Advanced Cryptography, Problem Set 1

Allison Bishop (no collaborators)

Problem 1 Consider the hash function \(H: \mathbb{Z}_p \times \mathbb{Z}_p \rightarrow G\) that takes two elements of \(\mathbb{Z}_p\) and hashes it to one group element of order \(p\). Assume that the hash function is defined by \(u^{x_1} \cdot v^{x_2}\), where \(u, v\) are chosen by a trusted setup authority. Show that if there exists an attacker that can find a collision, then the attacker (algorithm) can be used to break the discrete log problem in the same group.

Solution. We suppose that we are given \(g, g^a \in G\), and our task is to find \(a \in \mathbb{Z}_p\). We first check if \(g^a\) is the identity element. If so, we know that \(a = 0\). Otherwise, we set \(u = g^a\) and \(v = g^b\), where we choose \(b \in \mathbb{Z}_p\) randomly (though we make sure that \(b \neq 0\)). We then call our hash attacker to find distinct pairs \((x_1, y_1), (x_2, y_2) \in \mathbb{Z}_p \times \mathbb{Z}_p\) such that \(u^{x_1} \cdot v^{y_1} = u^{x_2} \cdot v^{y_2}\). We can rewrite this as: \(u^{x_1-x_2} = v^{y_2-y_1}\) or \(g^{a(x_1-x_2)} = g^{b(y_2-y_1)}\). Since the pairs \((x_1, y_1)\) and \((x_2, y_2)\) are distinct, at least one of \(x_1 - x_2\) and \(y_2 - y_1\) must be nonzero. Since \(a \neq 0\) and \(b \neq 0\), both of \(x_1 - x_2\) and \(y_2 - y_1\) must be nonzero. From \(g^{a(x_1-x_2)} = g^{b(y_2-y_1)}\), we can conclude that:

\[
a(x_1 - x_2) \equiv b(y_2 - y_1)(\text{mod } p).
\]

Since \(x_1 - x_2\) is not congruent to 0 modulo \(p\), we can divide by it on both sides to achieve:

\[
a \equiv b(y_2 - y_1)(\text{mod } p).
\]

Thus, we have found \(a\) and solved the discrete log problem in \(G\).

We note that if our hash attacker only succeeds in finding a collision with non-negligible probability \(\epsilon\), then our discrete log attack also succeeds with this probability, since it succeeds whenever it is given a collision. \(\Box\)
Problem 2 Prove that the security of El Gamal encryption is equivalent to the DDH assumption by showing that an attacker on DDH can break El Gamal. What does this say about the security of El Gamal in groups where there exists efficiently computable bilinear maps?

Solution. We assume that algorithm $A$ is a DDH attacker: given $g, g^a, g^b, T$ where $T = g^{ab}$ when $\gamma = 0$ and is randomly chosen otherwise (whenever we say “randomly chosen,” we mean uniformly), $A$ can guess $\gamma$ with probability $\frac{1}{2} + \epsilon$. We assume that $\epsilon$ is non-negligible. Using $A$, we will construct an El Gamal attacker algorithm $B$.

The El Gamal challenger gives $B$ input $g, Y = g^a$. $B$ then randomly chooses $M_0, M_1 \in G$. It sends these to the El Gamal challenger, who randomly sets $\gamma \in \{0, 1\}$ and sends back $T = Enc(M_\gamma)$ to $B$. By definition of El Gamal encryption, $Enc(M_\gamma)$ is a pair $C_1 = g^r$ and $C_2 = M_\gamma Y^r = M_\gamma g^{ar}$. Now, $B$ chooses $\gamma'$ randomly from $\{0, 1\}$ and computes $T = C_2/M_{\gamma'}$. It then sends $A$ the following input: $(g, g^a, g^r, T)$. $A$ returns $\gamma''$. If $\gamma'' = 0$, then $A$ is guessing that $T = g^{ar}$, and $B$ guesses that $\gamma = \gamma'$. If $\gamma'' = 1$, then $B$ guesses the opposite of $\gamma'$.

We now analyze the probability that $B$ guesses $\gamma$ correctly. We note that $\gamma = \gamma'$ with probability $\frac{1}{2}$. If $\gamma = \gamma'$ (i.e. $\gamma \oplus \gamma' = 0$ where $\oplus$ denotes addition modulo 2), then $T = g^{ar}$. If $\gamma \oplus \gamma' = 1$, then $T = g^{ar} M_1/M_0$ or $g^{ar} M_0/M_1$. Since $M_0, M_1$ are chosen randomly, their quotient is a random element of $G$, and so $T$ is random from $A$’s perspective (note that $A$ has no knowledge of $M_0$ or $M_1$). This is exactly the setup of the DDH problem. Hence, by definition, $A$ will guess $\gamma''$ which will be equal to $\gamma \oplus \gamma'$ with probability $\frac{1}{2} + \epsilon$, giving $B$ the same probability of correctly guessing $\gamma$.

