MIT 6.5620/6.875/18.425: Foundations of Cryptography **Due: Oct. 28, 2026**

Problem Set 3: Solutions

Your name **Your Collaborators**

Total Number of Points: 40.
Instructions: Solutions should be typeset in LaTeX and submitted as PDF by 11:59:59 PM ET on the due date. You can collaborate on the psets in small groups (of up to three), but must write your solutions separately, in your own words. This means not copying from a board nor from your collaborator's computer.

We strongly recommend ***not*** using AI tools to solve your problem sets. You are responsible for understanding everything you submit and you must be able to explain and justify every part of your solution. In particular, the oral problem set review (10% of the course grade) will evaluate your understanding of the solutions you submit. In all cases, you must list your collaborators, human or machine.

Problem 1. Authenticated Encryption. (12 points) Suppose Alice and Bob want to transmit messages to each other in a *private* and *authenticated* manner.

Let $(\mathsf{Gen}_{\mathsf{Enc}}, \mathsf{Enc}, \mathsf{Dec})$ be an IND-CPA secure symmetric encryption scheme, and let $(\mathsf{Gen}_{\mathsf{MAC}}, \mathsf{Mac}, \mathsf{Ver})$ be an EUF-CMA secure message authentication scheme (that is, existentially unforgeable under a chosen message attack). *You may assume in this problem that* $(\mathsf{Gen}_{\mathsf{Enc}}, \mathsf{Enc}, \mathsf{Dec})$ *has perfect correctness.*

Alice and Bob share the keys $k_1 \leftarrow \mathsf{Gen}_{\mathsf{Enc}}$ and $k_2 \leftarrow \mathsf{Gen}_{\mathsf{MAC}}$, and now they want to transmit a message M. For each of the following protocols, determine whether it is (i) IND-CPA secure and (ii) EUF-CMA secure. For each of (i) and (ii), if the protocol is secure, give a proof of its security, and if the protocol is not secure, give a counterexample that breaks its security.

Solution
There are two different definitions of EUF-CMA security. We will accept both answers as long as you are clear about which definition you are targeting.

- *Strong EUF-CMA*: The adversary is given access to a Ver oracle and a Mac oracle. The adversary wins if it outputs a message-tag pair (m, τ^*) such that $\mathsf{Ver}_{k_2}(m, \tau^*) = 1$ and (m, τ^*) has not been answered by the Mac oracle. In this definition, the adversary is allowed to forge a new tag on a message that has been queried to the Mac oracle.
- *EUF-CMA*: The adversary is given access to a Ver oracle and a Mac oracle. The adversary wins if it outputs a message-tag pair (m^*, τ^*) such that $\mathsf{Ver}_{k_2}(m^*, \tau^*) = 1$ and m^* has not been queried to the Mac oracle. In this definition, the adversary is forced to generate a tag on a new message that has not been queried to the Mac oracle.

(a) (4 points) $\mathsf{Enc}_{k_1}(M \| \mathsf{Mac}_{k_2}(M))$
Solution
IND-CPA secure: Since our new scheme is composed only of a ciphertext, using an encryption scheme that itself is computationally indistinguishable, this follows immediately. Formally, suppose there exists a PPT adversary $\mathcal{A}$ who succeeds in distinguishing our new scheme with non-negligible probability ϵ. We construct an adversary $\mathcal{A}'$ who does the following:

1. Sample a random secret key k_2 to be used for the MAC scheme Mac.
2. For each query M requested by $\mathcal{A}$ (on the *new* scheme), query the message $M \| \mathsf{Mac}_{k_2}(M)$ to the Enc oracle, and receive back the corresponding ciphertext C. Send C to $\mathcal{A}$, as the response to his query.

3. When $\mathcal{A}$ gives a pair of messages M_0, M_1, pass the pair $M_0 \| \mathsf{Mac}_{k_2}(M_0), M_1 \| \mathsf{Mac}_{k_2}(M_1)$ to $\mathcal{A}$'s Enc oracle, and receive back a challenge ciphertext C_b. Send C_b to $\mathcal{A}$ as the challenge ciphertext.
4. When $\mathcal{A}$ outputs a guess b', output b' as $\mathcal{A}'$'s guess.

