**实验文档**

**1. Identity-based Encryption with Verifiable Outsourcing Revocation**

1）比较方案1、方案2、方案3各个阶段的运行时间，包括setup、keygen、encrypt、decrypt和keyupdate中PKG与server的时间。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 方案1 | 方案2 | 方案3 |
| setup |  |  |  |
| keygen |  |  |  |
| encrypt |  |  |  |
| decrypt |  |  |  |
| Keyupdate-PKG | 0 | 0 |  |
| Keyupdate-server | 0 |  |  |

2）比较方案3中PKG直接进行密钥更新及密钥更新外包时PKG及server的时间。设定用户总数为215（或其他合理的数字），横坐标为删除用户比例（分别为5%，15%，25%，50%，75%），纵坐标为时间。

**方案1（合数阶群，没有删除算法，设定**为512bit**）**

**Setup**(): The setup algorithm is run by PKG. It chooses a bilinear group of order , where are distinct primes. Let denote the subgroup of order in . It then chooses , , and generates the public parameters. The master key is and a generator of .

**KeyGen**(): For a private key query on identity , PKG randomly selects ,, and computes:

.

**Encrypt**(): For a message , and an identity , randomly chooses and creates the ciphertext as:

.

**Decrypt**(): Take as input a ciphertext under an identity at time period , and a private key , it decrypts the ciphertext as below:

.

**方案2（素数阶群，支持删除用户，设定**为512bit**）**

**Setup**(): The setup algorithm is run by PKG. It chooses two groups of order, is a generator of, and  is a bilinear map.  are two collision-resist hash functions. It then chooses , , and computes . The public parameters as published as , and the master key is .

**KeyGen**(): For a private key query on identity , PKG first checks whether exists in a revocation list . If yes, the key generation algorithm returns . Else, PKG randomly selects , and computes:

.

Next, PKG reads the current time period from a time list , and randomly chooses , and sets:

.

Finally, PKG outputs and appends  to an update list .

**Encrypt**(): For a message , time period , and an identity , randomly chooses and creates the ciphertext as:

.

**Decrypt**(): Take as input a ciphertext under an identity at time period , and a private key , it decrypts the ciphertext as below:





.

**Revoke**(): If users with identities in the set  are to be revoked at time period , PKG updates time period as  and the revocation list as . Finally send a copy for the updated revocation list as well as the new time period to the server.

**KeyUpdate**(): Upon receiving a key update query on at time period , the server firstly checks whether exists in the revocation list , if so the server returns . Otherwise, the server obtains  from UL, randomly selects , and sets:

.

Finally, PKG outputs .

**方案3（合数阶群，支持删除用户，设定**为512bit**）**

**Setup**(): The setup algorithm is run by PKG. It chooses a bilinear group of order , where are distinct primes. Let denote the subgroup of order in . is a collision-resist hash function. It then chooses , , and generates the public parameters as follows:

.

The master key is and a generator of .

**KeyGen**(): For a private key query on identity , PKG first checks whether exists in a revocation list . If yes, the key generation algorithm returns . Else, PKG randomly selects ,, and computes:

.

Next, PKG reads the current time period from a time list , and randomly chooses , and sets:

.

Finally, PKG outputs and appends  to an update list .

**Encrypt**(): For a message , time period , and an identity , randomly chooses and creates the ciphertext as:

.

**Decrypt**(): Take as input a ciphertext under an identity at time period , and a private key , it decrypts the ciphertext as below:





.

**Revoke**(): If users with identities in the set  are to be revoked at time period , PKG updates time period as  and the revocation list as .

**KeyUpdate**(): Upon receiving a key update query on at time period , PKG firstly checks whether exists in the revocation list , if so PKG returns . Otherwise, PKG obtains  from UL, randomly selects , and sets:

.

Finally, PKG outputs .

**KeyUpdateOut**():Upon receiving a key update query on , PKG first check whether exists in the revocation list . If yes, PKG returns . Else, PKG chooses random , and computes . Upon receiving a key update query on , PKG firstly checks whether exists in the revocation list , if so PKG returns . Otherwise, PKG obtains  from UL, runs *Rand* forone time to create one blinding pair , sets and appends  to an outsourcing list .

PKG queries  in random order as:

, , .

Similarly, PKG queries  in random order as:

, , .

PKG verifies that both and generate the correct outputs, that is to say,

, ,

.

If not, PKG outputs “error”; otherwise, PKG selects random , computes





,

and sets , .