doi 10.3969/j.issn.1671-1122.2012.09.010

基于同态加密的可信云存储平台

李美云，李剑，黄超

（北京邮电大学，北京 100876）

摘 要：基于全同态加密技术的数据检索方法可以直接对加密的数据进行检索，不但能保证被检索的数据不被统计分析，还能对被检索的数据做基本的加法和乘法运算而不改变对应明文的顺序，既保护了用户的数据安全，又提高了检索效率。文章就是针对上述问题提出的一个系统级解决方案，主要致力于解决云存储系统中信息的安全存储与管理问题。为实现此目标，该方案采用了同态加密算法，以及在这种算法基础上提出的一种密文检索算法，既保证了用户数据的安全性，又保证了服务器能够对存储的用户密文直接进行操作，实现了对密文的直接检索，平衡了云存储系统中保证用户数据的安全性和服务于云计算之间的关系。

关键词：同态加密；同态加密检索；云计算；数据安全

中图分类号：TP393.08 文献标识码：A 文章编号：1671-1122（2012）09-0035-06

A Credible Cloud Storage Platform based on Homomorphic Encryption

LI Mei-yun, LI Jian, HUANG Chao

(*Bejing University of Posts and Telecommunications, Beijing* 100876, *China*)

Abstract: Data retrieval method based on homomorphic encryption techniques can be directly on the encrypted data retrieval, not only to ensure the data is not retrieved statistical analysis, to do basic addition and multiplication can be retrieved data without changing the corresponding plaintext the order both to protect the security of user data to improve the retrieval efficiency.This work is proposed to address the problem, a system-level solutions, to solve the problem of secure storage and management of information in the cloud storage system. To achieve this objective, the work uses a homomorphic encryption algorithm, this algorithm on the basis of a cipher text retrieval algorithms, both to ensure the security of user data, but also to ensure the server can store user-ciphertext direct operation, the ciphertext to retrieve the balance of the cloud storage system to ensure the security of user data and services, the relationship between cloud computing.

Key words: homomorphic encryption; homomorphic encryption to retrieve; cloud computing; data security

1 现状分析

可信云平台是实现互联网基于信任判定的安全保护机制的重要手段之一。本文结合可信计算与云安全的优势，研究可信云平台关键技术，主要是实现了在搭建云平台下的基于同态加密技术的云数据存储、数据加密、解密以及检索技术。可以解决云平台中海量数据信息的安全存储、高效检索以及智能处理，进一步保证用户的隐私安全等。

1.1 云存储平台整体架构

云存储平台整体架构可划分为 4 个层次，自底向上依次是：数据存储层、数据管理层、数据服务层以及用户访问层。云存储平台整体架构如图 1 所示。

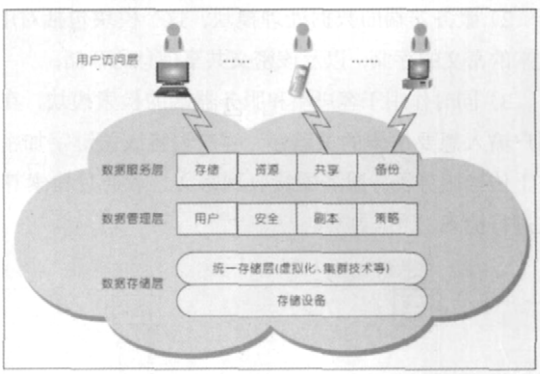

图 1 智能可信云存储平台整体架构

1.2 基于同态加密技术的云数据安全存储

同态加密是一种加密形式，它允许人们对密文进行特定的代数运算，得到仍然是加密的结果，与对明文进行同样的运算再将结果加密一样。换言之，这项技术令人们可以在加密的数据中进行诸如检索、

收稿时间：2012-07-28

基金项目：国家自然科学基金 [61100205]

