University of Maryland
CMSC456 — Introduction to Cryptography
Professor Jonathan Katz

Lecture 31

1 El Gamal Encryption

Again, we will let $\mathbb{G}$ denote a finite, cyclic group. Recall the DDH assumption in $\mathbb{G}$: we say the DDH problem in $\mathbb{G}$ is (t, ϵ)-hard if for all algorithms A running in time t we have:

$$\left|\Pr[x, y \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A(g, g^x, g^y, g^{xy}) = 1] - \Pr[x, y, z \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A(g, g^x, g^y, g^z) = 1]\right|.$$

We also recall the description of the El Gamal encryption scheme we gave last time:

1. $\mathcal{K}$ chooses a random generator $g \in \mathbb{G}$ and a random number $x \in \mathbb{Z}_{|\mathbb{G}|}$. It then computes $h = g^x$. The public key is (g, h) and the secret key is x.
2. The message space will be the group $\mathbb{G}$ itself. To encrypt a message $m \in \mathbb{G}$, the sender picks a random $r \in \mathbb{Z}_{|\mathbb{G}|}$ and sends ciphertext $(g^r, h^r m)$.
3. To decrypt a ciphertext (A, B), the receiver computes $m = B/A^x$. (For any group elements g', h', the notation g'/h' simply means $g'(h')^{-1}$.)

As we noted last time, clearly the encryption scheme is *insecure* if the discrete logarithm problem is easy in $\mathbb{G}$. However, it is not enough for the discrete logarithm problem to be hard in order for the scheme to be secure; in fact, we need to assume the stronger DDH assumption in $\mathbb{G}$. However, the DDH assumption will be sufficient to prove the El Gamal encryption scheme secure, as we now show:

Theorem 1 *If the DDH problem is (t, ϵ)-hard in $\mathbb{G}$, then the El Gamal encryption scheme is $(t, 2\epsilon)$-secure against ciphertext-only attacks.*

Proof Assume we have an adversary A which can "break" the security of El Gamal encryption; namely, there exist two messages m_0, m_1 for which (informally):

$$|\Pr[C \leftarrow \mathcal{E}_{PK}(m_0) : A(PK, C) = 0] - \Pr[C \leftarrow \mathcal{E}_{PK}(m_1) : A(PK, C) = 0]| = \delta. \qquad (1)$$

We will show that δ must be small. In particular, we show how to use adversary A to construct an algorithm A' which can solve the DDH problem with probability $\delta/2$. Since we know that the DDH problem is (t, ϵ)-hard, we must have $\delta \leq 2\epsilon$ and we are done.

We first re-write equation (1) in an equivalent form (we have done this before so we do not repeat the derivation here):

$$|2 \cdot \Pr[b \leftarrow \{0, 1\}; C \leftarrow \mathcal{E}_{PK}(m_b) : A(PK, C) = b] - 1| = \delta.$$

Now, define algorithm A' for the DDH problem as follows:

$A'(g, h_1, h_2, h_3)$
Set $PK = (g, h_1)$
$b \leftarrow \{0, 1\}$
set $C = (h_2, h_3 \cdot m_b)$
run $A(PK, C)$ to get output b'
if $b = b'$, guess 1 (this will represent the guess "Diffie-Hellman")
otherwise, guess 0 (this will represent the guess "random")

Let $x \stackrel{\text{def}}{=} \log_g h_1, r \stackrel{\text{def}}{=} \log_g h_2$, and $z \stackrel{\text{def}}{=} \log_g h_3$. Note the following:

- PK is always a valid public key with corresponding secret key x (of course, neither A nor A' know x, but this is irrelevant).
- If $z = rx$ (i.e., (g, h_1, h_2, h_3) is a Diffie-Hellman tuple), then the ciphertext C is a legitimate encryption of message m_b. This is so because C is then of the form $(g^r, g^{rx} \cdot m_b) = (g^r, h_1^r \cdot m_b)$. Thus:

  $$\Pr[x, r \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A'(g, g^x, g^r, g^{rx}) = 1] = \Pr[b \leftarrow \{0, 1\}; C \leftarrow \mathcal{E}_{PK}(m_b) : A(PK, C) = b].$$

- On the other hand, if z is uniformly distributed, independent of x and r, then the ciphertext C is (with high probability) neither an encryption of m_0 nor m_1. In fact, in this case the second component of the ciphertext $(h_3 \cdot m_b)$ is uniformly distributed in $\mathbb{G}$, independent of m_0, m_1, or b. Thus, *even if A is all powerful* it cannot possibly output $b' = b$ with probability any different from $1/2$. This implies:

  $$\Pr[x, r, z \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A'(g, g^x, g^r, g^z) = 1] = 1/2.$$

Thus:

$$\begin{aligned} \left|\Pr[x, r \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A'(g, g^x, g^r, g^{rx}) = 1] - \Pr[x, r, z \leftarrow \mathbb{Z}_{|\mathbb{G}|} : A'(g, g^x, g^r, g^z) = 1]\right| & \\ = \left|\Pr[b \leftarrow \{0, 1\}; C \leftarrow \mathcal{E}_{PK}(m_b) : A(PK, C) = b] - 1/2\right| & \\ = \delta/2 & \\ \leq \epsilon. & \end{aligned}$$

We conclude that $\delta \leq 2\epsilon$, proving the theorem. ■

2 El Gamal Encryption in Practice

The above proof of security did not rely on any properties of group $\mathbb{G}$ other than the fact that it was a finite, cyclic group in which the DDH problem was "hard". Thus, all we need to do is find a finite, cyclic group in which the DDH assumption is believed to hold and we have a secure encryption scheme! We give here one example of such a group.

