Computer Science and Application 计算机科学与应用, 2013, 3, 127-133
http://dx.doi.org/10.12677/csa.2013.32022 Published Online April 2013 (http://www.hanspub.org/journal/csa.html)

Oblivious Fuzzy Keyword Search Based on Blind GDH Signature*

Fei Han, Jing Qin#

School of Mathematics, Shandong University, Jinan
Email: hanf1987@163.com, #qinjing@sdu.edu.cn

Received: Feb. 11th, 2013; revised: Mar. 1st, 2013; accepted: Mar. 16th, 2013

Abstract: As Cloud Computing becomes prevalent, data privacy has been a bottleneck of Cloud Computing. When using Cloud Computing, users need to implement a large amount of data search. So the security of search is a key point of protecting user's data privacy. This paper mainly concentrates on the searchable encryption scheme. In order to guarantee the security property, a user needs to hide the keywords and the files he/she wants to search. In 2004 Wakaha Ogata and Kaoru Kurosawa proposed a secure searchable encryption scheme, named as oblivious keyword search (OKS), based on Chaum's blind signature. However we find their scheme connecting only one keyword with the data file that is encrypting the data file with its keyword. And then if a data file contains several keywords, we need to repeat the encryption with every keyword. This is quite inefficient. We solve this problem by constructing a new OKS scheme from blind signature based on GDH assumption and connect data files with all its possible keywords. Furthermore, when users are searching by keywords, they might fail to get the desire data files, owing to mismatching the exact keywords. Aiming to solve this problem, we extend our scheme to an oblivious fuzzy keyword search (OFKS) scheme, allowing a user securely search data files through fuzzy keywords. Compare to the original scheme, our scheme possesses more practicality and has been fuzzy keyword supported.

Keywords: Blind GDH Signature; Oblivious Transfer Protocol; Oblivious Keyword Search; Fuzzy Keyword; Oblivious Fuzzy Keyword Search

基于盲 GDH 签名的无记忆模糊关键词搜索*

韩　斐，秦　静#

山东大学数学学院，济南
Email: hanf1987@163.com, #qinjing@sdu.edu.cn

收稿日期：2013 年 2 月 11 日；修回日期：2013 年 3 月 1 日；录用日期：2013 年 3 月 16 日

摘　要：在云计算中，用户在计算过程中的数据安全问题已经成为制约云计算发展的一个瓶颈。本文针对云计算中的加密搜索问题，提出一个有效的加密搜索方案。在搜索过程中，为保证用户的数据安全，用户需要隐藏搜索的关键词以及搜索返回的结果。Wakaha Ogata 和 Kaoru Kurosawa 在 2004 年提出了一个基于盲 RSA 签名的无记忆关键词搜索(OKS)方案，提供了安全的加密搜索，然而该方案中每个数据文件只能关联一个关键词。本文改进了原方案，我们的方案基于盲 GDH 签名，同时使无记忆关键词搜索方案可以在一个数据文件中同时关联多个关键词供用户搜索。此外，用户在搜索的过程中，可能因为关键词不能完全匹配而导致搜索失败，针对这一问题我们将该方案扩展为无记忆模糊关键词搜索(OFKS)方案，使得用户能够对模糊关键词进行正确且安全的搜索操作.相比较原来的方案，我们的方案提供了更有效的搜索方式及对模糊关键词搜索的支持。

*基金项目：国家自然科学基金面上项目(61272091)，山东省自然科学基金一般项目(ZR2012FM005，ZR2011FL027，Y2007A13)。
#通讯作者。