作者简介：李美云（1991-），女，山东，本科，主要研究主向：信息安全；李剑（1976-），男，陕西，副教授，博士，主要研究主向：信息安全；黄超（1991-），女，重庆，本科，主要研究主向：通信工程。

比较等操作，得出正确的结果，而在整个处理过程中无需对数据进行解密。其意义在于，真正从根本上解决将数据及其操作委托给第三方时的保密问题。

在本文提出的解决方案中用户可以不用将加密后的数据从云服务器中下载下来进行检测、检索、更新等，这些行为可以直接在服务器上完成，而不是传统的下载到本地操作后再上传到服务器，由于传输的数据量更少，就和在本地直接进行操作效果一样。

2 实现方案

2.1 云存储系统的搭建

我们所使用的云存储系统最主要的目的在于构建一个平台，为加密、解密、检索算法提供实际的使用环境，并模拟实际应用中的云平台，为当今云系统面临的数据安全与信息处理问题提供解决方案。

该云存储系统使用的是典型的服务器 - 客户端模型。在客户端，用户首先选定想要上传的数据，然后客户端程序使用同态加密算法将这些数据进行加密，然后将加密后的密文上传到服务器端；服务器接受从客户端发送过来的密文数据，并把密文数据和相应的一些信息存储在服务器上；当该用户或者其它拥有数据共享权限的用户需要使用该数据时，从服务器端下载密文数据，在客户端解密得到明文消息；当用户需要进行关键词检索时，首先在客户端将明文关键字加密成密文，然后将密文关键字发送给服务器，服务器接收到密文关键字后在存储的密文数据中进行检索。

2.2 将云存储系统分为3个功能模块

1）客户端的上传 \ 下载数据模块，这个模块不仅包括了数据在服务器和客户端之间的传输，也包含了在向服务器上传时对明文的加密以及从服务器下载数据时对密文的解密；

2）服务器端的数据处理模块，这个模块包括对用户上传过来的密文的存储，以及该密文共享信息的存储；

3）同时作用于客户端和服务器端的检索模块，在客户端用户输入想要检索的关键字，客户端将该关键字加密，将密文上传给服务器，服务器接收到密文之后通过检索算法对密文进行检索。