If $\mathcal{A}$ succeeds in distinguishing the new scheme, then $C_{b'}$ is actually the encryption of $M_b \| \mathsf{Mac}_{k_2}(M_b)$, where M_b is the message that $\mathcal{A}$ queried on. Thus, $\mathcal{A}'$ has successfully distinguished between the two messages $M_0 || \mathsf{Mac}_{k_2}(M_0)$ and $M_1 || \mathsf{Mac}_{k_2}(M_1)$, which contradicts the IND-CPA security of Enc. Thus, the new scheme is IND-CPA secure.

For ***Strong*** **EUF-CMA security**, this scheme is **not secure**. *Forgeable counterexample*: Consider an encryption scheme Enc that encrypts the input and adds a random bit at the end of ciphertext. That is, $\mathsf{Enc}'_{k_1}(M) = \mathsf{Enc}_{k_1}(M) \| b$, where b is a random bit. This scheme is still IND-CPA secure. However, we can forge a new valid tag of the message by changing the last bit of $\mathsf{Enc}_{k_1}(M \| \mathsf{Mac}_{k_2}(M))$. This will still decrypt to $M \| \mathsf{Mac}_{k_2}(M)$, which is a valid tag for M.

For **EUF-CMA security** (weaker version), this scheme is secure. *Unforgeability*: We show a reduction to the unforgeability of the underlying MAC scheme. Suppose there exists a PPT adversary A who succeeds in forging our new scheme with non-negligible probability ϵ. We construct an adversary A' who does the following:

1. Sample a random secret key k_1 to be used for the encryption scheme Enc.
2. For each query M requested by A (on the *new* scheme), query the same message M and receive τ_M (corresponding to $\mathsf{Mac}(M)_{k_2}$ in the original scheme).
3. Generate the encryption $C = \mathsf{Enc}_{k_1}(M \| \tau_M)$, using the (known) secret key k_1, and send C to A, as the response to his query.
4. When A outputs a purported forgery (M^*, C^*) for the new scheme, decrypt $\tau^* = \mathsf{Dec}_{k_1}(C^*)$, and submit the pair (M^*, τ^*) as A''s forgery for the scheme Mac.

If (M^*, C^*) is a valid forgery for the new scheme, then we have that (1) M^* has not been queried by A (and also was not queried by A'), and (2) the relation $\mathsf{Ver}_{k_2}(M^*, tag^*) = 1$ holds with respect to the underlying scheme Mac, where $\tau^* = \mathsf{Dec}_{k_1}(C^*)$. That is, if A produces a valid forgery for the new scheme, then A' successfully produces a forgery for the underlying scheme Mac. By the security of Mac, it follows that the new scheme is also unforgeable.

(b) (4 points) $\mathsf{Enc}_{k_1}(M) \| \mathsf{Mac}_{k_2}(M)$

Solution

Not IND-CPA secure. This is because the MAC does not provide any sort of *privacy* guarantees. More formally, for any existentially unforgeable scheme Mac, consider a new scheme $\mathsf{Mac}'_k(M) := (M, \mathsf{Mac}_k(M))$. Then Mac' is clearly also existentially unforgeable (since a forgery for Mac' immediately gives a forgery for Mac), but sending $\mathsf{Mac}'_{k_2}(M)$ in the clear obviously does not maintain privacy, as it reveals the entire message M.

For ***Strong*** **EUF-CMA security**, this scheme is **not secure**. *Forgeable counterexample*: We can use the same encryption scheme from before. Consider an encryption scheme Enc that encrypts the input and adds a random bit at the end of ciphertext. That is, $\mathsf{Enc}'_{k_1}(M) = \mathsf{Enc}_{k_1}(M) \| b$, where b is a random bit. This scheme is still IND-CPA secure. As the whole $\mathsf{Enc}_{k_1}(M) \| \mathsf{Mac}_{k_2}(M)$ is a tag. We can forge a new valid tag of the message by changing the last bit of $\mathsf{Enc}_{k_1}(M)$ and outputting that concatenated with $\mathsf{Mac}_{k_2}(M)$. This is a valid new tag for M.

For **EUF-CMA security** (weaker version), this scheme is **secure**. *Unforgeability*: We show a reduction to the unforgeability of the underlying MAC scheme. Suppose there exists a PPT adversary A who succeeds in forging our new scheme with non-negligible probability ϵ. We construct an adversary A' who does the following:

1. Sample a random secret key k_1 to be used for the encryption scheme Enc.
2. For each query M requested by A (on the *new* scheme), query the same message M and receive τ_M (corresponding to $\mathsf{Mac}(M)_{k_2}$ in the original scheme).
3. Generate the encryption $C = \mathsf{Enc}_{k_1}(M) \| \tau_M$, using the (known) secret key k_1, and send C to A, as the response to his query.
4. When A outputs a purported forgery (M^*, C^*) for the new scheme, take τ^* as the latter part of C^*, and submit the pair (M^*, τ^*) as A''s forgery for the scheme Mac.