关键词：盲 GDH 签名；OT 协议；无记忆关键词搜索；模糊关键词；无记忆模糊关键词搜索

1. 引言

在当今信息急速膨胀的时代，尽管计算机的处理速度发展很快，但仍然不能满足人们对大量数据进行高速处理的需求。由此，出现了云计算服务。这满足了人们对于高速处理大量数据的要求。然而，云计算还存在很多的安全隐患和实施限制。[1]阐述了云计算在实施和发展过程中需要克服的十大困难。这里，我们致力于解决云计算中的保密问题。众所周知，用户在海量的数据中进行计算需要进行大量的搜索操作。基于安全性考虑，用户不能泄露搜索使用的关键词及搜索结果，同时服务器也不允许用户得到除搜索信息之外的其他信息。因此我们需要一个安全的加密搜索方案以保证两方的安全性。

目前出现的加密搜索方案分为基于公共数据库和秘密数据库两种情况。基于秘密数据库的加密搜索方案，存在分别基于公钥和私钥的加密搜索方案。例如，[2]提出了基于公钥加密的搜索方案——带关键词搜索的公钥加密(PEKS)，该方案以一个邮件系统为实例，发送方在发送邮件的同时发送对应关键词的公钥加密密文 PEKS 以供接收方搜索，接收方生成需搜索的关键词的陷门信息。最后运行验证算法，从而得到搜索的结果。但是该方案的安全实施必须建立在发送方诚实的建立关键词的 PEKS 的基础上。如果发送方为了其非法目的，私自篡改对应邮件的关键词，则接收方无法正确的完成搜索操作。[3,4]提出了基于私钥加密的加密搜索方案。在[3]的协议中，用户将已加密的数据上传到服务器中，同时给出了对应关键词在文档中可能出现的位置。随后，用户使用搜索方案进行顺序扫描，得到所需的搜索结果。该方案的局限性在于，用户将数据上传给服务器以备以后使用，而不是服务器持有原始数据。同时，用户还泄露了一些信息给服务器，如某关键词可能出现的位置。同时，许多更贴近于实际的搜索方案也被提出了，如[5]提出了安全且高效的排序搜索方案，该协议通过引入一个一对多保序映射实现了搜索结果的排序功能，同时加入了相关性检测机制，排序结果的验证机制等功能。[6]将连接词的搜索引入到了可搜索加密中，使用户可以进行多关键词搜索。

对于公共数据库中的安全搜索方案，[7]给出了一个经典的搜索方案——私有数据检索(PIR)。该方案中，用户在两个乃至多个均保存有数据文件的数据库中同时进行搜索操作以保证搜索的保密性。这样可以保证用户搜索过程和返回结果的安全性。然而在搜索过程中，服务器的安全性没有得到保证，通过搜索，用户可能得到多于所需数据的额外信息。同时，该方案需多个服务器同时持有副本，这也限制了方案的实用性。[8]对 PIR 进行了一系列的改进和完善，通过错误校验码的解码算法有效的提升了协议的效率和鲁棒性。

L L. Xu 等[9]引入属性的概念构造了具有访问控制功能的 OT 协议，增强了 OT 协议服务器方的控制能力，增强了协议的灵活度。W. Ogata 等[10]提出了根据 OT 协议构造了一个可行的安全加密搜索方案——无记忆关键词搜索(OKS)。该方案使用盲 RSA 签名以及 OT 协议构造了一个安全的加密搜索方案，在搜索过程中，服务器对每个文件与相关关键词的签名数据进行异或，一起发送给用户，用户对需搜索的关键词进行随机化操作，发送给服务器，服务器对接收的数据签名后返回给用户，用户进行解随机化操作，得到对应关键词的签名，随后进行匹配操作得到搜索的文件。该方案保证了用户和服务器的安全性。然而，该方案存在一定的局限性，在加密过程中，每个密文仅支持一个关键词的加密搜索。若一个文件存在多个关键词，须对该文件进行多次加密。显然，大多数文件都不仅仅存在一个关键词。该方案采用了盲 RSA 签名方案，在该方案中，用户必须选择一个公钥 e 大于模数 n 或者保证$(n, e) = 1$。