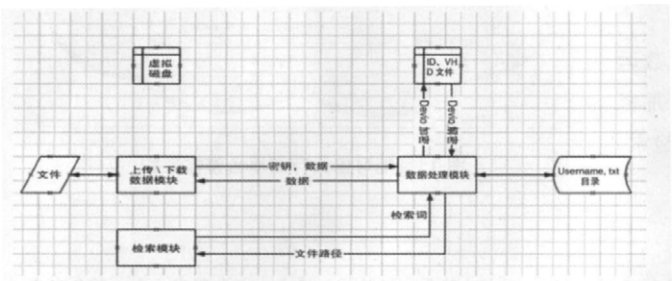

图2 云存储功能模块结构图

2.3 将云存储系统分为2个结构模块

1）客户端模块：这个模块包括以下几个功能：数据的同态加密、解密，密文数据的上传和下载，对检索关键字的加密。

2）服务器模块：这个模块主要功能是接收客户端发来的密文数据和相应的一些信息，并将它们存储在服务器上。在检索的过程中主要负责对密文检索字的匹配。

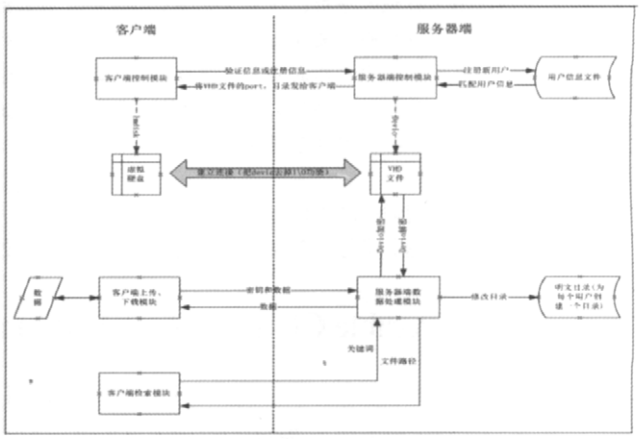

图3 云存储系统的结构图

2.4 云存储系统主要有4个控制模块

1）客户端注册模块，新用户通过这个模块加入到云存储系统中，可信第三方通过用户注册信息为新用户产生一对密钥并且分发给新用户；

2）客户端登陆模块，用户通过这个模块连接到服务器，实现数据的上传、下载、删除等操作；

3）服务器控制模块，服务器通过该模块监视用户的连接，实现数据的传输；

4）客户端功能模块，它和服务器控制模块共同作用实现对关键字的检索功能。

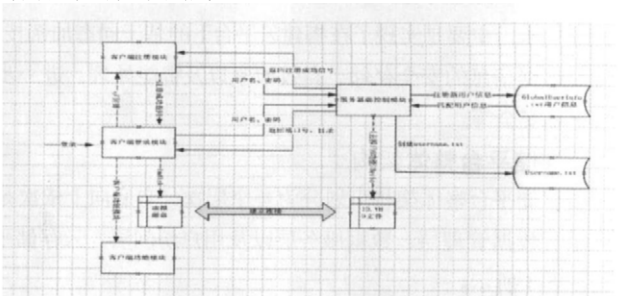

图 4 云存储控制模块结构图

搭建好该平台后，我们所要做的就是在上传 \ 下载模块添加使用同态加密算法的加密、解密程序，实现对用户数据的本地加密解密；在检索模块添加相对应的检索程序，实现在服务器端对密文数据的直接检索。

2.5 同态加密算法的实现

2.5.1 同态加密算法的优势

以往加密手段的一个弊端在于它通常是将数据保存在盒

子内而不让外界使用或者分析数据，除非使用解密密钥将盒子打开，这样就对数据的安全性构成威胁，同时使得效率降低。而完全同态加密方案可以让你在数据加密的情况下对数据进行分析和计算。用同态加密算法对数据进行加密后，无需解密就可以对其进行排序和搜索，提高了数据处理的效率，同时保证了数据的安全。

经典的加密算法中，具有很强的扩散性，如果密文发生错误，如被篡改、数据丢失等，则其所有的数据都被破坏，不能被恢复，而对于同态加密算法，正确的数据仍然可以得到正确的解密结果。

2.5.2 同态加密算法的原理

如果说，一种加密算法，对于乘法和加法都能找到一种操作，这种操作可以实现对加密后的数据进行处理，将处理后的数据再进行解密，得到的结果与对未加密的数据做同样的处理得到的结果相同，就称其为全同态加密算法。同态加密技术为解决物联网认证与访问控制、密钥协商管理、云计算以及电子商务等方面的安全和隐私保护问题提供了新的思路，并且在密码学领域一直都是研究的重点，但直到目前还没有真正实用的全同态加密算法，因为在同步加密方案成为实用工具前，还需要进行很多理论上的工作以提高其效率。不过，IBM 的研究员已使其在该方向上前进了一大步，有效性已在逐步改善。

