35

CSCI 5440: Theory of Cryptography **2020-2021 Term 1**

Assignment 2

Student: Yu Zheng Student ID: 1155113945

Acknowledge: Thank William H. Y. Mui for discussion.

Question 1

- Proof: Since R and R' are independent, there exist the cases that $R = R'$. In this case, both 0 and 1 are encrypted to get (g^R, h^R), which may result in wrong decryption. That is, the one who decrypts the ciphertext will get the message 0 in above both cases.
- When encrypting 0, the output is (g^R, g^{xR}); When encrypting 1, the output is $(g^R, g^{xR'})$. The probability of $R = R'$ is $1/q$, resulting in errors with $\Omega(1/q)$. Except the error cases, the decryption is to compare $(g^R)^x$ with the second part of the encryption. If $(g^R)^x$ is equal to the second part, the message is 0; otherwise the message is 1. So this scheme succeeds with probability $1 - \Omega(1/q)$.

9

THIS 1/q IS SUBSUME BY THE DDH ASSUMPTION

- Proof: Let us prove the statement by contradiction. Suppose the encryption is not message indistinguishable. This implies that there exists an adversary that has some advantage to distinguish the encryptions h^R and $h^{R'}$ with only knowing the h, g^R. This means that the adversary has some advantage to solve DDH, which contradicts the assumptions. Therefore, we proved that the encryption is message indistinguishable. The probability of successful guess is equal to that of random guessing, *i.e.*, $1/q$ for choosing the right secret key. Notably, when $R = R'$, the adversary can always guess that the message is 0. Suppose we fix this problem and get a better encryption scheme by choosing distinct R, R' (This does not influence much actually). The adversary's advantage is $\dfrac{1/q}{1 - \Omega(1/q)} = \dfrac{1}{q}$. The size is also s since one only needs to attack one DDH, *i.e.* $(g^R)^x$ to know x.

Question 2

- Proof: If $m = 1$, then $x \sim \{-1, 1\}$ and $r \sim \{-b, \ldots, b\}$. For $|r| \leq b - 1$, r is randomly chosen from $\{-b + 1, \ldots, b - 1\}$. Conditioned on $|r + x| \leq b - 1$, $r + x$ is randomly chosen from $\{-b + 1, \ldots, b - 1\}$ for $x = 1$; Similarly, $r + x$ is randomly chosen from $\{-b + 1, \ldots, b - 1\}$ for $x = -1$. Thus, $r + x$ is randomly chosen from $\{-b + 1, \ldots, b - 1\}$ for all x, which is identically distributed to r.

6

- Proof: Let's start with m=1. By Q2(a), we know that r conditioned on $|r| \leq b - 1$ is identically distributed to $r+x$ conditioned on $|r+x| \leq b-1$. For $|r| = b$, $r+x$ and r can be distinguished since $r \sim \{-b, b\}$ and $r+x \sim \{-b-1, -b+1, b-1, b+1\}$. For $|r+x| = b$, $r+x$ and r can be distinguished, similarly. Then we know r and $r + x$ are $O(1/b)$-statistically close. Next, we consider the cases $m \geq 2$. We denote the stings $r = r_1 r_2, \ldots, r_m$ and

$x = x_1x_2, \ldots, x_m$, where r_i and x_i are the i-th bit. For each $1 \leq i \leq m$, r_i and $r_i + x_i$ are $O(1/b)$-statistically close. Additionally, r_i and r_j are chosen independently, for $i \neq j$. Therefore, for all $1 \leq i \leq m$, r and $r + x$ are $O(m/b)$-statistically close.

THE PK SHOULD BE INCLUDED IN THE VIEW.

- Proof: Suppose that some prover P^* wins the eavesdropping game with the probability ϵ^*. When the verifier sends $c = 0$, P^* should send r; When the verifier sends $c = 1$, P^* should send $r + x$; By Q2(b), we know that r an $r + x$ are $O(m/b)$-statistically. This implies a simulator can efficiently sample a random variable with probability $O(m/b)$ for each protocol transcript. In the learning phase, an eavesdropper sees q' protocol transcripts. Since the verifier issues independent challenges, the simulator can efficiently sample a random variable with probability $O(q' \cdot m/b)$. Thus, we proved the statement.

