El Gamal
A DDH based encryption scheme

Foundations of Cryptography
Computer Science Department
Wellesley College

Fall 2016

Table of contents

The El Gamal encryption scheme

- El Gamal, based on the hardness of the decisional Diffie-Hellman (DDH) problem, is commonly used public-key encryption scheme.
- Before introducing the scheme proper, some background mathematics.

A useful lemma*

Lemma 11.15. Let $\mathbb{G}$ be a finite group, and let $m \in \mathbb{G}$ be an arbitrary element. Then choosing a uniform $k \leftarrow \mathbb{G}$ and setting $k' := k \cdot m$ gives the same distribution for k' as choosing uniform $k' \leftarrow \mathbb{G}$. That is, for any $\hat{g} \in \mathbb{G}$

$$\Pr[k \cdot m = \hat{g}] = 1/|\mathbb{G}|,$$

where the probability is taken over random choice of k.

Proof.

*In other words, the distribution of k' is independent of m; this means that k' contains no information about m.

*A perfectly-secret private-key encryption scheme**

- The sender and receiver share a random element $k \leftarrow \mathbb{G}$.
- To encrypt $m \in \mathbb{G}$, the sender computes the ciphertext $k' := k \cdot m$.
- To decrypt the ciphertext k', the receiver computes $m := k'/k$.

*The one-time pad is exactly of this form. The group is the set of all bit strings of some fixed length and the group operation is XOR.

A snag

- This only works if k is truly random, used only once, and shared in advance*.
- In a public-key setting, a different scheme is needed to allow the receiver to decrypt.
- Pseudorandom to the rescue. Element k will be defined in such a way that the receiver will be able to compute k from her private key, yet k will "look random".

*Bad news for any public-key scheme.

The El Gamal encryption scheme

Construction 11.16.

Let $\mathcal{G}$ be a polynomial-time algorithm that, on input 1^n, outputs a cyclic group $\mathbb{G}$, its order q (with $\|q\| = n$), and a generator g.

- Gen: On input 1^n run $\mathcal{G}(1^n)$ to obtain $(\mathbb{G}, q, g)$. Then choose a random $x \in \mathbb{Z}_q$ and compute $h := g^x$. The public key is $\langle \mathbb{G}, q, g, h \rangle$ and the private key is $\langle \mathbb{G}, q, g, x \rangle$.
- Enc: On input a public key $pk = \langle \mathbb{G}, q, g, h \rangle$ and a message $m \in \mathbb{G}$, choose a random $y \leftarrow \mathbb{Z}_q$ and output the ciphertext

$$\langle g^y, h^y \cdot m \rangle.$$

- Dec: On input a private key $sk = \langle \mathbb{G}, q, g, x \rangle$ and a ciphertext $\langle c_1, c_2 \rangle$, output

$$m := c_2 / c_1^x.$$

Correctness of El Gamal encryption scheme

To see that decryption succeeds, let $\langle c_1, c_2 \rangle = \langle g^y, h^y \cdot m \rangle$ with $h = g^x$. Then

$$\frac{c_2}{c_1^x} = \frac{h^y \cdot m}{(g^y)^x} = \frac{(g^x)^y \cdot m}{g^{xy}} = \frac{g^{xy} \cdot m}{g^{xy}} = m.$$

A simple example

Example 11.17. Let $q = 83$ and $p = 2q + 1 = 167$, and let $\mathbb{G}$ denote the group of *quadratic residues* modulo p.*

Since the order of $\mathbb{G}$ is prime, any element of $\mathbb{G}$ except 1 is a generator; take $g = 2^2 = 4 \mod 167$. Suppose the receiver chooses a secret key $x = 37 \in \mathbb{Z}_{83}$ so the public key is

$$pk = \langle p, q, g, h \rangle = \langle 167, 83, 4, [4^{37} \mod 167] \rangle = \langle 167, 83, 4, 76 \rangle,$$

where p represents $\mathbb{G}$.

To encrypt $m = 65 \in \mathbb{G}$, choose a uniform $y \in \mathbb{Z}_q$, say $y = 71$, then

$$\langle [4^{71} \mod 167], [76^{71} \cdot 65 \mod 167] \rangle = \langle 132, 44 \rangle$$

To decrypt, first compute $124 = [132^{37} \mod 167]$; then since $66 = [124^{-1} \mod 167]$, recover $m = 65 = [44 \cdot 66 \mod 167]$.

*Since p and q are prime, $\mathbb{G}$ is a subgroup of $\mathbb{Z}_p^*$ with order q.

Recall the Decisional Diffie-Hellman (DDH) problem

The *decisional Diffie-Hellman (DDH) problem* is to distinguish $\mathrm{DH}_g(h_1, h_2)$ from a random group element for randomly chosen h_1, h_2.

Definition 8.63. We say that the *DDH problem is hard relative to* $\mathcal{G}$ if for all probabilistic polynomial-time algorithms $\mathcal{A}$ there exists a negligible function negl such that

$$|\Pr[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] - \Pr[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1]| \leq \mathrm{negl}(n),$$

where in each case the probabilities are taken over the experiment in which $\mathcal{G}(1^n)$ outputs $(\mathbb{G}, q, g)$, and the random $x, y, z \in \mathbb{Z}_q$ are chosen.

The El Gamal encryption scheme

Theorem 11.18. If the DDH problem is hard relative to $\mathcal{G}$, then the El Gamal encryption scheme is CPA-secure.

Proof. We prove that El Gamal scheme Π has indistinguishable encryptions in the presence of an eavesdropping and let Proposition 11.3 take it from there.