2.5.3 同态加密算法的实现

1）加密算法

首先将明文比特分组（分组长度根据安全需求来确定），然后对每个明文分组长度 m_i 做加密运算，最后将加密得到的分组密文依次合并，得到加密密文。

具体的过程分为如下几步 ：

（1）选取随机产生的两个安全大素数 P 和 Q（目前两个数的长度都接近 512bit 是安全的）；

（2）计算乘积 $N=P\times Q$ ；并且生成一个随机数 $R1$ ；

（3）把消息 M 分组为若干长度 L（L 的长度应该小于 P）的消息分组 $M=m_1m_2m_3\cdots m_t$ ；

（4）使用加密算法 $c_i=(m_i+P\times R1)\bmod N$，同时计算出密文 $C=c_1c_2c_3\cdots c_t$。

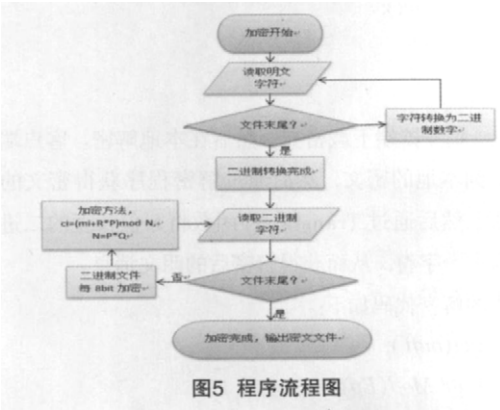

图5 程序流程图

2）解密算法

将接收到的密文分组，依次采用解密算法对分组密文进行解密，得到分组明文，然后将分组明文合并得到解密后的明文。

（1）接收方收到密文 C 并把密文 C 分组得到 $C=c_1c_2c_3\cdots c_t$ ；

（2）使用密钥 P 和解密算法 $m_i=c_i \bmod P$ 计算 m_t ；

（3）得到明文消息 $M=m_1m_2m_3\cdots m_t$。

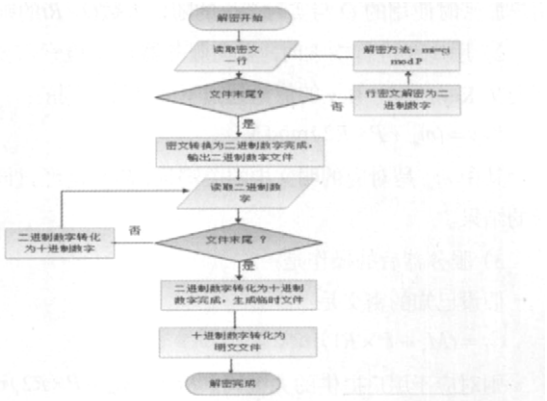

图6 程序流程图

2.5.4 对该算法进行同态性分析

1）同态加法特性验证

假设有两组明文 M_1 和 M_2，分别对他们用上述的加密算法进行加密得到 C_1 和 C_2。

$$C_1=(M_1+P\times R1)\bmod N$$

$$C_2=(M_2+P\times R2)\bmod N$$

则对于明文 $M_3=M_1+M_2$，有 ：

$$\begin{aligned}C_3&=C_1+C_2\\&=(M_1+M_2+P\times(R1+R2))\bmod N\\&=(M_3+P\times R3)\bmod N\end{aligned}$$

对 C_3 进行解密 ：

$$\begin{aligned}M_3&=C_3\bmod P\\&=(C_1+C_2)\bmod P\\&=(M_1+M_2+P\times(R1+R2))\bmod P\\&=M_1+M_2\end{aligned}$$

2）同态乘法特性验证

对于明文 $M_4=M_1\times M_2$，有 ：

$$\begin{aligned}C_4&=C_1\times C_2\\&=((M_1+P\times R1)\times(M_2+P\times R2))\bmod N\\&=(M_1\times M_2+M_1\times P\times R2+M_2\times P\times R1+P^2\times R1\times R2)\bmod N\\&=(M_1\times M_2+P\times(M_1\times R2+M_2\times R1+P\times R1\times R2))\bmod N\\&=(M_1\times M_2+P\times R3)\bmod N\end{aligned}$$

对 C_4 进行解密 ：

$$\begin{aligned}M_4&=C_4\bmod P\\&=C_1\times C_2\bmod P\\&=((M_1+P\times R1)\times(M_2+P\times R2))\bmod P\\&=(M_1\times M_2+M_1\times P\times R2+M_2\times P\times R1+P^2\times R1\times R2)\bmod P\\&=M_1\times M_2\end{aligned}$$