WHAT IS THE COLLISION?

- Proof: A Cheating prover P^* can choose any random number r. Since $H_A(x) = xA$ is a collision-resistant hash function, so the prover P^* can not find a collision x' by only knowing public key. (If P^* can, the collision-resistant hash breaks.) If the verifier sends the random bit $c = 0$, the prover P^*can just send r to verifier. If the verifier send the random bit $c = 1$, the prover P^* needs to generate a fake $r + x$. Since r and $r + x$ are $O(m/b)$-statistically (By Q2(b)) for x with one entry, prover P^* succeeds to generate fake $r + x$ with the probability $O((m/b)^{\Omega(b)})$ for $\Omega(b)$ entries. However, this probability is negligible. Conversely, if P^* generates a r and use it as $r + x$, the analysis is same as the above. Thus, we proved that no efficient cheating prover can handle both challenges for a collision-resistant hash H.

  Conclusion: From above, we can see that the cheating prover can pass the validation with probability $O(1/2 + (m/b)^{\Omega(b)})$. The chances that he pass the repeated t times should be $O((1/2+(m/b)^{\Omega(b)})^t)$. This is the case for eavesdropping attacks as long as the repetitions are carried our sequentially. Therefore, we concluded security against eavesdropping.

-

Question 3

- Insecure.

  Proof: An adversary can first learn $Tag(K, M_0M_1)$ and $Tag(K, M_1M_2)$ to know $(F_K(M_0, 0), F_K(M_1, 1))$ and $(F_K(M_1, 0), F_K(M_2, 1))$. Then, he can generate a forgery,

$$Tag(K, M_0M_2) = (F_K(M_0, 0), F_K(M_2, 1))$$

10

- Insecure.

  $Tag(K, M_0M_1) + Tag(K, M_1M_0) = F_K(M_0, 0) + F_K(M_1, 1) + F_K(M_1, 0) + F_K(M_0, 1) = (F_K(M_0, 0) + F_K(M_0, 1)) + (F_K(M_1, 0) + F_K(M_1, 1)) = Tag(K, M_0M_0) + Tag(K, M_1M_1)$. Thus, an attacker can find the forgery by using the above relation.

Question 4

- Proof: Let's start with $q = 1$. For $\{0,1\}^m$, it holds 2^m possibilities. Since an adversary plants a collision, there are $2^m - 1$ possibilities for h'. The distance between h' and h_K is $1/(2^m - 1)$ by considering the collision pair $(0, x')$. For a distinguisher D who makes q queries, D can succeed to distinguish the h' and h_K with probability at least $q/(2^m - 1)$. Therefore, for $q/(2^m - 1)$-statistically close, a distinguisher D can make at most q queries.

- Proof: Suppose H is (s^*, ϵ^*)-collision resistant. The probability the simulator can outputs a collision for H is at most ϵ^* for every circuit at most s^*. For h with a specific key, the simulator with output of Obf is at most size $s^* + t'$. For all keys of h, the simulator for H has at most size $s^* + t't = s$. Thus, $s^* = s - t't$. By the Q4(a), we know D^h and $D^{h'}$ are $q/(2^m - 1)$-statistically close, so h' is $\epsilon + q/(2^m - 1)$-statistically close. One may find the collision from the statistically distance, where the same output is from the different inputs H generated by h' and Obf (with probability ϵ') is $(\epsilon + \epsilon' + q/(2^m - 1))$-collision resistant. Therefore, we proved the statement.

10

- Proof: If a forger knows $\dot{x}$, he can ask the query $x = 0$ and get the tag T in the learning phase. Later, the forger just generates a fake $Tag'(K, h'(\dot{x})) = Tag'(K, h'(0))$ which has been learned before. The forger then can pass the verification $Ver'(K, h'(\dot{x}), T)$ with the probability 1. In this process, the forger does not need to know the key, and can succeed to pass verification for any key. Therefore, we proved the statement.