1. (a) (10 points) Give the definition of the RSA assumption.

   (b) (10 points) Construct a CPA-secure public-key encryption scheme under the RSA assumption. (giving the security proof is not necessary)
   Does your construction have CCA security? (in case of being secure, giving the security proof is not necessary)

2. (a) (10 points) Give the formal definition of DDH assumption and provide an example of a group which is generally believed to hold this assumption.

   (b) (10 points) Construct a CPA-secure encryption scheme under the DDH assumption. (giving the security proof is not necessary)
   Does your construction have CCA security? (in case of being secure, giving the security proof is not necessary)

Now choose one of the following problems: (Your last problem will be worth 30 points)

1. **(CIH from RSA)** Let $N = pq$ be an RSA modulus and take $e \in \mathbb{N}$ to be a prime that is also relatively prime to $\phi(N)$. Let $u \leftarrow_{\$} \mathbb{Z}_N^*$, and define the hash function

   $$\mathsf{H}_{N,e,u} : \mathbb{Z}_N \times \{0, ..., e-1\} \to \mathbb{Z}_N \quad \text{where} \quad \mathsf{H}_{N,e,u}(x, y) = x^e u^y \in \mathbb{Z}_N$$

   We want to show that under RSA assumption, $\mathsf{H}_{N,e,u}$ defined above is collision-resistant. Namely, suppose there is an efficient adversary $\mathcal{A}$ that takes as input (N, e, u) and outputs $(x_1, y_1) \neq (x_2, y_2)$ such that $\mathsf{H}_{N,e,u}(x_1, y_1) = \mathsf{H}_{N,e,u}(x_2, y_2)$. We use $\mathcal{A}$ to construct an efficient adversary $\mathcal{B}$ that takes as input (N, e, u) where $u \leftarrow_{\$} \mathbb{Z}_N^*$ and outputs x such that $x^e = u \in \mathbb{Z}_N$.

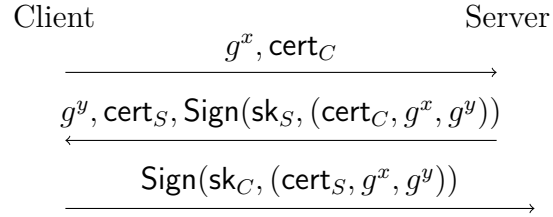
   (a) (15 points) Show that using algorithm $\mathcal{A}$ defined above, algorithm $\mathcal{B}$ can efficiently compute $a \in \mathbb{Z}_N$ and $b \in \mathbb{Z}$ such that $a^e = u^b \pmod N$ and $0 \neq |b| < e$. Remember to argue why any inverse you compute will exist (or alternatively, if they do not exist, then $\mathcal{B}$ can directly break RSA).

   (b) (5 points) Use the above relation to show how $\mathcal{B}$ can efficiently compute $x \in \mathbb{Z}_N$ such that $x^e = u$.

Hint: Since $|b| < e$ and e is prime, $gcd(b, e) = 1$. Now, apply Bezout's identity. Note that $\mathcal{B}$ does not know the factorization of N, so it is not able to compute $b^{-1} \pmod{\phi(N)}$.
Note: By Bezout's identity, if $gcd(b, e) = 1$, then there exists integers $s, t \in \mathbb{Z}$ such that $bs + et = 1$.

(c) (10 points) Show that if we extend the domain of $\mathsf{H}_{N,e,u}$ to $\mathbb{Z}_N \times \{0, ..., e\}$, then the function is no longer collision-resistant.

