TECHNISCHE UNIVERSITEIT EINDHOVEN
Faculty of Mathematics and Computer Science
Exam Cryptology, Friday 25 January 2019

Name :

TU/e student number :

Exercise	1	2	3	4	5	total
points						

Notes: Please hand in *all sheets* at the end of the exam. This exam consists of five exercises. You have from 13:30 – 16:30 to solve them. You can reach 100 points.
Make sure to justify your answers in detail and to give clear arguments. Document all your steps. In particular for algorithms do include all relevant parts of the internal state of the algorithm (at least the information given in the examples that were done during the lecture); it is not sufficient to state the correct result without the explanation.
If the exercise asks for the 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. Text written with one of the two will be ignored.
You are allowed to use any books and notes, e.g. your homework. You are not allowed to use the textbooks of your colleagues.
You are allowed to use a calculator without networking abilities.
Usage of cell phones or laptops (other than the ones brought be the TAs) is forbidden.

1. **Warm-up: Diffie-Hellman (3 points total).** This exercise is about the Diffie-Hellman key exchange. The system uses the multiplicative group $\mathbb{F}_p^*$ modulo the prime $p = 24097$. The element $g = 5 \in \mathbb{F}_{24097}^*$ has order 24096 and is thus a generator of the full multiplicative group.

   (a) Alice chooses $a = 192$ as her secret key. Compute Alice's public key. 1 point

   (b) Alice receives $h_b = g^b = 8964$ from Bob as his Diffie-Hellman keyshare.

   Compute the key shared between Alice and Bob, using Alice's secret key a from the first part of this exercise.

   2 points

2. **Discrete logarithms (27 points total).** This exercise is about computing discrete logarithms in the multiplicative group of $\mathbb{F}_p$ for $p = 24097$. The element $g = 5$ has order $q = 24096$. The factorization of q is $q = 24096 = 2^5 \cdot 3 \cdot 251$. Use the Pohlig-Hellman attack to compute the discrete logarithm b of Bob's key $h_b = g^b = 8964$. I.e., you should break down the computation into discrete logarithms in subgroups of the smallest order possible. So you will have to eventually compute discrete logarithms in subgroups of order 2, 3 and 251. To solve the discrete logarithm in the subgroup of order 251 use the baby-step, giant-step algorithm. Verify your intermediate and your final solutions!

   27 points

3. **RSA Encryption (30 points total).** This exercise is about RSA encryption.

   (a) Generate an RSA key pair. For this, you are given three digit random numbers $x_1, ..., x_4$ below for which you have to check primality until you found two prime numbers. (Hint: Use the x_i according to the order of the indices.) For primality testing use the Miller-Rabin primality test. As a-values in Miller-Rabin use the $a_1, ..., a_6$ given below (do only use each a_i once, i.e., if you already used a_1 in the primality test for x_1 do not use it in the primality test for x_2 and so on). For this exercise it is sufficient to achieve a probability of 0.875 that a number is prime.

i	1	2	3	4
x_i	671	673	677	679

i	1	2	3	4	5	6	7
a_i	61	151	13	481	101	17	222

   i. How many iterations of Miller-Rabin are necessary to achieve a probability of 0.875 that a number is prime?

   1 point

   ii. Output p, q, as well as public key (N, e) and secret key (N, d) using public exponent $e = 2^{16} + 1$.

   9 points

   (b) Bob uses public key $(N, e) = (128\,417, 17)$. Encrypt the message 77 for Bob using schoolbook RSA in combination with 'square & multiply'.

   3 points

   (c) Bob just received some interesting message but you only saw the ciphertext: $c = 114\,472$. So, lets break Bob's key. Factor Bob's $N = 128\,417$ using Pollard's roh method with iteration function $x_{i+1} = x_i^2 + 1$ and Floyd's cycle finding algorithm. Start with $x_0 = 7$. Do not do the gcd over 100 pairs, but do it for each pair x_i, x_{2i}.

   9 points

   (d) Because this was so much fun, factor Bob's N once more, this time using Pollard's $(p - 1)$ method. Use as exponent $s = \text{lcm}\{1, 2, 3, 4, 5, 6, 7\}$ and as basis $a = 17$. 5 points

   (e) Now decrypt $c = 114\,472$ using Bob's secret key. Provide Bob's secret key (N, d), and the decrypted message. 3 points

4. **Reductionist proofs (20 points total).** This exercise is about reductionist proofs.

