NWI-IMC061 – Applied Cryptography
Final Exam, Academic Year 2021–2022

Remarks:

- Exams must be made **individually**, without any discussion with fellow students or other parties. You are free to **passively** use the internet; i.e., you are allowed to read and use literature online but you are not allowed to change the web (by discussing about questions anywhere). There may be follow-up oral exams where you will be asked to clarify your solutions.
- Each student will get a **personalized appendix**. To obtain your own personalized appendix, send a request e-mail to `appliedcryptography@cs.ru.nl` (redirects to Bart, Simona, Krijn) with subject "Applied Cryptography: personalized appendix". The personalized appendices have a sequence number, and will be distributed in the order in which the request e-mails arrive. You will receive your personalized appendix within 12 hours after we received your request e-mail (typically much faster). You are free to send the request e-mail before the availability of the exam (i.e., after reading the draft version of this header); in this case, however, you will still only get the personalized appendix after the availability of the exam.
- The exam will be made available in Brightspace under "Assignments", and this is also where you have to hand in your answers. The hand-in format is a LaTeX file turned into PDF. **Use the LaTeX template that is provided on Brightspace.**
- You can revise your submissions in Brightspace; **your latest submission will be graded.**
- There will be **NO** deadline extension.
- To be clear, there will absolutely be **NO** deadline extension.
- You have three weeks for this exam. As a rule of thumb, the difficulty and time effort should be roughly comparable to an ordinary on-campus open book exam **under the assumption that you are well-prepared**.
- The exam questions are described as accurate as possible, but if you are in doubt on how to interpret a question, contact `appliedcryptography@cs.ru.nl` with subject "Applied Cryptography: question about exam". Do **NOT** post your questions at the Discord server.

Availability of Exam: Wednesday, June 8, 2022

Deadline: Wednesday, June 29, 2022, 23.59 (Dutch time)

(The exam starts on the next page!)

1. **(18 points)** Read the paper on the symmetric cryptographic scheme that is assigned to you (see your personalized appendix). You do **not** need to read the proof or highly technical parts, if any.

   (a) **(2pt)** Give a 1/2 page summary of the entire paper *in your own terminology.* Include at least the problem the authors want to solve and their results with respect to that problem.

   (b) **(1pt)** What type of symmetric cryptographic scheme is introduced in this paper? (E.g., stream encryption mode, block cipher, authenticated encryption, ...) No explanation necessary.

   (c) **(1pt)** What is the size of the key to the cryptographic scheme in terms of the size parameters of the underlying primitive(s)?

   (d) **(2pt)** Does the cryptographic scheme operate on fixed-length or variable-length inputs? Concisely explain your answer.

   (e) **(1pt)** Does the cryptographic scheme generate fixed-length or variable-length outputs? Concisely explain your answer.

   (f) **(1pt)** List *all* cryptographic primitives the cryptographic scheme is based on.

   (g) **(2pt)** Did the authors deliver a security proof for their construction, and if so, under what cryptographic assumptions is the scheme proven secure?

   (h) **(2pt)** What is the practical relevance of the introduced scheme, if any? Concisely explain your answer.

   (i) **(2pt)** Name *a* cryptographic scheme that you learned of during the lectures that aims to serve the same goal (i.e., that is of the same type as what you mentioned in question (b)).

   (j) **(2pt)** Give an advantage of the proposed scheme over the scheme you mentioned in question (i). Concisely explain your answer.

   (k) **(2pt)** Name a possible disadvantage of the proposed scheme, other than what is possibly suggested by the authors themselves as "future work"? Concisely explain your answer.

