ElGamal cryptosystem

Ganyu (Bruce) Xu

August 5, 2024

1 The ElGamal cryptosystem

The ElGamal cryptosystem is a public key encryption scheme that mainly operates on the discrete log problem. Each instance of the encryption scheme is parameterized by a cyclic group G with prime order q, a generator g of this cyclic group. The routines of the encryption scheme is shown in figure 1

Algorithm 1 `KeyGen`

1: $x \xleftarrow{\$} \mathbb{Z}_q$
2: $u \leftarrow g^x$
3: $\texttt{pk} \leftarrow u, \texttt{sk} \leftarrow x$
4: **return** $(\texttt{pk}, \texttt{sk})$

Algorithm 2 $\texttt{Enc}(\texttt{pk} = u, m \in G)$

1: $y \xleftarrow{\$} \mathbb{Z}_q$
2: $v \leftarrow g^y$
3: $w \leftarrow u^y$ $\qquad \triangleright\ w = g^{xy}$
4: $c \leftarrow (v, m \cdot w)$
5: **return** c

Algorithm 3 $\texttt{Dec}(\texttt{sk} = x, c)$

1: $(c_1, c_2) \leftarrow c$
2: $\hat{w} \leftarrow c_1^x$
3: $\hat{m} \leftarrow c_2 \cdot \hat{w}^{-1}$
4: **return** $\hat{m}$

Figure 1: ElGamal encryption scheme is IND-CPA secure if DDH holds

The IND-CPA security of the ElGamal cryptosystem depends on the hardness of the following two problems:

Definition 1.1 (Computational Diffie-Hellman Problem)**.** *Let G be a cyclic group with prime order q and generator g. Let $x, y \xleftarrow{\$} \mathbb{Z}_q$ be uniformly random samples. Given g, g^x, g^y, compute g^{xy}*

Definition 1.2 (Decisional Diffie-Hellman Problem)**.** *Let G be a cyclic group with prime order q and generator g. Let $x, y, z \xleftarrow{\$} \mathbb{Z}_q$ be uniformly random samples. Given g, g^x, g^y, distinguish g^{xy} from g^z*

Theorem 1.1. *For every IND-CPA adversary A against the ElGamal cryptosystem, there exists an adversary B against the DDH game such that*

$$\mathit{Adv}(A) = 2 \cdot \mathit{Adv}(B)$$

Because ElGamal ciphertexts are malleable, this encryption scheme is not secure against chosen-ciphertext attacks. However, a hybrid encryption scheme can be used to achieve chosen-ciphertext attack security [BS20]. Denote this construction by "ElGamal HPKE".

To construct the HPKE, we need the cyclic group G of prime order q and generator g. We also need a symmetric cipher $(\mathtt{Enc}_S, \mathtt{Dec}_S)$ defined over $(\mathcal{K}, \mathcal{M}, \mathcal{C})$, and a hash function $H : G \to \mathcal{K}$. The routines are listed in figure 2.

Algorithm 4 $\mathtt{KeyGen}$

$x \leftarrow \mathbb{Z}_q$
$u \leftarrow g^x$
$\mathtt{pk} \leftarrow u$
$\mathtt{sk} \leftarrow x$
return $(\mathtt{pk}, \mathtt{sk})$

Algorithm 5 $\mathtt{Enc}(\mathtt{pk} = u, m \in \mathcal{M})$

$y \xleftarrow{\$} \mathbb{Z}_q$
$v \leftarrow g^y$
$w \leftarrow u^y$ $\qquad \triangleright\ w = g^{xy}$
$k \leftarrow H(w)$
$c' \leftarrow \mathtt{Enc}_S(k, m)$
$c \leftarrow (v, c')$
return c

Algorithm 6 $\mathtt{Dec}(\mathtt{sk} = x, c)$

$(v, c') \leftarrow c$
$\hat{w} \leftarrow v^x$
$\hat{k} \leftarrow H(\hat{w})$
$\hat{m} \leftarrow \mathtt{Dec}_S(\hat{k}, c')$
return $\hat{m}$

Figure 2: ElGamal HPKE

Theorem 1.2. *For every IND-CCA adversary A against the HPKE, there exists an* interactive computational Diffie-Hellman problem *adversary B and an IND-CPA adversary C against the symmetric encryption scheme such that*

$$\mathit{Adv}(A) \leq \textit{NEED TO WRITE THIS PART}$$

While having a decryption oracle breaks the decisional Diffie-Hellman assumption, we still feel confident that the computational Diffie-Hellman remains hard, which is how we can reason about the security of the HPKE under chosen-ciphertext attacks.

Unfortunately, *this is not the case in Kyber.* Having a decapsulation oracle that can take arbitrary number of decapsulation queries will allows an adversary to complete recover the secret key, unlike ElGamal

HPKE where having a decryption oracle does not give away the secret key. *There is no immediate parallel loosening of security assumption we can make in Kyber*

References

[BS20] Dan Boneh and Victor Shoup. A graduate course in applied cryptography. *Draft 0.5*, 2020.