因此，该算既具有同态加法特性也具有同态乘法特性。

2.6 基于同态加密的检索方法的实现

基于上面的同态加密算法、密钥分发算法中，我们用到的参数主要有：确定的参数 P（P 为密钥，需要用户保存、保密），随机大数 Q、R，以及一对公私钥。

因此我们设计的加密算法，我们设计的检索算法如下：

1）用户在上传同态加密后的文档时，会在文档首部加入用户加密时使用的 Q 与实时产生的随机大数 $Q\times Rt$ 的乘积。

2）用户在进行检索时，要向服务器提供检索需要的关键词密文 Key。其中 Key 的值是通过下面计算得到的：

$$Key=(m_m+P\times R2)\bmod N$$

其中 m_m 是对应的明文中的关键词，Key 是经过同态加密后的结果。

3）服务器后的操作是：

假设已知的密文是：

$$C_i=(M_i+P\times R1)\bmod N$$

则对应于用户提供的关键词：$Key=(m_m+P\times R2)\bmod N$

检索时的操作是：

（1）用用户的公钥解密文章首部的 $Q\times Rt$。

（2）计算：

$$\begin{aligned}res&=((Key-c_i)\times Q\times Rt)\bmod N\\&=(((m_m+P\times R2)-(m_i+P\times R1))\times Q\times Rt)\bmod N\\&=((m_m-m_i)\times Q\times Rt+(R2-R1)\times P\times Q\times Rt)\bmod N\end{aligned}$$

（3）如果 res=0，则匹配，否则，不匹配。

具体分析如下：

因为：$N=P\times Q$，所以

$$((R2-R1)\times P\times Q\times Rt)\bmod N=0,$$

如果 $m_m=m_i$，那么

$$((m_m-m_i)\times Q\times Rt)\bmod N=0,$$

否则

$$((m_m-m_i)\times Q\times Rt)\bmod N>0,$$

由此可知：如果 $m_m=m_i$，那么 res=0；否则 $res>0$。

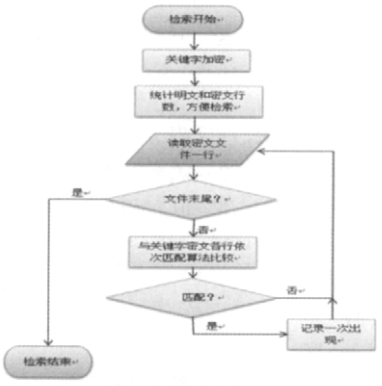

图7 程序流程图

算法优势：

与传统的检索算法针对明文不同，我们采用的基于同态加密的检索算法直接对密文进行操作，这样省去了解密密文的麻烦，提高了效率，同时保证了数据的安全性，对云系统的数据安全和信息处理具有重要意义。

2.7 采用的防攻击技术

由于云存储系统涉及到数据在网络中的传输（上传\下载），为了保障用户数据的安全性，我们必须在数据的传输过程中采用一些防攻击技术来防止在传输的过程中攻击者对传输数据实行篡改、中断等攻击。同时由于数据是存放在不可信的服务器上，攻击者可能通过攻击服务器而获得用户的注册信息，冒充用户登录服务器，获得系统内用户共享的大数 Q 等。

针对用户注册信息的保护：在用户向服务器发送注册请求之后，向服务器发送自己的账号和密码，为了保证用户密码的安全，在服务器端我们存放的不是用户的密码，而是用户密码的 HASH 散列值。用户每次登陆服务器，输入密码后，服务器都要对用户输入的密码进行 HASH 计算，然后比较计算得到的 HASH 值是否与服务器存储的 HASH 一致，若二者一致就成功登陆。对于攻击者而言，即使他通过攻击服务器得到了 HASH 值，但是他使用获得的 HASH 值登陆的时候，服务器会把这个 HASH 散列值进行 HASH 计算，得到了一个新的 HASH 值，这个值与存储在服务器上的显然是不一样的。所以攻击者即使获得了一定的注册信息，也仍然无法冒充用户登陆。