本文采用盲 GDH 签名作为搜索过程中的盲化方案，盲 GDH 签名没有上述盲 RSA 签名的安全问题，同时在相同的安全要求下，盲 GDH 签名的签名长度小于盲 RSA 签名的签名长度，这将有效的提高本协议的效率。本文提出的 OKS 协议将一个文件与它的多个关键词整合在一起，使用户可以搜索这个文件中的所有关键词，以验证是否为需要的文件，提高了用户的搜索能力。用户在执行搜索操作时，可能因输入错误或无法确定要搜索的确切关键词，导致所要搜索

的关键词没有包含在服务器保存的关键词集合中，最终搜索失败。本文根据[11]中的模糊关键词搜索算法，使用“编辑距离”定义了一个模糊关键词构造算法，在所提出的 OKS 协议的基础上构造了一个无记忆模糊关键词搜索(OFKS)协议，进一步增强了协议的搜索能力。

2. 预备知识

2.1. 盲 GDH 签名

2.1.1. GDH

在这里我们使用 GDH 的变体 co-GDH[12]。下面简要阐述 co-GDH，详见[12]。

令 G_1， G_2 为两个 P 阶循环群 $\langle g\rangle$。其中 g_1 为 G_1 的生成元，g_2 为 G_2 的生成元。ψ 为 G_1 到 G_2 的同构映射，有 $\psi(g_1)=g_2$。存在以下两个计算困难问题：

Computational co-Diffie-Hellman (co-CDH) problem on G_1，G_2：给定 $g_2, g_2^a \in G_2$， $h \in G_1$ 作为输入，计算 $h^a \in G_1$。

Decisional co-Diffie-Hellman (co-DDH) problem on G_1，G_2：给定 $g_2, g_2^a \in G_2$， $h, h^b \in G_1$ 作为输入，若 $a=b$，则输出 1，否则输出 0。当输出为 1 时，我们称 $\left(g_2, g_2^a, h, h^b\right)$ 是一个 co-Diffie-Hellman 四元组。

我们称一个循环群对 G_1， G_2 为一个(τ,t,ε)-Gap co-Diffie-Hellman 群对(co-GDH group pair)，若满足以下条件：

1) G_1， G_2 的计算及 G_1 到 G_2 的映射 ψ 可以在最多 τ 时间内完成。

2) (G_1, G_2) 上的 co-DDH 问题可以在最多 τ 的时间内解决。

3) 不存在算法可以 (τ,ε)-解决 G_1， G_2 上的 co-CDH 问题。

当 (G_1, G_2) 是一个 (τ,t,ε)-Gap co-Diffie-Hellman 群对时，我们称 G_1 是一个 (τ,t,ε)-Gap co-Diffie-Hellman 群(GDH group)。

2.1.2. 盲签名

盲签名是签名方案的重要组成部分。用户通过选取一个随机数将数据随机化后发送给签名者，签名者对该数据签名后返回给用户，用户通过“解随机化”还原出所需的数据签名。在这一过程中，签名者除了得到了一个有效的签名外，无法获得任何其他的信息，由此保证了用户的数据安全。最早的签名方案是由 Chaum 提出的基于 RSA 的签名方案[13]。在[12]中 Boneh 等给出了一个基于 co-GDH 的盲签名方案。方案如下：

(G_1, G_2) 是一个 (τ,t,ε)-Gap co-Diffie-Hellman 群对。$|G_1|=|G_2|=p$。签名 σ 是 G_1 的一个元素。$H:\{0,1\}^* \to G_1$ 是一个全域 Hash 函数。

该方案由五个算法组成。

1) 密钥生成算法：随机选取 $x \leftarrow_R Z_p^*$，计算 $y \leftarrow g_2^x$。公钥是 $y \in G_2$。私钥是 x。

