction) at position zero. The input of the transaction consists of the collected fees from other transactions and some newly minted bitcoins. The output is typically the public key of the miner. There is no signature to validate the coinbase transaction.

For other transactions, there is normally an ECDSA signature. We use the standard elliptic curve P256 having a group generated by some point G of prime order n. Coordinates are modulo p. We have $p + 1 - 2\sqrt{p} \leq n \leq p + 1 + 2\sqrt{p}$. The transaction itself specifies some UTXO with the same public key Q and several outputs, where Q is in the group. If the transaction hashes onto a number z, a valid signature is a pair (r, s) such that $r = R_x \bmod n$, where R_x is the x-coordinate of $R = \frac{z}{s}G + \frac{r}{s}Q$.

Satoshi Nakamoto is believed to have created many address of miners which never spent the collected bitcoins in the early days of the blockchain. Overall, this sums up to one million bitcoins. (Note that the recent record rate was about 100 000USD per bitcoin.)

Q.1 The signature algorithm starts by computing $R = k \cdot G$ with a random k then it takes $r = R_x \bmod n$ which is part of the signature. Given r, how many possible points R on the curve would give $r = R_x \bmod n$.

Q.2 Recall how the signature algorithm works to sign z by using the signing key d and what is the relation between d and Q.

Q.3 Imagine you have a quantum computer with enough qubits of memory. How would you collect the unspent bitcoins for yourself?

Q.4 Instead of paying to a public key, we can pay to a public key hash without exposing the public key. How can a miner protect their revenue against quantum computer?

Q.5 If a user signs any transaction, as soon as it appears on the blockchain, show that they can loose their remaining bitcoins in a similar attack even though they collected them by the pay-to-public-key-hash option.