2. **(Authenticated Key Exchange)** Consider the following protocol for authenticated key exchange (AKE) with mutual (i.e., two-sided) authentication. Both the client and the server have a public/private key-pair $(\mathsf{vk}_C, \mathsf{sk}_C)$ and $(\mathsf{vk}_S, \mathsf{sk}_S)$ for digital signature scheme, respectively. They also have certificates cert_C and cert_S that authenticate vk_C and vk_S, respectively. The AKE protocol operates over a group $\mathbb{G}$ of prime order p and generator g. The client samples a fresh $x \leftarrow_\$ \mathbb{Z}_p$ and the server samples a fresh $y \leftarrow_\$ \mathbb{Z}_p$ in each invocation of the protocol: In the second step, the client validates the signature with respect to



the verification key contained in cert_S before computing its third message. At the end of the protocol, if all of the signatures verify (with respect to the verification keys identified by the certificates), the client and server computes the shared key as $k \leftarrow \mathsf{H}(g, g^x, g^y, g^{xy})$. Moreover, the client outputs the party identified by cert_S as its peer in the connection and the server outputs the party identified by cert_C as its peer. Throughout this problem, you should consider an active network adversary that is allowed to register a certificate of its own (i.e., the adversary has a certificate cert_A for its identity A, which is different from both the client's identity C associated with cert_C and the server's identity S associated with cert_S).

(a) (10 points) Suppose the server does not sign cert_C in its reply on the client. Namely, the server computes $\mathsf{Sign}(\mathsf{sk}_S, (g^x, g^y))$ instead

of
$\mathsf{Sign}(\mathsf{sk}_S, (\mathsf{cert}_C, g^x, g^y))$. Show that there is an identity misbinding attack on this protocol.
Note: In *identity misbinding attacks* against authenticated key-exchange protocols, a legitimate but compromised participant manipulates the honest parties so that the victim becomes unknowingly associated with a third party.

(b) (10 points) Suppose the client only signed the server's certificate and not the Diffie-Hellman shares in the final message. Namely, the client computes $\mathsf{Sign}(\mathsf{sk}_C, \mathsf{cert}_S)$ instead of $\mathsf{Sign}(\mathsf{sk}_C, (\mathsf{cert}_S, g^x, g^y))$. Show that an adversary is able to establish a session with the server such that the adversary knows the shared key k, but the server thinks it is communicating with the party identified by cert_C (i.e., the client).
Hint: Remember that an active network adversary is allowed to observe (and tamper with) multiple interactions between the client and the server.

(c) (10 points) Suppose that the client signed its Diffie-Hellman share in its first message, and dropped the third message entirely. Namely, the client's first message is now $(g^x, \mathsf{cert}_C, \mathsf{Sign}(\mathsf{sk}_C, g^x))$ and the overall protocol now completes in two rounds. Show that there is an identity misbinding attack on this protocol.

(d)* (10 points) Suppose that instead of signing the pair (g^x, g^y), the client and the server instead signed g^{xy}. Explain why this is a bad idea.

3. **(Encrypting Twice)** Let $(\mathsf{Enc}, \mathsf{Dec})$ be a symmetric authenticated encryption scheme with key-space $\mathcal{K} = \{0,1\}^\lambda$. Consider the encrypt-twice cipher $(\mathsf{Enc}_2, \mathsf{Dec}_2)$ with independent keys where $\mathsf{Enc}_2((k_1, k_2), m) := \mathsf{Enc}(k_2, \mathsf{Enc}(k_1, m))$ and

$$\mathsf{Dec}_2((k_1, k_2), c) := \begin{cases} \mathsf{Dec}(k_1, \mathsf{Dec}(k_2, c)) & \mathsf{Dec}(k_2, c) \neq \bot \\ \bot & \textit{otherwise} \end{cases}$$

(a) (15 points) Show that $(\mathsf{Enc}_2, \mathsf{Dec}_2)$ is still an authenticated encryption scheme even if the adversary learns k_1 (but has no information about $k_2 \leftarrow\!\!\$\, \mathcal{K}$). Remember to show both CPA-security and

ciphertext integrity. To model knowledge of k_1, you can assume that the adversary is given k_1 at the beginning of the CPA-security and ciphertext integrity experiments.

(b) (15 points) Show that $(\mathsf{Enc}_2, \mathsf{Dec}_2)$ is no longer an authenticated encryption scheme if the adversary learns k_2 (but has no information about $k_1 \xleftarrow{\$} \mathcal{K}$). To model knowledge of k_2, you can assume that the adversary is given k_2 at the beginning of the CPA-security and ciphertext integrity experiments.