Let $\mathcal{A}$ be a PPT adversary attacking El Gamal in $\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n)$. We show that there is a negligible function negl such that

$$\Pr[\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n) = 1] \leq \frac{1}{2} + \mathsf{negl}(n).$$

If things were truly random

Consider $\tilde{\Pi}$ where Gen is the same as Π, but encryption of a message m with respect to the public key $pk = \langle \mathbb{G}, q, g, h \rangle$ is done by choosing a random $y \leftarrow \mathbb{Z}_q$ and $z \leftarrow \mathbb{Z}_q$ and outputting the ciphertext

$$\langle g^y, g^z \cdot m \rangle$$

Lemma 11.15 implies that the second component of the ciphertext is a uniformly-distributed group element, and, in particular is independent of the message m. It follows that

$$\Pr[\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\tilde{\Pi}}(n) = 1] = \frac{1}{2}.$$

*Building a PPT distinguisher for the DDH problem**

Algorithm D:

The algorithm is given $\mathbb{G}, q, g, h_1, h_2, h_3$ as input

- Set $pk = \langle \mathbb{G}, q, g, h_1 \rangle$ and run $\mathcal{A}(pk)$ to obtain two messages $m_0, m_1 \in \mathbb{G}$.
- Choose a random bit b, and set $c_1 := h_2$ and $c_2 := h_3 \cdot m_b$.
- Give the ciphertext $\langle c_1, c_2 \rangle$ to $\mathcal{A}$ and obtain an output bit b'. If $b' = b$ output 1; otherwise output 0.

We analyze the behavior of D. There are two cases.

*In other words, D receives $(\mathbb{G}, q, g, h_1, h_2, h_3)$ where $h_1 = g^x, h_2 = g^y$ and h_3 is either h^{xy} or h^z for some random x, y, z and the goal of D is to determine which is the case.

First verse

Case 1. Say the input D is generated by running $\mathcal{G}(1^n)$ to obtain $(\mathbb{G}, q, g)$ then choosing a random $x, y, z \leftarrow \mathbb{Z}_q$, and finally setting $h_1 := g^x, h_2 := g^y$, and $h_3 := g^z$.

Next D runs $\mathcal{A}$ on public key

$$pk = \langle \mathbb{G}, q, g, g^x \rangle$$

and a ciphertext constructed as

$$\langle c_1, c_2 \rangle = \langle g^y, g^z \cdot m_b \rangle$$

The view of $\mathcal{A}$ when run by D is distributed identically to its view in experiment $\mathrm{PubK}^{\mathrm{eav}}_{\mathcal{A},\tilde{\Pi}}(n)$. Since D outputs 1 precisely when $\mathcal{A}$ succeeds

$$\Pr[D(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] = \Pr[\mathrm{PubK}^{\mathrm{eav}}_{\mathcal{A},\tilde{\Pi}}(n) = 1] = \frac{1}{2}.$$

Second verse, (nearly) same as the first

Case 2. Say the input D is generated by running $\mathcal{G}(1^n)$ to obtain $(\mathbb{G}, q, g)$ then choosing a random $x, y \leftarrow \mathbb{Z}_q$, and finally setting $h_1 := g^x$, $h_2 := g^y$, and $h_3 := g^{xy}$.

Next D runs $\mathcal{A}$ on public key

$$pk = \langle \mathbb{G}, q, g, g^x \rangle$$

and a ciphertext constructed as

$$\langle c_1, c_2 \rangle = \langle g^y, g^{xy} \cdot m_b \rangle$$

The view of $\mathcal{A}$ when run by D is distributed identically to its view in experiment $\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n)$. Since D outputs 1 precisely when $\mathcal{A}$ succeeds

$$\Pr[D(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n) = 1].$$

Day is done

Since the DDH problem is hard relative to $\mathcal{G}$, there exists a negligible function negl such that

$$\begin{aligned} \mathsf{negl}(n) &\geq \left|\Pr[D(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] - \Pr[D(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1]\right| \\ &= \left|\frac{1}{2} - \Pr[\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n) = 1]\right| \end{aligned}$$

This implies that $\Pr[\mathsf{PubK}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n) = 1] \leq \frac{1}{2} + \mathsf{negl}(n)$ as required. □

Encoding binary strings

Remark. In order to fully specify a usable encryption scheme, we need to show how to encode binary strings as element of $\mathbb{G}$. Such an encoding depends upon the group under consideration. Here is on possibility when $\mathbb{G}$ is the subgroup of quadratic residues modulo a strong prime p, i.e., $q = (p-1)/2$ is also a prime.

Mapping. We show that the mapping $f : \{0, 1, \ldots, (p-1)/2\} \to \mathbb{G}$ given by $f(\tilde{m}) = [\tilde{m}^2 \mod p]$ is a bijection and effectively reversible.

Encoding. We can now map a string $\tilde{m}$ of length $n-1$ to an element $m \in \mathbb{G}$ as follows: given $\tilde{m} \in \{0,1\}^{n-1}$, interpret it as an integer and add 1 to obtain an integer $\tilde{m}$ with $1 \leq \tilde{m} \leq q$.* Then take $m = [\tilde{m}^2 \mod p]$.

*Recall that $n = \| q \|$

Chosen ciphertext attacks: Sad news once again

- In section 11.2.3 the authors give a precise definition of security against chosen-ciphertext attacks together with a number of realistic scenarios where such an attack might be carried out.
- Sadly, none of the public-key schemes discussed are secure under this definition. Indeed, the text also details attacks against each of textbook RSA (not surprising), PKCS #1 v1.5, and El Gamal.