2. **(16 points)** Let $k, t, n, a, b \in \mathbb{N}$, and let $\widetilde{E} : \{0,1\}^k \times \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ be a tweakable block cipher. Consider the following nonce-based authenticated encryption scheme CrAp (for Applied Cryptography), that gets as input a key K of k bits, a nonce N of t bits, and a message M of arbitrary length, and that generates a ciphertext C and a tag T:

   - The message M is first padded with a 1 and a sufficiently many number of 0s so that it is of length a multiple of $n - a$ bits. It is then partitioned into $(n-a)$-bit blocks $M_1, \ldots, M_\ell$.
   - Each block is encrypted as $C_i \leftarrow \widetilde{E}_K(N, M_i \| \langle i \rangle_a)$, where $\langle i \rangle_a$ is the encoding of i as an a-bit string.
   - The tag is computed as $T \leftarrow \mathsf{trunc}_b\big(\widetilde{E}(K, N, \langle \mathsf{len}(M) \rangle_{n-a} \| \langle 0 \rangle_a)\big)$, where $\langle \mathsf{len}(M) \rangle_{n-a}$ encodes the bit length of M as an $(n-a)$-bit string, and trunc_b truncates its input to b bits.

   Refer to your personalized appendix for the actual values k, t, n, a, b for which you have to make the exercise, and for a picture of the authenticated encryption scheme. (Here, $\widetilde{E}_K^N(\cdot)$ is a shorthand notation for $\widetilde{E}(K, N, \cdot)$.)

   (a) **(2pt)** Describe how CrAp_K^{-1} operates: what are the inputs and their sizes, the outputs and their sizes, and how are the outputs computed from the inputs?

   (b) **(2pt)** CrAp_K accepts messages M of arbitrary length, however, strictly seen there is a limit on the maximum message size due to the fact that we use a counter. What is the maximum length of M in bits?

(c) **(2pt)** What security property do we require of $\widetilde{E}$ in order to be able to prove security of CrAp as an authenticated encryption scheme? Concisely explain your answer informally.

The ae-security (as in slide 4 of "Authenticated Encryption") can be split into confidentiality and authenticity

$$\begin{aligned}\mathbf{Adv}^{\mathrm{conf}}_{\mathsf{CrAp}}(\mathcal{D}) &= \left|\mathbf{Pr}\left(\mathcal{D}^{\mathsf{CrAp}_K} = 1\right) - \mathbf{Pr}\left(\mathcal{D}^{\$} = 1\right)\right| \,,\\ \mathbf{Adv}^{\mathrm{auth}}_{\mathsf{CrAp}}(\mathcal{D}) &= \left|\mathbf{Pr}\left(\mathcal{D}^{\mathsf{CrAp}_K,\mathsf{CrAp}_K^{-1}} = 1\right) - \mathbf{Pr}\left(\mathcal{D}^{\mathsf{CrAp}_K,\perp} = 1\right)\right| \,.\end{aligned}$$

(d) **(2pt)** Derive an upper bound on $\max_{\mathcal{D}} \mathbf{Adv}^{\mathrm{conf}}_{\mathsf{CrAp}}(\mathcal{D})$, where the maximum is taken over all distinguishers that make 1 oracle query, of length exactly ℓ padded message blocks. (Hint: you can rely on the security proof of $\mathsf{CTR}[E_K]$, but with a twist.)

(e) **(2pt)** Suppose we would extend our analysis of question (d) to distinguishers that make 2 oracle queries, both of length exactly ℓ padded message blocks, and both for different nonces. Explain how the security bound of (d) bound would change.

(f) **(2pt)** Assume one does not implement CrAp_K with unique nonces, but rather with random nonces. In other words, for each new evaluation (for a new, or possibly repeated, message M) the implementation selects a random t-bit nonce N and evaluates $\mathsf{CrAp}_K(N, M)$. What is, for your personalized version of the question, the expected number of evaluations an attacker must make in order to obtain a repeated nonce?

(g) **(2pt)** One can break the authenticity of CrAp with high probability, in 0 encryption queries and 2^b forgery attempts. Describe the corresponding distinguisher $\mathcal{D}$.