If (M^*, C^*) is a valid forgery for the new scheme, then we have that (1) M^* has not been queried by A (and also was not queried by A'), and (2) the relation $\mathsf{Ver}_{k_2}(M^*, \tau^*) = 1$ holds with respect to the underlying scheme Mac, where τ^* is the latter part of C^*. That is, if A produces a valid forgery for the new scheme, then A' successfully produces a forgery for the underlying scheme Mac. By the security of Mac, it follows that the new scheme is also unforgeable.

(c) (4 points) $\mathsf{Enc}_{k_1}(M) \| \mathsf{Mac}_{k_2}(\mathsf{Enc}_{k_1}(M))$

Solution

IND-CPA secure.: For this property, note that the scheme uses the *same* ciphertext $\mathsf{Enc}_{k_1}(M)$ in both places. Thus, distinguishing between any two pairs $\big(\mathsf{Enc}_{k_1}(M), \mathsf{Mac}_{k_2}(\mathsf{Enc}_{k_1}(M))\big)$ and $\big(\mathsf{Enc}_{k_1}(M'), \mathsf{Mac}_{k_2}(\mathsf{Enc}_{k_1}(M'))\big)$ implies distinguishing the distribution $\mathsf{Enc}_{k_1}(M)$ from the distribution $\mathsf{Enc}_{k_1}(M')$, which contradicts the indistinguishability of the underlying encryption scheme Enc.

For both ***Strong*** **EUF-CMA** and **EUF-CMA security**, this scheme is **secure**. Let A be an adversary who forges the *new* scheme with non-negligible probability ϵ. We construct an adversary A' who forges in the *original* scheme Mac with success probability ϵ. This is done in a way similar to the reduction above. Namely,

1. A' samples his own secret key k_1 for the encryption scheme.
2. For each query M made by A on the new scheme, A' queries his Mac challenger on the *encrypted* message $\mathsf{Enc}_{k_1}(M)$, and receives back some value τ. A' sends the pair $(\mathsf{Enc}_{k_1}(M), \tau)$ to A as the response to his M query on the new scheme.
3. Finally, A submits a forgery $(M^*, (C^*, \tau^*))$ for the new scheme. A' submits the pair (C^*, τ^*) as his forgery (on message C^*).

If A succeeded in forging the new scheme, then (1) A did not query on M^*, (2) $M^* = \mathsf{Dec}_{k_1}(C^*)$, and (3) τ^* is a valid tag for C^*. Recall that our constructed adversary A' made Mac queries on the values $\mathsf{Enc}_{k_1}(M_i)$, where $M_1, M_2, \ldots$ were the messages queried by A. Due to IND-CPA security, with high probability C^* has not already occurred as one of A''s queries to the Mac oracle. (That is, it is very unlikely to obtain the same ciphertext by two queries of a ppt adversary to an encryption oracle). Thus, the fact that τ^* is a valid tag for C^* means that A' has (with high probability) successfully forged a new tag.

Problem 2. Average case to worst case DDH. (12 points) Let $\mathbb{G}$ be a cyclic group of prime order q generated by $g \in \mathbb{G}$. Let $n = \texttt{len}(q)$, be the number of bits required to represent q. Assume that we have an efficient algorithm $\mathcal{A}$ that can distinguish a random **DH** triple from a non-**DH** triple with non-negligible advantage, that is, for $\alpha, \beta, \gamma \leftarrow \mathbb{Z}_q, g \leftarrow \mathbb{G}$ sampled uniformly at random, we have

$$\left|\mathbf{Pr}[\mathcal{A}(g, g^\alpha, g^\beta, g^\gamma) = 1] - \mathbf{Pr}[\mathcal{A}(g, g^\alpha, g^\beta, g^{\alpha\beta}) = 1]\right| \geq \frac{1}{p(n)} \tag{1}$$

for some polynomial $p(n)$. Note that this algorithm requires α, β, γ, g to be sampled at random.