2) 随机化消息算法：用户选择要签名的消息 M，选择随机数 $r \in_R Z_p^*$，计算 $M' = H(M)\cdot g^r$。

3) 签名算法：给定一个 $x \in Z_p^*$，签名者接受消息 $M' \in \{0,1\}^*$， $\sigma' \leftarrow (M')^x$。得到的签名就是 σ'，发送给用户。

4) 解随机化算法：用户收到 σ'，计算 $\sigma = \sigma' \cdot y^{-r}$，并输出签名对 (M,σ)。

验证算法：对于公钥 $y \in G_2$， $M \in \{0,1\}^*$，签名 $\sigma \in G_1$，计算 $h \leftarrow H(M) \in G_1$，验证 (g_2, y, h, σ) 是一个有效的 co-GDH 四元组。若成功，则签名有效，否则签名失败。

2.1.3. GDH 的安全性

[14]中给出了关于 ct-CDH 问题及假设的定义如下：

定义 1：Chosen-target Computational Diffie-Hellman (ct-CDH)问题和假设：

ct-CDH 问题：令 G 为一个 p 阶群， $x \leftarrow_R Z_p^*$， $y \leftarrow g^x$，从 Hash 函数族中随机选取一个 Hash 函数 $H:\{0,1\}^* \to G$。敌手 B 持有 (p, g, H, y)，并可以访问随机预言机 T_G，可以得到 G 上的随机点 z_i 以及加密预言机 $(\cdot)^x$。令 q_t, q_H 分别为 B 访问随机预言机和加密预言机的次数。

B 攻击 ct-CDH 问题的优势 $ADV_G^{\text{ct-CDH}}(B)$ 定义为以下事件的概率：B 通过访问 q_t 次随机预言机和 q_H 次加密预言机后 $(q_H < q_t)$。成功输入了 1 个有效的签名，及生成了一个元素的集合 $V((v_1, j_1), \cdots, (v_l, j_l))$，其中，对所有的 $1 \le i \le l$，有 $1 \le j_i \le q_t$，使得 $v_i = (z_{j_i})^x$，v_i 各不相同。

ct-CDH 假设是指不存在多项式时间能力的敌手

B 能够以不可忽略的优势 $ADV_G^{\text{ct-CDH}}(B)$ 解决 ct-CDH 问题。在此，GDH 问题就包含在 CDH 问题中，所以我们可以将GDH的安全性规约到ct-CDH问题的安全性上。

定义 2：我们称 ct-CDH 问题是困难的，若对于具有多项式时间计算能力的敌手 B，它的优势 $ADV_G^{\text{ct-CDH}}(B)$ 是可忽略的。

定理 1：如果在群 G 上的 ct-CDH 问题是困难的，则 G 上的 GDH 问题也是困难的，那么我们称 GDH 盲签名是安全的，即可以抵抗选择消息攻击下的再一次伪造攻击。

3. 无记忆关键词搜索(OKS)

我们将提出一个基于盲 GDH 签名的 OKS 协议，该协议使得用户可以搜索一个文档中的所有关键词。我们假设服务器持有一些保密数据供用户进行关键词搜索。为保证安全性，用户不允许服务器知晓自己要搜索的关键词和返回的数据文件。

一个 OKS_k^n 是一个服务器 S 和用户 U 之间的两方协议。在这个协议中，用于与服务器在由 n 个数据组成的数据库集合中进行 k 次搜索。令 $(c_1,\cdots,c_n)$ 为保密数据集合，W 为保密数据中关键词的集合，对于每个数据文件 c_i，其包含的关键词集合为 $(w_1^i,w_2^i,\cdots)\in W$。

承诺阶段，S 承诺了 n 个数据 $B_1,\cdots,B_n$，其中 $B_i=\left((w_1^i,w_2^i,\cdots),c_i\right)$，在这一阶段，服务器须根据 i 生成一个随机数 p_i，以保证 c_i 被安全加密。

