End Semester - Fall, 2025
M. Tech. Cryptology & Security - 2^{nd} Year
Advanced Cryptology

November 28, 2025

Maximum Marks: 50
Maximum Time: 3 hr
Open note/book exam

Instructions

- Total marks provided in the paper: 70.
- Be short and precise. Partial marking will be provided for a partially correct answer/attempt.
- All parts of the same questions must be done at the same place, and in order. Marks will be deducted for violating this – this will be followed strictly.
- Verbosity (e.g. unnecessary/irrelevant sentences) is highly discouraged, and may lead to deduction of marks, if found incorrect/misleading.
- The seating arrangements and other regulations circulated from Dean's office must be followed adequately.

Questions

1. Recall the BLS signature proof: if CDH holds across the pairing groups $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, then BLS is existentially unforgeable. The reduction B will try to solve CDH using the signing adversary A. Also recall that CDH assumption in this setting states that given g_1^x and g_2^y for uniform random x, y, it is hard to find g_1^{xy}. Assume that the isomorpshism $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ is easy to compute (Type-2).
   (a) In the first step, B obtains g_1^x and g_2^y from the CDH challenger. What does B do in the next step? (hint: recall what A expects in the beginning of the unforgeability game)
   (b) B then receives two types of queries from A, one is a set of random oracle queries, and another is a bunch of signing queries. Elaborate how these queries are handled by B. Here, assume that if A makes a signing query on M, then it must have made a random oracle query on M beforehand. Also assume that Q is the maximum number of random oracle queries A may ask. Q can be set to very large, but still a polynomial in the security parameter.
   (c) Once A returns a forgery, what does B do?
   (d) If A can forge with probability ε, then what is probability with which B can break CDH. Provide arguments in support of this, possibly with a probability calculation.
   (e) Where does the reduction use the easy isomorphism?
   (f) Where does the reduction use the fact that the forgery message M^* can not be asked as a signing query?

$(2 + 6 + 4 + 4 + 2 + 2 = 20)$

2. Recall the Exponentiated El-Gamal Encryption Scheme. Consider that as a commitment scheme.

   (a) Prove that it is unconditionally binding.

   (b) Prove that it is computationally hiding.

   (c) How can you delete a part of the ciphertext so that the scheme becomes unconditionally hiding and computationally binding?

   (d) For the changed scheme, prove both the hiding and binding the properties?

   (4+4+2+8 = 18)

3. Recall the definition of a Verifiable Random Function (VRF). A VRF is a PRF, but with a public verifiability property. Typically a VRF scheme consists of three algorithms $(\mathsf{KGEN}, \mathsf{Eval}, \mathsf{Verify})$, such that $\mathsf{KGEN}(1^\kappa)$ outputs a key-pair $(\mathsf{sk}, \mathsf{vk})$; $\mathsf{Eval}(x)$ outputs (y, π), where y is pseudorandom and $\mathsf{Verify}(\mathsf{vk}, x, y, \pi)$ outputs 1 if and only if (x, y) are the correct input/output pair. VRF correctness and verification are similar to that of a signature; pseudorandomness of y is similar to a PRF. However, VRF additionally has a uniqueness property, which ensures that for each x, there is a single y for which there exists a π such that $\mathsf{Verify}(\mathsf{vk}, x, y, \pi)$ outputs 1. Suppose you have a signature scheme. Then:

   (a) Which property you'd need to convert that signature scheme to a to a VRF?

   (b) If the required property is met, how would you construct a VRF out of that? (you can also use a hash function)

   (c) Prove that the final scheme is a VRF, assuming the underlying properties of the signature scheme.

   (2+4+6 = 12)

4. Again, recall the exponentiated El-Gamal Encryption scheme with secret key $\mathsf{sk} \in \mathbb{Z}_p$ and public key g^{sk}. Let $s \in \mathbb{Z}_p$ is the secret shared using a (t, n)-Shamir sharing to generated n shares $s_1, \ldots, s_n$ each in $\mathbb{Z}_p$. Now, a prover correctly encrypts s and all $s_1, \ldots, s_n$ using the Exp-El-Gamal to produce ciphertexts of the form:

   - $c := (g^r, \mathsf{pk}^r g^s)$
   - $c_1 := (g^{r_1}, \mathsf{pk}^{r_1} g^{s_1})$
   - $\cdots$
   - $c_n := (g^{r_n}, \mathsf{pk}^{r_n} g^{s_n})$

   A verifier is given all ciphertexts and the public key. Now, design a HVZK interactive sigma protocol by which the prover can prove that the ciphertexts $(c_1, \ldots, c_n)$ are indeed a (t, n) Shamir's sharing of the plaintext encrypted in the first ciphertext c. (8)

5. Recall the GSW fully homomorphic encryption scheme.

   (a) Prove that the encryption scheme is CPA-secure.

   (b) Prove that the encryption scheme is fully homomorphic.

   (6+6)