In groups were there exist efficiently computable bilinear maps, the DDH assumption is false. Given an efficiently computable bilinear map $e$, one can construct a DDH attacker as follows: given $g, g^a, g^b, T$, compute $e(g, T) = e(g^a, g^b)$. If they are equal, guess that $\gamma = 0$. Otherwise, guess that $\gamma = 1$. This will always succeed in correctly guessing $\gamma$, since for $T = g^r$, $e(g, T) = e(g, g)^r$ will equal $e(g^a, g^b) = e(g, g)^{ab}$ if and only if $r = ab$ in $\mathbb{Z}_p$.

As we have shown above, this DDH attacker can break El Gamal, so El Gamal is not secure in such groups. $\square$
Problem 3 The decisional linear problem is that given a group $G$ of prime order $p$ and elements $g, u, v, g^a, u^b$, distinguish whether $T = v^{a+b}$ or is random. (A uniformly random bit $\gamma \in \{0,1\}$ is flipped to determine $T$.) The assumption is that no (polynomial-time) attacker should be able to guess $\gamma$ with more than $\frac{1}{2} + \epsilon$ negligible advantage. Create a public key encryption system that is CPA secure under this assumption. First, describe three algorithms: Setup, Encrypt, and Decrypt. Then show that if the decisional linear assumption holds, your system is semantically secure under CPA attacks.

Solution. Our Setup algorithm fixes a group $G$ of order $p$, a generator $g$, and $c,d \in \mathbb{Z}_p$ chosen randomly. It publishes $g, u = g^c, v = g^d$ as the public key and keeps $c,d$ as the secret key. In the Encryption algorithm, the encrypter chooses $a, b \in \mathbb{Z}_p$ randomly. When the message is $M \in G$, the cipher text produced is: $C_1 = g^a, C_2 = u^b, C_3 = M \cdot v^{a+b}$. The Decryption algorithm uses knowledge of $c$ to compute $C_2^{1/c} = (u^b)^{1/c} = g^b$, and hence can then compute $C_1 \cdot C_2^{1/c} = g^a \cdot g^b = g^{a+b}$. Then, $d$ can be used to compute $(g^{a+b})^d = v^{a+b}$. From here, $M$ can be computed as $C_3 / v^{a+b}$.

We now show that if $A$ is a CPA attacker for this system, we can build an algorithm $B$ which breaks the decisional linear assumption. This will show that if the decisional linear assumption holds, then our system is semantically secure under CPA attacks. The decisional linear challenger (denoted DL-challenger) starts by giving $B$ the parameters $g, u, v, g^a, u^b, T$, where $T = v^{a+b}$ if $\gamma = 0$ and is randomly chosen otherwise. $B$ then gives $A$ the public key $g, u, v$. $A$ generates two messages $M_0$ and $M_1$ and sends them back to $B$. $B$ randomly sets $\gamma' \in \{0,1\}$ and sends back to $A$ the following cipher text: $C_1 = g^a, C_2 = u^b, C_3 = M_{\gamma'} T$. $A$ then sends its guess $\gamma''$ back to $B$. If $\gamma' = \gamma''$, then $B$ guesses that $\gamma = 0$. If $\gamma' \neq \gamma''$, then $B$ guesses that $\gamma = 1$.

To analyze the success probability of $B$, we suppose that $A$ succeeds with probability $\frac{1}{2} + \epsilon$ when given a valid encryption of $M_{\gamma'}$, where $\epsilon$ is non-negligible. If given an invalid encryption, we only assume that $A$ must output 0 or 1 in polynomial time. If $\gamma = 0$, then $T = v^{a+b}$, and so $A$ is given a valid encryption of $M_{\gamma'}$, and hence will guess $\gamma'$ correctly with probability $\frac{1}{2} + \epsilon$. In this case, $B$ guesses $\gamma$ correctly with probability $\frac{1}{2} + \epsilon$. If $\gamma = 1$ and $T$ is chosen randomly, then $M_{\gamma'} T$ is random.
Problem 4 Consider the following security game for an encryption system with a 
(finite) message space $\mathcal{M}$. The game is similar to CPA security except the attacker 
only gets to submit one message $M^\ast$. The challenger flips a bit $\gamma$ and encrypts $M^\ast$ 
if $\gamma = 0$ and a uniformly random message $M_R \in \mathcal{M}$ if $\gamma = 1$. Show that this new 
game is equivalent to the traditional CPA game by arguing that an encryption system 
secure under this new game must also be secure under the traditional CPA encryption 
game. (You only need to show this direction since the other one is trivial.)

