TECHNISCHE UNIVERSITEIT EINDHOVEN
Faculty of Mathematics and Computer Science
Introduction to Cryptology, Monday 7 April 2025

Name :

TU/e student number :

Exercise	1	2	3	4	5	6	total
points							

Notes: Please hand in this sheet at the end of the exam. You may keep the sheet with the exercises.
This exam consists of 6 exercises. You have from 18:00 – 21:00 to solve them. You can reach 100 points.
Make sure to justify your answers in detail and to give clear arguments. Document all steps and intermediate results, in particular of algorithms, on the exam paper, do not use the scrap paper. It is not sufficient to state the correct result without the explanation and the steps that lead to it.
If the problem statement asks for usage of a particular algorithm other solutions will not be accepted even if they give the correct result.
All answers must be submitted on TU/e letterhead; should you require more sheets ask the proctor. State your name on every sheet.
Do not write in red or with a pencil.
You are not allowed to use any books, notes, or other material.
You are allowed to use a simple, non-programmable calculator without networking abilities. Usage of laptops and cell phones is forbidden.

1. This exercise is about LFSRs. Do the following subexercises for the sequence
$$s_{i+6} = s_{i+5} + s_{i+3} + s_i.$$

   (a) Draw the LFSR corresponding this sequence. $\boxed{\text{3 points}}$

   (b) State the characteristic polynomial f and compute its factorization. You do not need to do a Rabin irreducibility test but you do need to argue why a factor is irreducible.
   Reminder: Factors may appear with multiplicity larger than one. $\boxed{\text{10 points}}$

   (c) Write the factorization of f from (b) in the form $f = \prod f_i^{e_i}$ with integers $e_i > 0$ and f_i different irreducible polynomials, i.e., group equal factors.
   For each of the $f_i^{e_i}$ compute the order.
   Reminder: We have shown in class that $f_i^{e_i}$ has order different from that of f_i (for $e_i > 1$) and you need to compute the order directly. We do not have a theorem for this. $\boxed{\text{7 points}}$

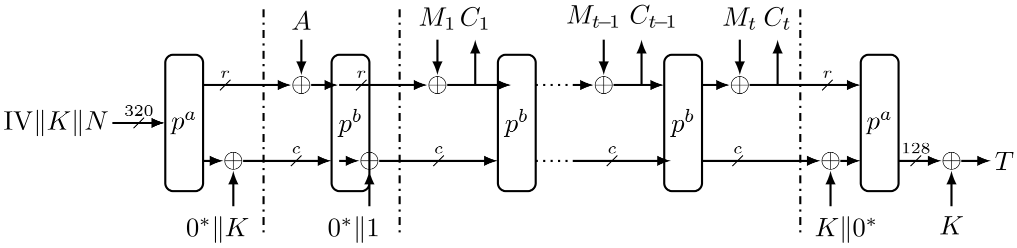
   (d) What is the longest period generated by this LFSR?
   Make sure to justify your answer. $\boxed{\text{3 points}}$

   (e) State the lengths of all subsequences so that each state of 6 bits appears exactly once.
   Make sure to justify your answer and to check that all 2^6 states are covered. $\boxed{\text{13 points}}$

2. ASCON-128 is a cipher for authenticated encryption which permits to authenticate additional data which is not encrypted but only authenticated. ASCON-128 is designed to be light weight and deliver 128 bits of security. ASCON-128 requires the message and additional data each to be split into blocks of length 64 bits and internally uses a non-linear mixing function p which takes as input 320 bits and produces 320 bits of output. Assume for simplicity that both the additional data and the message have lengths that are multiples of 64. Let the additional data A be one block of 64 bits and let the message M be split into 64-bit blocks $M_1, \dots, M_t$. Let K denote the 128-bit key shared by Alice and Bob and IV be the fixed 64-bit initialization vector `80400c0600000000` in hexadecimal. (Each member of the ASCON family, such as ASCON-128a, has a different IV which is fixed for all messages). The sender picks a fresh 128-bit nonce N per message, so that the concatenation IV$||K||N$ has 320 bits. Then $\mathsf{ASCON\text{-}128enc}(K, IV, N, A, M) = (C, T)$, where ciphertext C consists of t blocks $C_1, \dots, C_t$ of 64 bits each and a 128-bit

authentication tag T so that $\mathsf{ASCON\text{-}128dec}(K, IV, N, A, C, T) = M$ and that the decryption function outputs "failure" if the tag T is not correct.

Here is a schematic description of ASCON-128.



Initialization Associated Data Plaintext Finalization

Image credit: adapted from ASCON team.