   Let $\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a digital signature scheme that uses a secret key $\mathrm{sk} \in \{0,1\}^{\ell n}$ consisting of ℓn random bits where n is the security parameter. Furthermore assume that Sig has the special property that there exists a key derivation algorithm DKey which given a secret key $\mathrm{sk} \in \{0,1\}^{\ell n}$ outputs a matching public key pk for Sig.

   We can construct a signature scheme Sig' from Sig that works essentially as Sig with the only difference that it has secret keys $\mathrm{sk}' \in \{0,1\}^n$ consisting of just n random bits (instead of ℓn bits). For this we make use of a pseudorandom generator $\mathsf{G} : \{0,1\}^n \to \{0,1\}^{\ell n}$ with expansion factor ℓ.

   Key generation: The key generation algorithm Gen of Sig gets replaced by a new key generation algorithm Gen'. The new key generation Gen' first samples a random n bit secret key $\mathrm{sk}' \leftarrow_R \{0,1\}^n$, then it runs $\mathrm{pk} \leftarrow \mathsf{DKey}(\mathsf{G}(\mathrm{sk}'))$ to derive a public key by first expanding sk' to ℓn bits and then applying DKey.

   (a) Write down the signature generation and the signature verification algorithms Sign', and Vrfy' of Sig'.

   5 points

   (b) Now you have to show that the new scheme is secure. You are asked to show that if Sig is EU-CMA-secure and G is a secure pseudorandom generator, then Sig' is also EU-CMA-secure.

   More specifically you are asked to show that if there is a non-negligible difference in the success probability of an adversary $\mathcal{A}$ when it runs in the EU-CMA experiment against Sig or Sig' this can be used to break the pseudorandomness of G.

   Recall, the success probability of a distinguisher $\mathcal{D}$ against the pseudorandomness of pseudorandom generator G is defined as

$$\epsilon_{\mathrm{prg}}(n) = \big|\Pr[\mathcal{D}(x) = 1 \mid r \leftarrow_R \{0,1\}^n, x \leftarrow \mathsf{G}(r)] - \Pr[\mathcal{D}(x) = 1 \mid x \leftarrow_R \{0,1\}^{\ell n}]\big| .$$

   We call G a secure pseudorandom generator when $\epsilon_{\mathrm{prg}}(n)$ is a negligible function in n for all polynomial-time distinguishers $\mathcal{D}$.

   The EU-CMA security of a signature scheme is defined using the following experiment:

Experiment $\mathrm{Exp}^{\mathrm{EU-CMA}}_{\mathcal{A},\mathsf{Sig}}(n)$

i. $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathsf{Gen}(1^n)$
ii. $(m, \sigma) \leftarrow \mathcal{A}^{\mathsf{Sign}_{\mathrm{sk}}(\cdot)}(\mathrm{pk})$. Let $\{m_i\}_1^q$ denote $\mathcal{A}$'s queries to $\mathsf{Sign}_{\mathrm{sk}}$
iii. If $(\mathsf{Vrfy}_{\mathrm{pk}}(m, \sigma) := 1$, and $m \notin \{m_i\}_1^q)$ return 1
iv. Else return 0.

The success probability of an adversary $\mathcal{A}$ in winning the above game is defined as

$$\epsilon_{\mathrm{eu-cma}}(n) = \Pr[\mathrm{Exp}^{\mathrm{EU-CMA}}_{\mathcal{A},\mathsf{Sig}}(n) = 1].$$

We say that Sig is EU-CMA-secure if $\epsilon_{\mathrm{eu-cma}}(n)$ is a negligible function in n for all polynomial-time adversaries $\mathcal{A}$.

15 points

5. **Perfect secrecy (20 points total).** Let $q \in \mathbb{Z}$, $q > 1$ be prime. Consider the following secret key encryption scheme $\mathcal{E}$ over message space $\mathcal{M} = \{1, \ldots, q-1\}$:

   $\mathsf{Gen}()$: Outputs (k_1, k_2), a pair of random numbers, as secret key where $k_1 \leftarrow_R \{1, \ldots, q-1\}$ and $k_2 \leftarrow_R \{0, \ldots, q-1\}$.

   $\mathsf{Enc}_k(m)$: Returns $c = k_1 m + k_2 \mod q$.

   $\mathsf{Dec}_k(c)$: Returns $m = (c - k_2)k_1^{-1} \mod q$.

   Prove that $\mathcal{E}$ is a perfectly secret encryption scheme.

   20 points