Solution. We assume we have a program $A$ which is a CPA attacker for a given 
encryption system. We suppose that $A$ succeeds with probability $\frac{1}{2} + \epsilon$ for some 
non-negligible $\epsilon$. Now, we will construct an attacker $B$ in the new security game, 
which starts with the challenger sending $B$ the public parameters. $B$ sends these 
public parameters to $A$, which generates two messages $M_0, M_1$ and sends them back 
to $B$. $B$ then randomly sets $\gamma' \in \{0, 1\}$ and sends $M^\ast = M_{\gamma'}$ to the challenger. 
The challenger sends $T$ back to $B$, who forwards it to $A$. We note that $T$ is the 
encryption of $M^\ast$ if $\gamma = 0$ and is the encryption of a random message otherwise. 
Now, $A$ generates a guess $\gamma''$ and sends it to $B$. If $\gamma' = \gamma''$, then $B$ guesses that $\gamma = 0$. 
Otherwise, it guesses that $\gamma = 1$.

To analyze the success probability of $B$, we note:

$$Pr[B \text{ succeeds}] = \frac{1}{2}Pr[B \text{ succeeds}|\gamma = 0] + \frac{1}{2}Pr[B \text{ succeeds}|\gamma = 1].$$

If $\gamma = 0$, then $T$ is the encryption of $M'_\gamma$, so $\gamma'' = \gamma'$ with probability $\frac{1}{2} + \epsilon$ by 
deinition of $A$. If $\gamma = 1$, then $T$ does not depend at all on the value of $\gamma'$, and hence 
$A$ has no information about $\gamma'$ and will guess $\gamma'' = \gamma'$ with probability $\frac{1}{2}$. Hence, 
$Pr[B \text{ succeeds}] = \frac{1}{2} \left( \frac{1}{2} + \epsilon \right) + \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{2} + \frac{\epsilon}{2}$. We have thus shown that an encryption 
system which is not secure in the CPA game is not secure in the new game, and so a 
system that is secure in the new game must also be secure in the CPA game. □
Problem 5 Assume there exists an IBE system secure under the standard (full) definition of security with algorithms: Setup, KeyGen, Encrypt, Decrypt. Moreover, assume identities in the system are in the space \{0,1\}^\gamma, where \gamma is the security parameter. Create a new IBE system that is selectively secure, but not fully secure. First, describe your system with algorithms: Setup', KeyGen', Encrypt', Decrypt'. Your new algorithms can build upon the existing ones (some might even be the same). Then prove that the system is selectively secure. Finally, show an attack in the full security game. Hint: Your system might look “unnatural” to allow such an attack.

Solution. We let Setup' be just like Setup, except that we randomly pick an ID from \{0,1\}^\gamma, generate a secret key for it, and add this identity along with its secret key to the public parameters. (In essence, we are uniformly randomly choosing one key to be artificially vulnerable.) The algorithms KeyGen', Encrypt', and Decrypt' are then exactly like KeyGen, Encrypt, and Decrypt. An attack in the full security setting is clear: the attacker sees the vulnerable ID in the public parameters along with its secret key, announces that ID to be ID^* and can then simply use the revealed secret key to decrypt. Nonetheless, we will prove this new IBE system is selectively secure.

We will show that if there is an attacker A on the new scheme in the selective security game, then this attacker can be used to create an algorithm B which attacks the original IBE scheme in the full security game, contradicting our assumption that the original system was secure in this setting. In the full security game, the IBE challenger first sends the public parameters to B. B calls on A to announce ID^*, which A will try to attack. Then B randomly chooses ID \in \{0,1\}^\gamma and generates a corresponding secret key by querying the IBE challenger for a key to ID in the first key phase of the full security game. It then sends ID, its secret key, and the public parameters to A. It simulates the first key phase with A in the selective game by just passing along the queries from A to the IBE challenger as part of the first key phase in the full game and sending back the received secret keys. Then A generates M_0 and M_1 which it sends to B. B sends ID^*, M_0, and M_1 to the IBE challenger, who randomly sets \gamma \in \{0,1\}. The IBE challenger sends B the encryption of M_\gamma using the public parameters and ID^*. B passes this value to A. B then simulates the second key phase with A by again forwarding queries and answers to and from
the IBE challenger. Finally, $A$ produces a guess $\gamma'$, which $B$ sets as its own guess.

We suppose that the selective IBE' attacker $A$ succeeds with probability $\frac{1}{2} + \epsilon$ in the selective security game. Now, if $ID^* = ID$, then we fail, since we have asked the IBE challenger for the secret key to the identity we wish to attack. The chance of this happening is $2^{-\gamma}$. The IBE' attacker fails with probability $\frac{1}{2} - \epsilon$, so by the union bound, the probability that either the IBE' attacker fails or $ID^* = ID$ is at most $\frac{1}{2} - \epsilon + 2^{-\gamma}$. If neither of these two things happens, then $\gamma' = \gamma$, and $B$ succeeds. So $B$ succeeds with probability at least $\frac{1}{2} + \epsilon - 2^{-\gamma}$. We note that $\epsilon - 2^{-\gamma}$ is non-negligible because $\epsilon$ is assumed to be non-negligible and $2^{-\gamma}$ is negligible (exponentially small in the security parameter). □