3 性能测试

3.1 同态加密、解密的性能测试

1）测试方案

我们在用户本地实现对明文的加密。客户端程序读取本地明文，获得数字、字符、汉字的 ASCII 编码，并将这些编码通过 Translate() 函数将每个字符转化为 8 位长度的二进制数字，并将每个 8 位长度的二进制数字转化为 CBigInt 类型的大数。然后使用 P 和 Q 对这个 CBigInt 类型的大数进行加密操作。

具体的加密代码如下：

```
tmp = P.Mul(rand());
C = C.Add(tmp);
C = C.Add(Buff);
N = P.Mul(Q);
C = C.Add(N);
C = C.Mod(N);
```

用户从服务器端下载密文，然后在本地解密。客户端程序读取下载到本地的密文，然后通过解密程序获得密文的二进制数据流。然后通过 Translate() 函数将 8 位长度的二进制数字转化为一个字符，从而获得解密后的明文消息。

具体的解密代码如下：

```
Buff.get(buff);
M = Buff.Mod(Ep);
```

2）测试结果（数据）

（1）加密前的明文，如图 8 所示。

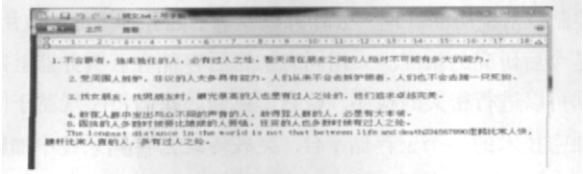

图8 加密前的明文

（2）加密后得到的部分密文，如图 9 所示。

图9 加密后得到的部分密文

（3）解密后得到的明文，如图 10 所示。

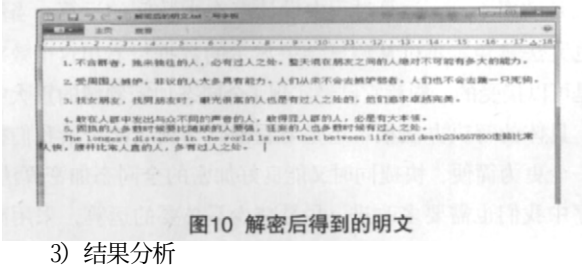

图10 解密后得到的明文

3）结果分析

从结果中我们可以看出，待加密的明文中包含了数字、字符、汉字，而解密之后的明文不管是数字、汉字、字符都和加密之前的明文是一致的。密文是由无规则的 8 进制数字组成，不可能直观地由密文信息推测出明文信息。同态加密成功实现。

3.2 针对该加密算法的检索算法的性能测试

1）测试数据

程序运行结果，如图 11 所示。

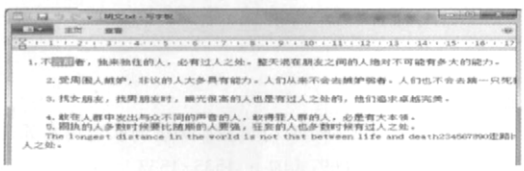

图11 测试数据程序运行结果

2）检索结果为，如图 12 所示。

图12 检索结果示意图

如图 12 所示，密文的第 6 行、第 146 行、第 164 行、第 300 行存储的都是“ long ”对应的密文。而对明文进行分析也能发现，“ long ”在明文中确实出现了 1 次。密文检索成功实现。

4 难点与创新点

4.1 项目的难点

如何使用户高效地对云服务器上的数据进行检测、检索、更新。

数据安全问题是阻碍云存储得到更广泛应用的重要原因。现有的提供云存储中数据安全的方法存在用户负担过大、缺少问责机制等问题，导致难以在实际系统中应用。针对这一问题，本项目将研究采用同态加密算法加密存储在云服务上的数据，从而使用户可以方便地对加密后的数据进行检测、检索、更新。