(a) (9 points) Given $\mathcal{A}$, construct an efficient algorithm $\mathcal{B}$ that can distinguish **DH** triples for all inputs with negligible error, that is, $\forall \alpha, \beta, \gamma \in \mathbb{Z}_q$ and $\forall g \in \mathbb{G}$ such that $\gamma \neq \alpha\beta$ and g is not the identity, we have

$$\left|\mathbf{Pr}[\mathcal{B}(g, g^\alpha, g^\beta, g^\gamma) = 1] - \mathbf{Pr}[\mathcal{B}(g, g^\alpha, g^\beta, g^{\alpha\beta}) = 1]\right| \geq 1 - \mu(n), \tag{2}$$

for some negligible function $\mu(n)$.

Solution

We need to use self-reduction to show that if this problem is solved in the average case where g, α, β, γ are sampled randomly, then this is solved with non-negligible probability for all feasible inputs. Construct a $\mathcal{B}$ which takes input $(g, g^\alpha, g^\beta, g^\gamma)$. It samples $r, r_0, r_1, r_2 \leftarrow \mathbb{Z}_q$ uniformly at random. It gives $\mathcal{A}$, the input

$$(g^{r_0}, (g^{r_0})^{r\alpha + r_1}, (g^{r_0})^{\beta + r_2}, (g^{r_0})^{r\gamma + r\alpha r_2 + \beta r_1 + r_1 r_2}), \tag{3}$$

and returns the same output as $\mathcal{A}$. There are many things to break down here.

Firstly, let's show that this is the input which $\mathcal{A}$ expects. If $\gamma = \alpha\beta$, then the input is

$$(g^{r_0}, (g^{r_0})^{r\alpha + r_1}, (g^{r_0})^{\beta + r_2}, (g^{r_0})^{(r\alpha + r_1)(\beta + r_2)}), \tag{4}$$

which is what we expect.

Now, we want to show that this is close to the distribution where $g \leftarrow \mathbb{G}$ and $\alpha, \beta, \gamma \leftarrow \mathbb{Z}_q$ are sampled uniformly at random. g^{r_0} is going to be random generator of $\mathbb{G}$ independent of everything else. $r\alpha + r_1$ and $\beta + r_2$ are also distributed uniformly randomly independent of each other. Let's look at $r\gamma + r\alpha r_2 + \beta r_1 + r_1 r_2 = r(\gamma - \alpha\beta) + \underbrace{(r\alpha + r_1)(\beta + r_2)}_{dependent}$. If $\gamma - \alpha\beta \neq 0$, then r randomizes the first term and the expression is distributed uniformly at random in $\mathbb{Z}_q$, independent of the others. So, this input satisfies the distribution which $\mathcal{A}$ needs.

Now, we need to show that $\mathcal{B}$ can compute the input for $\mathcal{A}$ in polynomial time. Given, g^x, one can compute $g^{xr_0 + r_1}$ in polynomial time. We use repeated squaring on g^x to get g^{xr_0}, on g to get g^{r_1}. Then, multiply them to get $g^{xr_0 + r_1}$. Similarly, once we have r, r_0, r_1, r_2, one can apply repeated squaring and multiplication on the terms of input to get

$$(g^{r_0}, g^{r_0 r\alpha + r_0 r_1}, g^{r_0\beta + r_0 r_2}, g^{r_0 r\gamma + r_0 r\alpha r_2 + r_0\beta r_1 + r_0 r_1 r_2}), \tag{5}$$

which is the input which we give to $\mathcal{A}$.

Hence, $\mathcal{B}$ succeeds with the same probability as $\mathcal{A}$. So, we have constructed a $\mathcal{B}$ which on all inputs

$$\left|\mathbf{Pr}[\mathcal{B}(g, g^\alpha, g^\beta, g^\gamma) = 1] - \mathbf{Pr}[\mathcal{B}(g, g^\alpha, g^\beta, g^{\alpha\beta}) = 1]\right| \geq \frac{1}{p(n)} \tag{6}$$

Now, we can apply the $\mathcal{B}$ polynomial number of times and take the majority vote. By using the Chernoff bound, we can get the error probability to be negligible.

(b) (3 points) Given $\mathcal{A}$, construct an efficient algorithm $\mathcal{B}'$ that can distinguish, with non-negligible probability, (g, g^α, g^β) from $(g, g^\alpha, g^{\alpha^2})$ for all $\alpha, \beta \in \mathbb{Z}_q$ and generator $g \in \mathbb{G} \setminus \{\mathbf{1}\}$, given $\beta \neq \alpha^2$.