传输阶段，包括 k 次传输。每次传输过程中，U 选择关键词 $w_l^*\ (1\le l\le k)$，通过交互式运算得到 $\text{Search}(w_l^*)$。其中定义 $\text{Search}(w_l^*)=\{i\,|\,w_j^i=w\}$。

选择阶段，用户 U 根据传输阶段返回的 $i=\text{Search}(w)$，与服务器 S 进行一个基于 OT_1^n 协议的交互操作，即，S 持有 n 个数据 $p_1,\cdots,p_n$，U 持有 i，执行 OT_1^n 协议，从 S 中获取 p_i，而对其他 $p_{i,j\ne i}$，同时，S 不能获取 i 的值，然后解密得到需要的搜索结果 c_i。

在这一过程中，用户没有得到额外的信息，而数据提供者也不能得到关于关键词 $w_1^*,\cdots,w_k^*$ 的信息。更形式化的定义协议的安全性如下：

用户的安全性：我们称该协议对于用户是安全的，若对于任意恶意的数据提供方 S'，$(w_1^*,\cdots,w_k^*)\ne(w_1,\cdots,w_k)$，$S'$ 对 $(w_1^*,\cdots,w_k^*)$，$(w_1,\cdots,w_k)$ 运行的视图计算不可区分的。

数据提供方的安全性：引入一个理想情况 Ideal World 与实际情况 Real World 进行对比。理想情况中，有一个可信第三方(TTP)收到 $(B_1,\cdots,B_n)$。然后将用户的搜索结果 $\text{Search}(w_l^*)$ 返回给用户。

定义 3：我们称一个协议对于数据库是 $\varepsilon(l)$-安全的，若对于任意恶意用户 U，存在一个模拟器 A 在 Ideal World 中充当用户的角色，使得对于任意多项式时间区分器 D，

$$\left|\Pr\left(D\left(U's\ \text{output}\right)=1\right)-\Pr\left(D\left(A's\ \text{output}\right)=1\right)\right|<\varepsilon(l)$$

若 $\varepsilon(l)$ 是一个可忽略函数，则这个协议对于数据库是安全的。

定义 4：我们称一个 OKS_k^n 协议是安全的，若该协议对于用户和服务器都是安全的。

3.1. 基于盲 GDH 签名的 OKS 协议

令 G 为一个伪随机数生成器，H_1，H_2 为两个 Random Oracle。全局信息 $I=(p,g,H_1,G)$，H_2 由服务器私有，私钥 $sk=x$，公钥 $pk=(p,g,H_1,G,y)$，$y=g^x$。

承诺阶段：对于 c_i 中的每个关键词 w_j^i，计算 $K_{i_j}=H_1\left(w_j^i\right)^x$，同时选择一个 Hash 函数 H_2，计算 $p_i=H_2(i)\ (1\le i\le n)$，生成 $(p_1,\cdots,p_n)$，随后对所有的文件进行承诺，计算 $E_i=\left(G(p_i)\oplus c_i\right)\|K_{i_1}\|K_{i_2}\|\cdots$，将 $E_1,E_2,\cdots,E_n$ 发送给用户。

传输阶段：第 l 次传输中，用户收到 $E_1,E_2,\cdots,E_n$ 后，选择一个随机数 $r\in_R Z_p^*$，对于要搜索的关键词 w_l^*，计算 $K_l=H_1\left(w_l^*\right)\cdot g^r$，将 K_l 发送给服务器。服务器计算 $K_l'=(K_l)^x=H_1\left(w_l^*\right)^x\cdot y^r$，将 K_l' 发送给用户。用户计算 $K_l^*=K_l'\cdot y^{-r}$，得到关于 w_l^* 的签名 K_l^*。将收到的 $E_i=E_i'\|K$ 拆分，得到对应每个 c_i 的所有关键词的签名 $K_{i_1},K_{i_2},\cdots$，使用 K_l^* 进行匹配，对于任意的 i，若 $K_l^*=K_{i_j}$，则确定需要的文件为 c_i，并得到密文 E_i'。

