FOR INTERNAL SCRUTINY (date of this version: 25/3/2024)

UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INTRODUCTION TO MODERN CRYPTOGRAPHY PG AND UG

November 2019

23:50 to 23:59

INSTRUCTIONS TO CANDIDATES

1. **Note that ALL QUESTIONS ARE COMPULSORY.**
2. **DIFFERENT QUESTIONS MAY HAVE DIFFERENT NUMBERS OF TOTAL MARKS. Take note of this in allocating time to questions.**
3. **This is an OPEN BOOK examination: books, notes and other written or printed material MAY BE CONSULTED during the examination. The use of electronic devices or electronic media is NOT PERMITTED.**

Year 4 Courses

Convener: ITO-Will-Determine
External Examiners: ITO-Will-Determine

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

FOR INTERNAL SCRUTINY (date of this version: 25/3/2024)

All your answers need to be justified via a formal argument.

1. **Symmetric-key Cryptography**

   (a) (Shift cipher) Assume a user has a 6-character password and uses the Shift cipher to encrypt the password. The message space is $\mathcal{M} = \{actrpg, vionlu, pcihfo\}$. Show whether the scheme is or is not perfectly secret. [*4 marks*]

   (b) (PRF and PRG) Let F be a length-preserving keyed pseudorandom function with key (and input) of length n. Let G be a pseudorandom generator that takes as input a string of length n, and outputs a string of length $2n$. State whether each of the following PRF candidates is or is not a pseudorandom function. If yes, prove it; if not, show a distinguisher that succeeds with non-negligible probability (the input of each PRF is a key $k \in \{0,1\}^n$ and an input $x \in \{0,1\}^n$):

      i. $F'(k,x) \stackrel{\text{def}}{=} F(\tilde{k}_1 \ldots \tilde{k}_n, x)$, where $\tilde{k} \leftarrow G(k)$, and we parse $\tilde{k}$ as $\tilde{k}_1 \ldots \tilde{k}_{2n}$ (i.e., $\tilde{k}_1 \ldots \tilde{k}_n$ represents the first n bits returned by the evaluation of the PRG). [*5 marks*]

      ii. $F''(k,x) \stackrel{\text{def}}{=} F(x,k)$ [*5 marks*]

      iii. $F'''_k(x) \stackrel{\text{def}}{=} F(k,x) \oplus \overline{F(k,x)}$, where $\overline{F(k,x)}$ means flipping every bit of $F(k,x)$ (e.g., if $F(k,x) = 010001$ then $\overline{F(k,x)} = 101110$) [*4 marks*]

   (c) (MAC) A secure MAC ensures that an adversary cannot generate a valid tag on a new message that was never previously authenticated. However, it does not rule out the possibility that an attacker might be able to generate a new *tag* on a previously authenticated message. In some settings, it is useful to consider a stronger definition of security for MACs where such behavior is ruled out. We consider a modified experiment $\mathsf{Mac\text{-}strongForge}$ that is defined in exactly the same way as $\mathsf{Mac\text{-}forge}$, except that now the set $\mathcal{M}$ contains *pairs* of oracle queries and their associated responses. (That is, $(m,t) \in \mathcal{M}$ if $\mathcal{A}$ queried $\mathsf{Mac}_k(m)$ and received in response the tag t). We say that the adversary wins in $\mathsf{strongForge}$ (i.e., $\mathsf{strongForge}_{\mathcal{A},\Pi}(n) = 1$) iff. $\mathcal{A}$ outputs (m,t) such that $\mathsf{Vrfy}_k(m,t) = 1$ and $(m,t) \notin \mathcal{M}$.

   Definition 1. *A message authentication code* $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ *is a* strong MAC*, if for all probabilistic polynomial time adversary* $\mathcal{A}$*, there is a negligible function* ϵ *such that*

   $$\Pr[\mathsf{Mac\text{-}strongForge}_{\mathcal{A},\Pi}(n) = 1] \leq \epsilon(n).$$

   Prove or disprove the following claim.