Solution

Construct the above $\mathcal{B}$ and give it the input $(g, g^\alpha, g^\alpha, g^\beta)$. We can distinguish if $\beta \stackrel{?}{=} \alpha^2$. From part (a), we know $\mathcal{B}$ can accept such an input, so this works.

Problem 3. Circular Security. (8 points) In this problem, we consider a security property (of encryption schemes) called circular security: namely, security even when an adversary is given an encryption of the secret key sk.
One variant of circular security for secret key encryption schemes is defined as follows:

Definition 1 (Circularly Secure SKE)*A secret key encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be circularly secure if for all p.p.t. oracle algorithms* $\mathcal{A}^{\mathcal{O}(\cdot)}$*, we have that*

$$\left| \mathop{\mathbf{Pr}}_{sk \leftarrow_R \mathsf{Gen}(1^n)} [\mathcal{A}^{Left_{sk}(\cdot)}(c^*) = 1] - \mathop{\mathbf{Pr}}_{sk \leftarrow_R \mathsf{Gen}(1^n)} [\mathcal{A}^{Right_{sk}(\cdot)}(c^*) = 1] \right| \leq \mathsf{negl}(n), \tag{7}$$

where $c^* \leftarrow \mathsf{Enc}(sk, sk)$*,* $Left_{sk}(\cdot)$ *is an oracle that on input* (m_L, m_R) *outputs an encryption of* m_L*, and* $Right_{sk}(\cdot)$ *is an oracle that on input* (m_L, m_R) *outputs an encryption of* m_R*.*

Think about why this is a reasonable definition and how it is similar to the IND-CPA security discussed in lectures.
An analogous definition for public key encryption is as follows.

Definition 2 (Circularly secure PKE)*A public key encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be circularly secure if any p.p.t. algorithm* $\mathcal{A}$ *wins the following game (interacting with a challenger) with probability at most* $\frac{1}{2} + \mathsf{negl}(n)$*:*

(i.)The challenger samples $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$ *and sends* (pk, c^*) *to* $\mathcal{A}$*, where* $c^* \leftarrow \mathsf{Enc}(pk, sk)$ *is a ciphertext where the message is the secret key.*

(ii.)$\mathcal{A}$ *sends two messages* (m_0, m_1) *to the challenger.*

(iii.)The challenger selects $b \leftarrow_R \{0, 1\}$ *and sends* $c_b \leftarrow \mathsf{Enc}(pk, m_b)$ *to* $\mathcal{A}$*.*

(iv.)$\mathcal{A}$ *outputs a bit* $\tilde{b}$*. We say that* $\mathcal{A}$ *wins if* $\tilde{b} = b$*.*

It turns out that not every IND-CPA secure public key encryption scheme is also circularly secure. Construct a public-key encryption scheme which is IND-CPA secure but not circularly secure, relying only on the existence of public-key encryption schemes. Prove that your scheme is IND-CPA secure but not circularly secure.

Solution

We provide a counterexample. In the following solution, we convert any public key encryption scheme into an encryption scheme with a trivial "secret-key recognition" process, which will prove useful when constructing the counterexample.

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ denote a public key encryption scheme with the following parameters: $\mathsf{Gen}(1^n; r)$ uses randomness $r \leftarrow \{0,1\}^n$ (using a PRG, we can guarantee this without loss of generality), and Enc encrypts messages in $\{0,1\}^n$.

We now define an encryption scheme $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ which will remain semantically secure but *will not* be circular secure:

- $\mathsf{Gen}'(1^n; r)$ computes $(pk, sk) = \mathsf{Gen}(1^n; r)$ and ouputs $(pk', sk') = (pk, r)$.
- $\mathsf{Enc}'(pk, m)$ does the following:
  1. If $\mathsf{Gen}(1^n; m) = (pk, x)$ for some $x \in \{0,1\}^n$, then output $\mathsf{Enc}(pk, m)||\perp||m$, where $\perp$ is a special symbol not used in ordinary ciphertexts.
  2. Otherwise, output $\mathsf{Enc}(pk, m)$.
- $\mathsf{Dec}'(r, c)$ outputs $\mathsf{Dec}(sk, c')$, where $(pk, sk) = \mathsf{Gen}(1^n; r)$ and c' is the substring of c occuring before the $\perp$ symbol, if one exists.