选择阶段：引入一个安全的 OT_1^n 协议，用户与服务器进行安全的交互运算。用户将所需文件的序号 i 发送给服务器，服务器从生成的 $p_1,\cdots,p_n$ 中，选择 p_i 发送给用户。由于使用 OT_1^n 协议，服务器无法得知用户需要那个文件，而用户也无法得到除了 p_i 之外的其他文件。用户得到 p_i 后，计算 $G(p_i)$，进而计算 $c_i=E_i'\oplus G(p_i)$ 得到模糊关键词匹配的文件 c_i。

基于盲 GDH 签名的 OKS 协议框架见图 1。

3.2. OKS 协议的安全性

正确性：每次传输状态返回的 i 值正确的概率至少为 $1-\frac{n}{2^l}$，基于 OT_1^n 的安全性，我们得到搜索正确的概率至少为 $1-\frac{n}{2^l}$。

用户的安全性：S 无法得到关于 $w_1^*,\cdots,w_k^*$ 的任何信息，因为这些关键词在盲 GDH 方案中被盲化了。

数据库的安全性：我们在假设 ct-CDH 问题是困难的情况下使用 Real World/Ideal World 方法证明数据库的安全性。假设存在一个恶意用户 $\tilde{U}$，$\tilde{U}$ 向 S 提出 k 次查询请求，在 Ideal World 中假设存在一个模拟器 A。承诺阶段，A 生成 p,g,H_1,H_2,y,x，并将 p,g,H_1,y 发送给 $\tilde{U}$，同时，A 随机生成 $E_1,E_2,\cdots,E_n$，也发送给 $\tilde{U}$。传输阶段，A 进行与 S 相同的操作，最后 A 将 $\tilde{U}$ 的输出作为自己的输出。

A 模拟 H_1：若 $\tilde{U}$ 向 H_1 查询 w，则 A 随机选择 y_w，$y_w=H_1(w)$。

A 模拟 H_2：若 $\tilde{U}$ 向 H_2 查询 i，则 A 随机选择 p_i，使得 $p_i=H_2(i)$。

A 模拟 G：假设 $\tilde{U}$ 先向 H_1，H_2 提出查询请求，随后向 G 发送对 p_i 的请求。令 $cnt=0$，$QA\text{-}list$ 列表为空。

假设 $\tilde{U}$ 首次向 G 查询 p_i。

1) 若 $K\neq H(w)^x \bmod p$，则 A 随机选择一个 E^*，使得 $E^*=G(p_i)$。

2) 若 $K=H(w)^x \bmod p$，则

若 w 在 $QA\text{-}list$ 列表中，则直接运行 4)。否则令 $cnt=cnt+1$。

3) 若 $cnt\geq k+1$，则 A 将一个随机值赋给 $G(p_i)$。否则 A 向可信第三方(TTP)发出关于 w 的 Search 请求，并得到结果 $\text{Search}(w)$，A 将 $(w,\text{Search}(w))$ 添加到 $QA\text{-}list$ 列表中。

4) 假定关键词 w 已存在于 $QA\text{-}list$ 中，即 $(w,\text{Search}(w))\in QA\text{-}list$。

若 $i=\text{Search}(w)$，则 A 设置 $G(p_i)=E_i'\oplus c_i$，否则，A 随机选取一个值赋给 $G(p_i)$。

令 $Fail$ 为 $cnt\geq k+1$ 的时间。若 $Fail$ 不发生，则 A 可以很好的模拟 G 实现协议运算。而 $Fail$ 失败的概率 $Pr(Fail)$是 $\tilde{U}$ 能够成功的进行关于盲 GDH 签名的基于选择明文攻击的再一次伪造攻击的概率。由定理 1，我们知道若 ct-CDH 问题是困难的，则盲 GDH 签名是安全的，从而若 ct-CDH 是困难的，则 A 的输出与 $\tilde{U}$ 的输出是计算不可区分的。这就证明了上述 OKS 是安全的。

