Please list your collaborators and provide any references that you may have used in your solutions.

**Question 1**

Consider the following encryption algorithm based on the shortLWE assumption. The secret key is a shortLWE secret \( x \sim \nu^n \) and the public key is \( PK = (A, Ax + e) \), where \( A \) is a random \( n \times n \) matrix over \( \mathbb{Z}_q \) and \( e \sim \nu^n \). The encryption of a message represented by \( M \in \mathbb{Z}_q \) under public key \( PK = (A, b) \) is

\[
Enc(PK, M) = (e' + x'A, e'' + x'b + M), \quad x' \sim \nu^n, e' \sim \nu^n, e'' \sim \nu.
\]

(A is a matrix, \( x, e, b \) are column vectors, \( x', e' \) are row vectors, and \( e'', M \) are scalars.)

(a) Give the corresponding decryption algorithm. Show that the scheme is functional assuming that the message is encoded in the \( \log q - \log n - 2 \log b - O(1) \) most significant bits of \( M \).

(b) Prove that the scheme is \((s', \varepsilon')\)-message simulatable under the \((s, \varepsilon)\)-shortLWE assumption.

(Calculate the dependence of \( s' \) and \( \varepsilon' \) on \( s, \varepsilon \), and other relevant parameters.)

**Question 2**

In this question you will analyze the following LWE-based public-key identification protocol. The secret key is a random \( x \sim \{-1, 1\}^m \). The public key is \((A, xA)\) where \( A \) is a random \( m \times n \) matrix over \( \mathbb{Z}_q \). All arithmetic is modulo \( q \).

1. Prover chooses a random \( r \sim \{-b, \ldots, b\}^m \) and sends \( rA \).

2. Verifier sends a random bit \( c \sim \{0, 1\} \).

3. Prover sends \( r + cx \).

(a) Show that if \( m = 1 \) then conditioned on \(|r + x| \leq b - 1\), \( r \) and \( r + x \) are identically distributed.

(b) Now let \( m \) be arbitrary as in the protocol. Show that \( r \) and \( r + x \) are \( O(m/b) \)-statistically close.

(c) Show that the view of an eavesdropper who sees \( q' \) protocol transcripts is \( O(q'n/b) \)-statistically close to some random variable that can be efficiently sampled by a simulator that is given only the public key.

(d) Let \( h_A(x) = xA \), where the entries of \( x \) are of magnitude at most \( 2(b+1) \). Show that if \( h \) is a collision-resistant hash function then no efficient cheating prover can handle both challenges \( c = 0 \) and \( c = 1 \). Conclude that, if repeated sufficiently many times, the protocol is secure against eavesdropping.

(Work out the dependences between the security parameters.)

(e) (Optional) Prove that the protocol is secure against impersonation.
Question 3

In this question you will show that using an obfuscator, an adversary can plant a collision in a hash function that makes it insecure against him, but secure against everyone else. Let $h: \{0,1\}^m \rightarrow \{0,1\}^n$ be a collision-resistant hash, $Obf$ an obfuscator, and $A$ the following algorithm:

1. Sample a random key $K$ and a random input $\hat{x} \sim \{0,1\}^m \setminus \{0\}$.

2. Construct a circuit $h'$ that implements the function

$$h'(x) = \begin{cases} h_K(0), & \text{if } x = \hat{x}, \\ h_K(x), & \text{if not.} \end{cases}$$

3. Output $H = Obf(h')$.

Then $A$ knows a collision for $H$, namely the pair $(0, \hat{x})$. We can view $H$ both as a random key and the function described by it, so $(s, \varepsilon)$-collision-resistance means that the probability that $C(H)$ outputs a collision for $H$ is at most $\varepsilon$ for every $C$ of size at most $s$.

(a) Show that the views $D^{h_K}$ and $D^{h'}$ are $q/(2^m - 1)$-statistically close for any distinguisher $D$ that makes at most $q$ queries to its oracle.

(b) Show that if $h$ is $(s, \varepsilon)$-collision resistant and $Obf$ is $(s + 2t + O(n), \varepsilon')$-VBB secure, $H$ is $(s - tt', \varepsilon + \varepsilon' + q/(2^m - 1))$-collision resistant, where $t$ and $t'$ are the sizes $h$ and the VBB simulator, respectively.

(c) Show that the MAC from Theorem 5 in Lecture 6 is insecure against a forger that knows $\hat{x}$.
**Question 4**

Bob has some database $D$ that Alice wants to query, but she suspects that Bob might not give her correct answers. To ensure integrity Alice also has a short collision-resistant hash $h(D)$ of the database. When Alice wants to retrieve the contents $D(r)$ of database row $r$, Bob sends Alice the whole database $D$ and she can verify that the hash is correct. This is impractical when the database is large. In this problem you will model this scenario cryptographically and explore a more efficient solution based on Merkle trees.

A database is a function $D: \{1, \ldots, R\} \rightarrow \{0,1\}^n$ that maps a row $x$ to a data item $D(x)$. A **database commitment protocol** has the following format. Alice has no input and Bob’s input is the database $D$. In the setup phase, Bob sends Alice a commitment $\text{com}$ to the database. In the query phase,

1. Alice sends a query $x \in \{1, \ldots, R\}$ of her choice to Bob.
2. Bob returns an answer $y = D(x)$ and a certificate $\text{cert}$.
3. Upon receiving $y$ and $\text{cert}$, Alice runs a verification which accepts or rejects.

The functionality requirement is that when Bob is honest Alice accepts with probability 1.

(a) Give a definition of $(s, \varepsilon)$-security. The adversary is a cheating Bob.\(^1\) You may assume the availability of a random public key $K$ available to all the parties (as in the collision-resistant hash setup).

(b) Let $\text{com} = h_K(D)$ and $\text{cert} = D$ where $h$ is a collision-resistant hash function. Describe the verification and prove that the protocol is secure.

(c) The certificate in part (b) is $nR$-bits long. Now assume $h$ is the Merkle tree-based collision resistant hash of depth $\log R$ from Lecture 6. Describe a different certificate of length $n(\log R + 1)$, the corresponding verification, and prove that the protocol is secure.

**Hint:** It is sufficient for Bob to reveal the hashes at $\log R + 1$ nodes in the Merkle tree.

\(^1\)There is no need for a “learning phase” as there is no secret information to be learned.