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

Remarks:

- Resits 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 orals 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 resit (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 resit.
- The resit 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 one week for this resit. As a rule of thumb, the difficulty and time effort should be roughly comparable to an ordinary on-campus open book resit **under the assumption that you are well-prepared**.
- The resit 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 resit". Do **NOT** post your questions at the Discord server.

Availability of Exam: Friday, July 8, 2022

Deadline: Friday, July 15, 2022, 23.59 (Dutch time)

(The resit starts on the next page!)

1. **(17 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) **(2pt)** Does the cryptographic scheme operate on fixed-length or variable-length inputs? Concisely explain your answer.

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

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

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

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

   (h) **(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)).

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

   (j) **(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. **(17 points)** Let $k, n \in \mathbb{N}$, and consider the tweakable block cipher $\widetilde{M}$ of your personal appendix.

   The security of $\widetilde{M}$ is measured by its (S)TPRP security (inclusion of the S depends on the use case), where $\widetilde{p}$ is a family of random permutations:

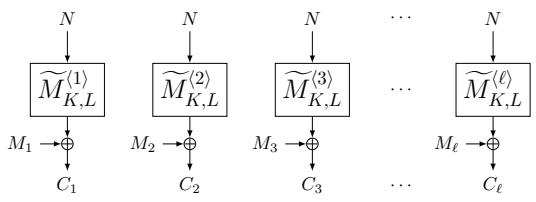
$$\mathbf{Adv}^{\mathrm{tprp}}_{\widetilde{M}}(\mathcal{D}) = \left|\mathbf{Pr}\left(\mathcal{D}^{\widetilde{M}_{K,L}} = 1\right) - \mathbf{Pr}\left(\mathcal{D}^{\widetilde{p}} = 1\right)\right|,$$
$$\mathbf{Adv}^{\mathrm{stprp}}_{\widetilde{M}}(\mathcal{D}) = \left|\mathbf{Pr}\left(\mathcal{D}^{\widetilde{M}_{K,L},\widetilde{M}^{-1}_{K,L}} = 1\right) - \mathbf{Pr}\left(\mathcal{D}^{\widetilde{p},\widetilde{p}^{-1}} = 1\right)\right|.$$

   We remark that $\widetilde{M}$ turns out to be secure in the TPRP-setting but insecure in the STPRP-setting.

   (a) **(5pt)** Mount an STPRP security attack against $\widetilde{M}$ in at most 4 queries. Concisely explain your answer. [Hint: devote the first 2 queries to recovering subkey L.]

   (b) **(4pt)** Compute the success probability of the attack of (a).

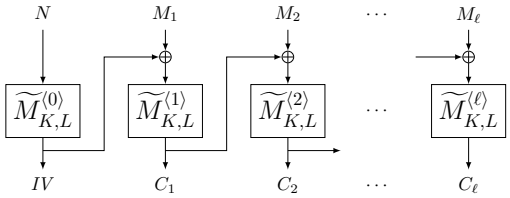
   (c) **(4pt)** Suppose we use $\widetilde{M}$ in the following variant of counter mode, CTR-TBC, where $\langle i \rangle$ denotes the encoding of i as an n-bit tweak:



Derive an upper bound on $\max_{\mathcal{D}} \mathbf{Adv}^{\mathrm{prf}}_{\mathsf{CTR\text{-}TBC}}(\mathcal{D})$, where the maximum is taken over all distinguishers that make q oracle queries, each of length exactly ℓ blocks. Distinguisher $\mathcal{D}$ is not allowed to repeat nonces. In this question, you can assume that $\widetilde{M}$ is TPRP-secure, in other words, we expect an upper bound of the form

$$\max_{\mathcal{D}} \mathbf{Adv}^{\mathrm{prf}}_{\mathsf{CTR\text{-}TBC}}(\mathcal{D}) \leq \text{something} + \mathbf{Adv}^{\mathrm{tprp}}_{\widetilde{M}}(\text{something})\,.$$

(d) **(4pt)** Suppose we put $\widetilde{M}$ in the following variant of CBC mode:



Is this wise? Concisely explain your answer (no need to give a proof or attack).