Perfect correctness of $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ is clear by construction. Moreover, this scheme is manifestly not circular secure; an encryption $\mathsf{Enc}(pk', sk')$ always reveals sk', so an adversary given such an encryption can clearly recover the secret key (and hence break security).

Thus, it suffices to show that $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ is semantically secure. Suppose that some efficient adversary $\mathcal{A}(pk)$ wins the computational indistinguishability game for $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ with non-negligible advantage. We will then construct an adversary $\mathcal{B}$ which breaks the semantic security of $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$. $\mathcal{B}$ works as follows: $\mathcal{B}(pk)$ runs $\mathcal{A}(pk)$ to produce a message pair (m_0, m_1) as its challenge. Then, $\mathcal{B}$ calls $\mathsf{Gen}(1^n; m_0)$ and $\mathsf{Gen}(1^n; m_1)$; if either of the calls $\mathsf{Gen}(1^n; m_b) = (pk', x)$ produces $pk = pk'$ as the public key, then $\mathcal{B}$ uses $(0^n, 1^n)$ as its own challenge and outputs $\mathsf{Dec}(x, c^*)$ as its answer, where c^* is the challenge ciphertext. Otherwise, $\mathcal{B}$ uses (m_0, m_1) as its own challenge and returns the output of $\mathcal{A}(c^*)$.

To see that this breaks semantic security of $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, we note that if $\mathsf{Gen}(1^n; m_0)$ or $\mathsf{Gen}(1^n; m_1)$ produces (pk, x) as output, then $\mathcal{B}$ will always win its indistinguishability game by the perfect correctness of $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ (since (pk, x) was produced by Gen, it must be the case that $\mathsf{Dec}(x, \mathsf{Enc}(pk, m)) = m$ with probability 1). Thus, $\mathcal{B}$ succeeds *strictly more often* than $\mathcal{A}$ does; namely, whenever $\mathcal{A}$ succeeds, either $\mathsf{Gen}(m_0)$ or $\mathsf{Gen}(m_1)$ produces (pk, x), in which case $\mathcal{B}$ wins, or $\mathcal{B}$ fully executes $\mathcal{A}$[1] and also wins. Thus, we conclude that $\mathcal{B}$ breaks the semantic security of $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, so if $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is secure, then so is $(\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$.

[1] Since $\mathcal{B}$ explicitly checked the output of $\mathsf{Gen}(1^n; m_0)$ and $\mathsf{Gen}(1^n; m_1)$, we know that $\mathsf{Enc}'(pk, m_b; \rho) = \mathsf{Enc}(pk, m_b; \rho)$ for every ρ.

Problem 4. Upgrading Lamport Signatures. (8 points)

Recall Lamport's signature scheme from class, based on a OWF $f : \{0,1\}^n \to \{0,1\}^\ell$, that produces an n^2-bit signature for an n-bit message:

$\mathsf{Gen}(1^n)$

1: $x_{1,0}, \ldots, x_{n,0} \leftarrow \{0,1\}^n$
2: $x_{1,1}, \ldots, x_{n,1} \leftarrow \{0,1\}^n$
3: $sk := (x_{1,0}, \ldots, x_{n,0}, x_{1,1}, \ldots, x_{n,1})$
4: $vk := (y_{1,0}, \ldots, y_{n,0}, y_{1,1}, \ldots, y_{n,1})$, where $y_{i,c} = f(x_{i,c})$
5: **return** (sk, vk)

$\mathsf{Sign}(sk, m \in \{0,1\}^n)$

1: **parse** $sk = (x_{1,0}, \ldots, x_{n,0}, x_{1,1}, \ldots, x_{n,1})$
2: **return** $\sigma := (x_{1,m_1}, \ldots, x_{n,m_n})$

$\mathsf{Ver}(vk, m \in \{0,1\}^n, \sigma)$

1: **parse** $\sigma := (\sigma_1, \ldots, \sigma_n)$
2: **if** $\forall i \in [n], f(\sigma_i) \stackrel{?}{=} y_{i,m_i}$: **return** 1
3: **else** : **return** 0

In this problem, we will look at a stronger definition of one-time unforgeability known as *one-time strong unforgeability* which states that not only is the adversary unable to produce a signature on a different message, but also that she is unable to produce a *different* signature σ^* on the same message it requested a signature on.