   Any secure MAC scheme (under the standard definition) $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ is also a strong MAC scheme.
   (Hint: this claim does not hold.). [*10 marks*]

2. **Hash Functions**

(a) Let $H_1 : \{0,1\}^* \to \{0,1\}^n$ and $H_2 : \{0,1\}^* \to \{0,1\}^n$ be any two hash functions. Suppose that *either* one of them is collision-resistant (i.e., only one of this hash function is collision-resistant, but we do not know which one). State whether each of the following hash-function candidates is collision-resistant or not, and formally argue your claim.

i. $H'(x) \stackrel{\text{def}}{=} H_1(x) \parallel H_2(x)$. *[5 marks]*

ii. $H''(x) \stackrel{\text{def}}{=} H_1(x_1 \dots x_{n-1}||0)||H_2(x_1 \dots x_{n-1}||1)$
(where $x_1 \dots x_{n-1}$ denote the first $n-1$ bits of x and $||$ is the concatenation operator). *[5 marks]*

3. **Public-key encryption**

(a) Let $\texttt{Gen}$ be the key-generation algorithm for the El Gamal encryption scheme. Consider the following variation of El Gamal encryption (note that this differs from El Gamal only in how the encryption and the decryption algorithms work), $(\texttt{Gen}', \texttt{Enc}', \texttt{Dec}')$, that works as follows:

$\texttt{Gen}'(1^n)$:
$(g^k, k) \leftarrow \texttt{Gen}(1^n)$
$pk \leftarrow g^k$
$sk \leftarrow k$
return (pk, sk)

$\texttt{Enc}'(pk, m)$:
$r_1 \stackrel{\$}{\leftarrow} \mathcal{Z}_p$
$r_2 \stackrel{\$}{\leftarrow} \mathcal{Z}_p$
$ct_1 \leftarrow pk^{r_1} \cdot pk^{r_2} \cdot m$
$ct_2 \leftarrow g^{r_1}$
$ct_3 \leftarrow g^{r_2}$
return (ct_1, ct_2, ct_3)

$\texttt{Dec}'(sk, (ct_1, ct_2, ct_3))$:
$m \leftarrow \dfrac{ct_1}{ct_2^{sk} \cdot ct_3^{sk}}$
return m

Prove that this new encryption scheme is IND-CPA secure. *[5 marks]*

(b) (Signatures) Let H be a collision-resistant cryptographic hash function and consider the following candidate signature scheme $\Pi = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$.

i. $\mathsf{KeyGen}(1^n)$ uniformly at random selects $2n$ elements from $\{0,1\}^n$ denoted by $x_1^0, x_1^1, x_2^0, x_2^1, \ldots x_n^0, x_n^1$. Then it computes $y_i^b = H(x_i^b)$ for all $1 \leq i \leq n$ and $b \in \{0,1\}$. The secret key SK and the public key PK are defined as follows

$$SK = \begin{pmatrix} x_1^0 & x_2^0 & \ldots & x_n^0 \\ x_1^1 & x_2^1 & \ldots & x_n^1 \end{pmatrix} \; PK = \begin{pmatrix} y_1^0 & y_2^0 & \ldots & y_n^0 \\ y_1^1 & y_2^1 & \ldots & y_n^1 \end{pmatrix}.$$

ii. $\mathsf{Sign}(m, PK, SK)$. Let m be equal to $m_1 m_2 \ldots m_n$ where m_i denotes the i-th bit of m. The signature algorithm returns $(x_1^{m_1}, x_2^{m_2}, , \ldots x_n^{m_n})$ as the signature s.

iii. $\mathsf{Verify}(PK, m, s)$, where $m = m_1 m_2 \ldots m_n$ and $s = (s_1, s_2, , \ldots s_n)$. Return 1 iff $H(s_i) = y_i^{m_i}$ for all $1 \leq i \leq n$.

We say that a signature scheme is k-time-secure if the adversary is allowed only k queries to the signature oracle. It should be easy to see that this scheme is 1-time-secure (due to the security of the hash function). But is the scheme 2-time-secure? Provide formal arguments to support your answer. [*7 marks*]