3. **(16 points)** Let $b, k, m \in \mathbb{N}$ with $b = 2k + m$. Let $f : \{0,1\}^b \to \{0,1\}^b$ be a public random function. (Typically, one would take a permutation, but we consider a random function for simplicity of the exercise.) Consider the nonce-based authenticated encryption scheme SmaDa (for Small Data) of your personal appendix. Also refer to your personalized appendix for the actual values b, k, m for which you have to make the exercise. [Hint: visualize SmaDa by making a drawing of the scheme (for yourself; no need to hand in).]

Confidentiality of this scheme is defined as follows:

$$\mathbf{Adv}^{\mathrm{conf}}_{\mathsf{SmaDa}}(\mathcal{D}) = \left|\mathbf{Pr}\left(\mathcal{D}^{\mathsf{SmaDa}^f_K, f} = 1\right) - \mathbf{Pr}\left(\mathcal{D}^{\$, f} = 1\right)\right|\,.$$

Here, K is a random k-bit key and f is a random b-bit function. Distinguisher $\mathcal{D}$ may not repeat nonces.

(a) **(4pt)** Suppose you are a distinguisher that make q oracle queries (to SmaDa^f_K or \$) for distinct nonces, and t primitive queries (to f). What is the best (and only) way to distinguish (SmaDa^f_K, f) from $(\$, f)$? What is the approximate probability that your attack succeeds (you can be off by a constant factor)? Concisely explain your answer.

(b) **(3pt)** Suppose we release the nonce-uniqueness condition, so $\mathcal{D}$ may repeat nonces. Demonstrate how to break the confidentiality of SmaDa.

(c) **(4pt)** Compute $\mathbf{Adv}^{\mathrm{conf}}_{\mathsf{SmaDa}}(\mathcal{D})$ for your distinguisher of question (b).

(d) **(3pt)** Describe how the authenticated decryption function SmaDa^{-1}_K is performed.

(e) **(2pt)** What is the success probability of a generic forgery attempt against SmaDa? Concisely explain your answer, including how this generic attack operates.

(The resit continues on the next page!)

4. **(20 points)** Let g be a generator of a group G of prime order q. Both g and G are public parameters. Consider the encryption cryptosystem given in your personal appendix.

   (a) **(4pt)** Show that the computational Diffie-Hellman (CDH) problem for G is hard if the Decisional Diffie-Hellman (CDH) problem for G is hard.

   (b) **(4pt)** Show that your cryptosystem is OW-CPA secure if the computational Diffie-Hellman (CDH) problem for G is hard.

   (c) **(3pt)** Show that Eve can create a valid ciphertext of $N \cdot M$ from a valid ciphertext of M, where N is given in your personal appendix.

   (d) **(3pt)** Show how to mount a chosen ciphertext attack against the cryptosystem.

   (e) **(6pt)** Describe a scenario when you can mount a meet-in-the-middle attack against the cryptosystem (or a modified version of it) and describe the steps of the attack.

5. **(30 points)** Let g be a generator of a group G of prime order q. Both g and G are public parameters. Assume that finding discrete logarithms in the group G is a computationally hard problem.

   (a) **(3pt)** Recall from the lecture slides Schnorr's Zero-Knowledge protocol, and its more efficient single round variant, Schnorr's identification protocol. For each of the properties "soundness", "special soundness", and "zero-knowledgness", state which is satisfied by which of the two protocols. How many times should the first protocol be repeated to reach the same soundness error as the second protocol?

   Consider now the protocol given in your personalized appendix that we will call $\mathsf{ID}_{\text{Schnorr2}}$.

   (b) **(3pt)** Explain the steps of the protocol $\mathsf{ID}_{\text{Schnorr2}}$ in your own words.

   (c) **(3pt)** Show formally that the protocol is complete.

   (d) **(4pt)** Show formally that the protocol is special sound under the computational discrete logarithm assumption in the group G.

   (e) **(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.

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

   (g) **(6pt)** Turn $\mathsf{ID}_{\text{Schnorr2}}$ into a digital signature scheme. Write explicitly the signing and verification algorithms. What kind of security does the digital signature have, and under which hardness assumption?

   (h) **(3pt)** Calculate the size of the signature in bytes. See your personalized appendix for the bit length of q.

   (i) **(3pt)** Compare the size of the signature to that of Schnorr's signature. Which one has smaller signatures? Concisely explain your answer.