(h) **(2pt)** One can break the authenticity of CrAp with high probability in 1 encryption query and 2^a forgery attempts. Describe the corresponding distinguisher $\mathcal{D}$.

3. **(16 points)** Let P be a 800-bit permutation, and consider the compression function F^P of your personalized appendix.

Suppose we would place this compression function into a Merkle-Damgård mode of operation of slide 10 of "Cryptographic Hash Functions and Key Derivation".

(a) **(3pt)** What are the chaining value size and the message block size of your scheme?

(b) **(3pt)** Assume that we use plain injective padding that takes a message M, and pads it with a 1 and a sufficiently many number of 0s so that it is of length a multiple of the message block size (i.e., your answer to question (a)). Assume that the length encoding $\mathsf{len}(M)$ is of length exactly the message block size as well (i.e., your answer to question (a)). How many evaluations of the permutation P are needed in order to hash a message of size $|M| = 1234567$ bits in order to get a digest?

Next, we will consider the exact preimage resistance of your compression function in the *ideal permutation model*. Here, we consider an adversary $\mathcal{A}$ that is given a target image, without loss of generality a 0-string of length the chaining value size (i.e., your answer to question (a)). It can then make Q queries to P. These are forward queries $x \mapsto y$ or inverse queries $y \mapsto x$. The adversary wins if one of these Q permutation queries corresponds to an evaluation of your compression function that hits the target image.

(c) **(3pt)** Consider a single permutation query tuple (x, y) (the query direction does not matter). What are *exactly* the conditions that x and y need to satisfy in order to be a valid preimage for the compression function F^P? Concisely explain your answer.

(d) **(2pt)** Suppose that the adversary makes exactly one forward query $x \mapsto y$. Suppose that x meets the conditions stated in question (c). What is the probability that $x \mapsto y$ forms a preimage for F^P? Concisely explain your answer.

(e) **(2pt)** Suppose that the adversary makes exactly one inverse query $y \mapsto x$. Suppose that y meets the conditions stated in question (c). What is the probability that $y \mapsto x$ forms a preimage for F^P? Concisely explain your answer.

(f) **(3pt)** Suppose that the adversary can make Q queries, derive an upper bound on the probability that it finds a preimage for F^P. Concisely explain your answer. Here, you can ignore minor constants. If you were unable to solve questions (d) and (e), you can use "Pr($3d$)" and "Pr($3e$)" as placeholders.

4. **(17 points)** Read the paper on public-key cryptography that is assigned to you (see your personalized appendix). You do **not** need to read the highly technical parts.

(a) **(2pt)** Summarize the content of the paper in at most 1/2 page, focusing on the purpose of the work and the results. As for question 1, use your own words and try to reflect your understanding of the content.

(b) **(1pt)** What kind of public-key cryptosystem is introduced in the paper? No explanation necessary.

(c) **(1pt)** What sort of security is proven for the scheme? If more than one, list all of them.

(d) **(2pt)** What is the hardness assumption that the authors base the security of their scheme on? Write it down formally.

(e) **(3pt)** Give a high-level sketch of how the security of the scheme can be broken if the hard underlying problem could be solved efficiently.

(f) **(2pt)** What is the strongest type of security that the authors claim to achieve? What technique do the authors use to achieve it?

(g) **(1pt)** Explain how this cryptosystem can be used in practice (i.e., for what purpose). If this involves multiple steps, write them down.

(h) **(1pt)** List the recommended parameters proposed by the authors that provide the lowest security level treated in the paper. What is this security level in bits, and for what kind of adversaries?

(i) **(1pt)** What is the size of the public and private key for these parameters in bytes?

(j) **(1pt)** Name a cryptosystem that you have seen in the lectures that achieves the same functionality and has the same claimed security classically (i.e., when the adversary does not have access to a quantum computer).

(k) **(1pt)** Write one strong point of the proposed scheme as an argument for using this scheme in practice.

(l) **(1pt)** Write one weak point of the proposed scheme as an argument for NOT using this scheme in practice.