We mentioned in an earlier lecture that the multiplicative group $\mathbb{Z}_p^*$ (for p prime) is a cyclic group. Even better, the discrete logarithm problem in this group is believed to be hard (for large p). Unfortunately (as you will show), the DDH assumption *does not* hold in this group, making it unsuitable for the El Gamal encryption scheme.

We need a slightly more complicated example. Let $p = 2q + 1$ where p and q are both prime. Let $\mathbb{G}$ denote the set of quadratic residues in $\mathbb{Z}_p^*$. Since we know that exactly half the elements of $\mathbb{Z}_p^*$ are quadratic residues, we have $|\mathbb{G}| = (p-1)/2 = q$. It is not too difficult to show that $\mathbb{G}$ is a group (under multiplication), and only a little harder to show that it is in fact a cyclic group. It is also easy to identify elements of this group (or, for that matter, to choose random elements in this group) since it is possible to efficiently determine whether elements $x \in \mathbb{Z}_p^*$ are quadratic residues or not. Finally, the DDH assumption is believed to hold in this group. This group is most commonly used to instantiate the El Gamal encryption scheme.

There are other examples of cyclic groups (most well-known are those based on elliptic curves) in which the DDH assumption is believed to hold. We stress that as long as the DDH assumption is believed to hold (and as long as multiplication in the group can be done efficiently), the structure of the group is unimportant as far as the security of the El Gamal encryption scheme is concerned.

3 Stronger Notions of Security

Our treatment of public-key encryption parallels our treatment of private-key encryption. In the private-key case, we first gave a definition of security against ciphertext-only attacks and showed a construction of a scheme secure with respect to this definition. We then considered a definition of security against chosen-plaintext attacks (that is, security in the sense of left-or-right indistinguishability) and showed that this was a *strictly stronger definition*. Namely, there exist schemes secure against ciphertext-only attacks which are definitely *not* secure in the sense of indistinguishability (an example of such a scheme is the one-time pad). What happens in the public-key case?

Recall the *left-or-right oracle* that we used also when defining security of private-key encryption. This oracle is indexed by a bit b and a public key PK (output by some key generation algorithm for a public-key encryption scheme). The oracle $\mathsf{LR}_{b,PK}$ takes two inputs; $\mathsf{LR}_{b,PK}(m_0, m_1)$ returns $\mathcal{E}_{PK}(m_b)$, where this encryption is done randomly each time the oracle is accessed. With the definition of this oracle in place, we may give the following equivalent definition of security against ciphertext-only attacks:

Definition 1 *Public-key encryption scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ is (t, ϵ)-secure against ciphertext-only attacks if for all adversaries A running in time t we have:*

$$\left| 2 \cdot \Pr[(PK, SK) \leftarrow \mathcal{K}; b \leftarrow \{0,1\} : A^{\mathsf{LR}_{b,PK}(\cdot,\cdot)}(PK) = b] - 1 \right| \leq \epsilon,$$

where A may only query the LR oracle a single time.

Note that A — in addition to having oracle access to LR — is also explicitly given PK as input because we are in the public-key setting. (A's access to LR is as a "black box", and thus A cannot automatically determine PK [or, of course, b] from its interaction with the oracle; it is for this reason that PK is explicitly given to A.)

This naturally leads us to a definition of security in the sense of left-or-right indistinguishability. As inthe private-key case, this definition also corresponds to security against chosen-plaintext attacks, and also security when multiple messages are encrypted under the same public key.

Definition 2 *Public-key encryption scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ is (t, ϵ)-secure in the sense of indistinguishability if for all adversaries A running in time t we have:*

$$\left| 2 \cdot \Pr[(PK, SK) \leftarrow \mathcal{K}; b \leftarrow \{0,1\} : A^{\mathsf{LR}_{b,PK}(\cdot,\cdot)}(PK) = b] - 1 \right| \leq \epsilon,$$

where A may only query the LR oracle an unlimited number of times (of course, the number of queries will be fewer than t).

Clearly, this definition is no weaker than the definition of security against ciphertext-only attacks. In fact, a somewhat surprising result is that these two definitions are in fact *equivalent*! Thus, any public-key encryption scheme secure against ciphertext-only attacks is automatically also secure in the sense of indistinguishability. We give a formal statement of this result, and a full proof of this assertion, next time.