Definition 3 (One-time strong unforgeability)
Let $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ *be a digital signature scheme with message space* $\mathcal{M}$ *and key space* $\mathcal{K}$ *with security parameter* n*. This scheme is one-time strongly unforgeable if for all pair of PPT algorithms* $(\mathcal{A}_1, \mathcal{A}_2)$*, there exists a negligible function* negl *such that for all* n*,*

$$\Pr\left[\begin{array}{l|ll} (sk, vk) \leftarrow \mathsf{Gen}(1^n); & & \\ (m, \mathsf{state}) \leftarrow \mathcal{A}_1(vk); & : & (m^*, \sigma^*) \neq (m, \sigma) \wedge \\ \sigma \leftarrow \mathsf{Sign}(sk, m); & & \mathsf{Ver}(vk, m^*, \sigma^*) = 1 \\ (m^*, \sigma^*) \leftarrow \mathcal{A}_2(\sigma, \mathsf{state}) & & \end{array}\right] \leq \mathsf{negl}(n).$$

(a) (3 points) Show an attack on the one-time strong unforgeability of Lamport's scheme. That is, assuming the existence of length-preserving one-way functions, construct a one-way function f such that the Lamport signature scheme using f is *not* one-time strongly unforgeable.

Solution

Let $f : \{0,1\}^{n-1} \to \{0,1\}^{n-1}$ be an arbitrary length-preserving one-way function. Define $f' : \{0,1\}^n \to \{0,1\}^{n-1}$ by $f'(x) = f(x_{[1:n-1]})$. f' is also a OWF: given a preimage x' for $f'(x)$, $x'_{[1:n-1]}$ is a preimage for $f(x_{[1:n-1]})$, which is a correctly distributed challenge in the OWF game for f.

Lamport signatures with f' are clearly not one-time strongly unforgeable: Given a signature σ for m, the adversary can (for example) produce $\sigma_{[1:n^2-1]} \,||\, \overline{\sigma_{n^2}}$, a valid and different signature for m.

(b) (2 points) Prove that if f is an injective one-way function, then Lamport's scheme is one-time strongly unforgeable.

Solution

The proof of one-time unforgeability is directly from class. For there to be any difference between one-time strong unforgeability and one-time unforgeability, there must exist distinct

signatures for the same message m and verification key vk. However, since f is injective, there is a unique signature that will pass verification for any given message m and verification key vk.

(c) (3 points) Show that if one-way functions exist, then there exists a one-way function such that Lamport's scheme is one-time strongly unforgeable. Note that building injective one-way functions from arbitrary one-way functions is an open question. (You may use statements from class or earlier in this problem set.)

Solution

We will use the fact from earlier in this problem set that if one-way functions exist, then targeted collision resistant hash functions exist. Then, we will apply one of the previous problems to (provably) get a one-way function of the form $F(k, x) = (k, H_k(x))$, where H_k is sampled from a targeted collision resistant hash function family $\mathcal{H}$.

We use this F as the one-way function. The proof of one-time unforgeability is directly from class. Suppose there is an adversary pair (A_1, A_2) that breaks one-time *strong* unforgeability but not one-time unforgeability. We will define A to be an adversary for targeted collision resistance as follows. On input (random) (k, x_1), uniformly randomly guess some index $i \in [n]$ and $b \in \{0, 1\}$ for which the pair (A_1, A_2) will forge on. Then, in the (i, b) location of sk and vk, put (k, x_1) and $F(k, x_1) = (k, H_k(x_1))$ respectively, and for all other locations of vk, sample i.i.d. k' and x' and put (k', v') and $(k', H_{k'}(x'))$ in the corresponding locations of sk and vk. Then, run A_1, Sign, and A_2 to get some $\sigma^* \neq \sigma$ that both verify for the same message m (assuming the adversary pair (A_1, A_2) succeeds, which holds with probability $1/p(n)$ for some polynomial $p(n)$ infinitely often). With probability at least $1/(2n)$ independently of (A_1, A_2), A has correctly guessed a location in which $\sigma \neq \sigma^*$ which both verify for the same message. By definition of F and the guess, this implies $H_k(x_1) = H_k(x_2)$ for some $x_1 \neq x_2$. Outputting x_2 (which is contained within σ^*) breaks weak collision resistance, with probability $1/p(n) \cdot 1/(2n)$, which is non-negligible.