1）全同态加密算法的实现。这并不是一个简单的问题。算法已经提出，但是算法的实现过程并不容易。我们构造了数据类型 CBigint，并赋予其多种操作，然后才能进入全同态加密算法的初始描述阶段。描述过程中还要钻研其他的算法如 RSA 加密算法，对大数的方法进行修改、补充，以及结构的优化等。最终良好的实现了乘法同态和加法同态，并且运算并不很复杂，应用起来也比较方便。

2）基于此同态加密算法设计的一种检索算法并将其实现。同态加密算法的实现很好地解决了数据的安全存储，同时也为数据的操作提供了良好的保证。在此基础上，我们自行设计了一种密文数据检索方法，可以在保证密文不被解密的情况下快速实现数据的匹配。理论的分析以及算法实现均验证了它的正确性和实用性。综合上述两种算法及其实现可以说是为云系统中的信息安全和数据处理提供了良好的解决方案。

3）整个云平台的搭建。

4.2 项目的创新点

本作品的特色与创新之处主要体现在 ：

1）本作品中对用户的密码的保存采用 MD5 算法。即保存的不是用户的原始密码，而是对用户的密码进行 MD5 算法进行加密后获得的散列序列。优势是在对于攻击者而言，即使他通过攻击服务器得到了 HASH 值，但是他使用获得的 HASH 值登陆的时候，服务器会把这个 HASH 散列值进行 HASH 计算，得到了一个新的 HASH 值，这个值与存储在服务器上的显然是不一样的。所以攻击者即使获得了一定的注册信息，也仍然无法冒充用户登陆。这是云系统平台的需要，并且它也对云系统上的信息安全具有重要意义。

2）实现同态加密算法。同态加密是一种加密形式，它允许人们对密文进行特定的代数运算得到仍然是加密的结果，与对明文进行同样的运算，再将加密结果一样。采用同态加密技术来解决数据处理与隐私保护问题。不需要解密就能对已加密的数据进行处理，实现与对原始数据直接进行处理相同的效果。为解决物联网、云系统等中海量数据信息的安全存储、高效检索以及智能处理等问题提供了新的思路。

3）实现了基于同态加密算法的数据检索。基于同态加密的检索，可以实现密文的快速检索；以往经典的加密算法通常需要对文章进行解密后才能实现内容的检索，而基于同态加密算法的检索则不需要进行解密，可直接对明文进行检索。这样就可以解决物联网中海量数据信息的安全存储、高效检索以及智能处理等问题，保证用户的隐私安全。

5 结束语

5.1 同态加解密及基于同态加密的检索算法

随着云存储技术的快速发展，服务器端数据的安全越来越受到重视。虽然有很多安全性比较好的加密算法，如MD5、SHA系列算法等，但是这些算法都不能很好的表现明文和密文的映射关系，从而难以解决服务器端对密文数据的查找搜索操作。同时现有的多数同态加密算法要么是只对加法同态（例如Paillier算法），要么是只对乘法同态（例如RSA算法），或者同时对加法和简单的标量乘法同态（例如IHC算法和MRS算法）。少数的几种算法同时对加法和乘法同态（例如Rivest加密方案），但是由于严重的安全问题，也未能应用于实际。

本文探讨了应用于云存储系统中的同态加密算法，这种算法既保证了用户数据的安全性，也保证了服务器能够对存储的用户密文进行直接操作，实现了对密文的直接检索，无需像传统的数据处理方法一样先对密文进行解密，也提高了数据处理的效率。平衡了云存储系统保证用户数据安全性和服务于云计算之间的关系。并且在同态加密算法的基础上自行设计并实现了一种检索算法，可以快速在密文中实现字、词、句的检索，保证准确性，效率可以满足要求，为云时代的数据处理、信息安全等提供了良好的解决方案。