5. **(33 points)** The computational graph isomorphism (CGI) problem that was mentioned in the lectures can be formulated as follows: *Given two isomorphic graphs $\mathcal{G}_0$ and $\mathcal{G}_1$, find an isomorphism ϕ such that $\mathcal{G}_1 = \phi(\mathcal{G}_0)$.*

In order to proceed without ambiguity, we agree on the following:

- A graphs of size n has n vertices and is represented using a polynomial $\Gamma = \sum_{1 \leq i \leq j \leq n} \alpha_{i,j} x_i x_j$ with $\alpha_{i,j} \in \{0, 1\}$. Here, $\alpha_{i,j} = 1$ means that the vertices i and j are connected.
- An isomorphism ϕ on a graph $\mathcal{G}$ transforms its polynomial $\Gamma = \sum_{1 \leq i \leq j \leq n} \alpha_{i,j} x_i x_j$ to $\Gamma = \sum_{1 \leq i \leq j \leq n} \alpha_{i,j} x_{p(i)} x_{p(j)}$ for some permutation p on $\{1, \dots, n\}$.
- A permutation p on $\{1, \dots, n\}$ is stored in memory as an array $\{p(1), \dots, p(n)\}$ of $\lceil \log_2 n \rceil$-bit numbers.

- The composition of isomorphisms $\phi\psi$ is defined as: $\phi\psi(\mathcal{G}) = \psi(\phi(\mathcal{G}))$.

(a) **(3pt)** Consider the computational problem given in your personalized appendix that for simplicity we will call CGI2. Prove, using a security reduction, that if the CGI problem is hard then CGI2 is hard as well. Calculate exactly the security loss of your security reduction.

Consider now the identification protocol given in your personalized appendix, that for simplicity we will call $\mathsf{ID}_{\text{CGI2}}$.

(b) **(2pt)** Explain the steps of the protocol $\mathsf{ID}_{\text{CGI2}}$ in your own words.

First, we extract a commitment scheme from the first communication step of $\mathsf{ID}_{\text{CGI2}}$.

(c) **(2pt)** Write down formally the commitment scheme, with clearly stated input, output, domain, and range.

(d) **(4pt)** Investigate the binding and hiding properties of the extracted commitment and show whether each of them is perfect, statistical, or computational. You only need to show the strongest property that the commitment scheme possesses.

Next, we focus on the properties of $\mathsf{ID}_{\text{CGI2}}$.

(e) **(1pt)** Show formally that the protocol is complete.

(f) **(3pt)** Show formally that the protocol is sound. How big is the soundness error?

(g) **(2pt)** How many times should the protocol be repeated in order to achieve a $1/2^{\lambda}$ soundness error? See your personalized appendix for the value of the security parameter λ.

(h) **(3pt)** Can it be shown that the protocol is zero-knowledge using the technique introduced in the lectures? If yes, prove that the protocol is zero-knowledge. If no, state why the technique fails.

(i) **(4pt)** Prove that $\mathsf{ID}_{\text{CGI2}}$ is a Σ-protocol.

(j) **(2pt)** Briefly explain how $\mathsf{ID}_{\text{CGI2}}$ can be used as a protocol for identification (entity authentication). What is the advantage of this protocol over the standard password-based entity authentication? Concisely explain your answer.

(k) **(2pt)** Assume that the function used to generate the isomorphisms was implemented badly and it always generates faulty isomorphisms that keep the first k vertices of a graph fixed. Explain exactly how this impacts the security of the protocol. See your personalized appendix for the value of k.

(l) **(2pt)** Turn $\mathsf{ID}_{\text{CGI2}}$ into a digital signature scheme. What kind of security does the digital signature have, and under which hardness assumption?

(m) **(2pt)** Calculate the size of the signature in bytes.

(n) **(1pt)** Think of a way to optimize the size of the signature. One optimization is enough. Briefly explain your answer.