In this schematic description, the top horizontal arrows carry $r = 64$ bits and the bottom arrows carry $c = 256$ bits. The exponent of p states the number of times the non-linear mixing function p is run, which is $a = 12$ in initialization and finalization and $b = 6$ while handling A and M. The dots indicate that the intermediate blocks are handled the same way as M_1 and M_{t-1}. The exponent $*$ on 0 means as many zeros as are needed to reach c bits.

Hint for all subexercises: To structure your answer, it helps to give names to the upper and lower parts, e.g., H_i and L_i. Make sure to describe how to compute them when you use them.

(a) Describe how ASCON-128 encryption of long messages works by writing C_0 and a general C_i in terms of K, IV, N, A, M_1, M_i, and other M_j and C_j as necessary. $\boxed{\text{4 points}}$

(b) Describe how decryption of long messages and verification of the authentication tag work by writing M_1 and a general M_i in terms of K, IV, N, A, C_1, C_i, and (if necessary) other M_j and C_j and describing how the receiver can check the authentication tag T. $\boxed{\text{5 points}}$

(c) Assume that ciphertext C_j gets modified in transit. Show which message blocks get decrypted incorrectly and explain why others get decrypted correctly. Show how the authentication tag T catches this error. $\boxed{\text{5 points}}$

(d) Assume that the additional data A gets modified in transit. Show which message blocks get decrypted incorrectly and explain why others get decrypted correctly. Show how the authentication tag T catches this error. $\boxed{\text{4 points}}$

3. This problem is about RSA encryption. Let $p = 311$ and $q = 419$. Compute the public key using $e = 65537$ and the corresponding private key.
   Reminder: The private exponent d is a positive number. $\boxed{\text{8 points}}$

4. This problem is about the DH key exchange. The public parameters are the group G and generator g, where $G = (\mathbb{F}_{1033}^*, \cdot)$ and $g = 5$. Alice's public key is $h_A = 875$. Bob's private key is $b = 22$, Compute the DH key that Bob shares with Alice. $\boxed{\text{8 points}}$

5. The integer $p = 29$ is prime. You are the eavesdropper and know that Alice and Bob use the Diffie-Hellman key-exchange in $\mathbb{F}_{29}^*$ with generator $g = 2$. Alice's public key is $h_A = g^a = 20$. Use the Baby-Step Giant-Step method to compute Alice's private key a.
   Verify your result, i.e. compute g^a. Make sure to detail the steps you take in verifying the result. $\boxed{\text{11 points}}$

6. Winternitz signatures are based on cryptographic hash functions and nothing else. This simplicity makes them interesting as having minimal requirements, but they also have a downside, namely they are one-time signatures. The latter means that a public key can be used only once.

   There are further constructions that turn Winternitz one-time signatures into few-time signatures but these are beyond the scope of this exercise.

   The following describes how to sign messages of 256 bits.

   Let h be a cryptographic hash function having an output length of 256 bits.

   The private key s consists of 34 randomly picked strings $s_0, s_1, \ldots, s_{33}$ of 256 bits each.

   The public key p consists of 34 strings $p_0, p_1, \ldots, p_{33}$ of 256 bits each which are computed as $p_i = h^{255}(s_i)$, i.e., the 255-fold application of h to s_i.

Let message M hash to the 256-bit message hash m. To sign M, first write m in base 2^8 as $m = \sum_{i=0}^{31} m_i 2^{8i}$, then compute

$$c = \sum_{i=0}^{31} (2^8 - m_i)$$

and write c in base 2^8 as $c = c_0 + c_1 2^8$. Finally compute the signature $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{33})$ where $\sigma_i = h^{m_i}(s_i)$ for $0 \leq i < 32$ and $\sigma_{j+32} = h^{c_j}(s_{j+32})$ for $0 \leq j < 2$.

(a) Explain how the verifier can check that σ is correct, given the message and the public key.
Make sure to comment on why $c_1 < 2^8$. 5 points

(b) As an intermediate step to an attack, assume that only the first 32 parts of the signature would be published, i.e., that c_0 and c_1 as well as σ_{32} and σ_{33} would not be computed. In this situation, show how an attacker can produce a valid signature on some other message.

For concreteness assume that $m_0 = 3, m_1 = 7, m_3 = 14$ and $m_4 = 9$ and explain for which values of m'_0, m'_1, m'_2 and m'_4 the attacker can sign. This part ignores the other 28 parts of the message as they would be handled accordingly. 6 points

(c) Explain how including c_0 and c_1 stops the attack you found in b). 4 points

(d) Explain why Winternitz signatures are one-time signatures, i.e., show how an attacker could forge a third signature given signatures σ and σ' on M and M'.

For concreteness assume that $m_0 = 3, m'_0 = 127$, and that $m_i = m'_i$ for $1 \leq i < 32$ and show for which values of m''_0 you can sign and how. 4 points