云平台方面，可以实现用户的数据存储、注册等基本模拟了现如今大多数云系统的模型，对实际应用有一定的可参考性和实用性，同时具有良好的可扩展性，为项目的完善和功能的增加提供了有利的基础。

5.2 前景与展望

云存储的出现，突破了传统存储方式的性能和容量瓶颈，使云存储提供商能够联结网络中大量各种不同类型的存储设备形成异常强大的存储能力，实现性能与容量的线性扩展，让海量数据的存储成为了可能。从而让企业拥有相当于整片云的存储能力，成功解决存储海量的难题。

被冠以“密码学的圣杯”的称号的“同态加密算法”凭借其高效的检索速率和可靠的安全性，在云技术领域有着很大的前景。同态加密算法的出现对于云存储具有划时代的意义。同态加密领域的突破性进展必将为各行各业的发展带来新的契机，尤其对于目前在全球范围内风生水起的物联网和云计算领域。作为一种新兴的技术，物联网和云系统的安全性能和隐私保护是物联网发展的关键，将直接影响人们对物联网和云系统的接受程度。同时同态加密技术在物联网认证与访问控制、密钥协商管理、云计算以及电子商务等方面的应用，将是今后研究的重点。采用同态加密后存储在云服务器上的数据可以进行密文的检测、检索、更新。我们将实现基于同态加密技术的一个云存储平台，突破传统云存储系统中加密数据进行检测、检索、更新困难等问题。

5.3 项目存在的问题及改进措施

1）时间效率问题。设计一个可以高效的全同态加密算法并不是一个简单的问题。目前也尚没有真正可用于实际的全同态加密算法。现有的多数同态加密算法无法完全满足全同态加密的性质，由于严重的安全问题，也未能应用于实际。即便有提出的全同态加密算法也由于同步工作效率有待改进而未能投入实际应用。根本原因在于全同态加密的属性与效率两者难以兼得，至少在目前是这样。所以我们这个项目的一个弊端是效率不高，运算过程中设计多次大数间的运算，当然这也无法避免。不过从测试结果显示的数据可看出这个效率还是可以接受的，毕竟它已经实现了全同态加密领域的重大突破。具体的解决办法根本还是要看密码学领域能否给我们提供一个更为简便、快捷同时又能良好加密的全同态加密算法。程序中我们也需要多改进，尽量减少不必要的运算，采用高效的算法。比如可以采取更加快捷的检索算法如启发式算法、动态规划算法，而不是单纯的遍历，这样可以提高我们作品的运行效率。

2）空间效率问题。加密过程中设计多次大数、数组等之间的操作，这在存储空间上会有一定的开销。虽然这在如今内存和硬盘大小完全满足的情况下不是主要问题，不过我们还是要尽量优化，比如有的大数组可以采用动态内存申请，或者采用更优的检索算法等。

3）云平台的完善。 （责编 张岩）

参考文献：
[1] 石中盘，蔡萃燕，王显峰 . 面向数据库加密的秘密同态算法的研究 [J]. 计算机应用研究，2009，(04)：1535-1537.
[2] 黄永峰，张久岭，李星 . 云存储应用中的加密存储及其检索技术 [J]. 中兴通讯技术，2010，16（04）：33-36.
[3] 向广利，陈莘萌，马捷，张俊红 . 实数范围上的同态加密机制 [J]. 计算机工程与应用，2005，(20)：12-14.
[4] 孙国梓，董宇，李云 . 基于 CP_ABE 算法的云存储数据访问控制 [J]. 通信学报，2011，(07)：146-152.
[5] 杨勇，方勇，周安民 . 秘密同态技术研究及其算法实现 [J]. 计算机工程，2005，(02)：157-159.
[6] 刘艮，蒋天发 . 同态加密技术及其在物联网中的应用研究 [J]. 信息网络安全，2011，(05)：61-64.
[7] 谷利泽，郑世慧，杨义先 . 现代密码学教程 [M]. 北京：北京邮电大学出版社，2007.