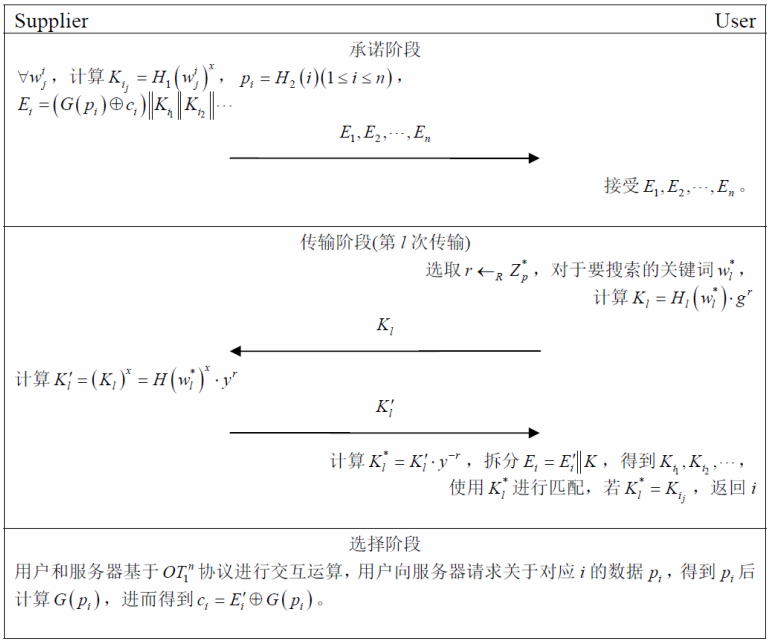

Figure 1. The framework of OKS scheme
图 1. OKS 协议框架

4. 无记忆模糊关键词搜索(OFKS)

用户在实施搜索的过程中，不可避免的会出现输入错误，或不能确定要搜索的确切关键词。因此我们引入模糊关键词的概念来完善 OKS 协议。我们使用 Edit Distance 来定义关键词 w 的模糊度 d，生成的模糊关键词集合记为 $S_{w,d}$。Edit Distance 指，由字符串 A 转换到字符串 B 所经过的最小的字符操作数，字符操作包括：在 A 中增加一个字符，在 A 中删去一个字符，将 A 的某个字符转换成另一个字符。如关键词 word 与关键词 ward 的模糊度为 $d = 1$，word 与 wrd 的模糊度也为 $d = 1$。若 $d = 0$ 时，显然 OFKS 协议即上述的 OKS 协议。假设我们需要搜索关键词 word，根据模糊关键词集合构造算法(见附录)可以生成一个关于 word 的模糊关键词集合 $S_{\text{word},1} = \{\text{word}, \text{ord}, \text{wrd}, \text{wod}, \text{wor}\}$。然后对生成的关键词集合运行 OFKS 协议。

4.1. OFKS 协议

协议背景与上述 OKS 协议类似。假设服务器持有秘密数据供用户搜索。秘密数据集合为 $c_1, c_2, \cdots, c_n$，其中每个 c_i 中包含的关键词集合为 $\left(w_1^i, w_2^i, \cdots\right) \in W$，每个关键词 w_j^i 生成的模糊关键词集合为 $S_{w_j^i} = \left\{w_{j1}^i, w_{j2}^i, \cdots\right\}$，则对于每个 c_i，对应所有关键词的模糊关键词集合为 $S_{c_i} = \bigcup_{w_j^i \in W} S_{w_j^i}$。假设 G 为一个伪随机数生成器，H_1，H_2 为两个 Random Oracle。全局信息 $I = \left(p, g, H_1, G\right)$，$H_2$ 由服务器私有，私钥 $sk = x$，公钥 $pk = \left(p, g, H_1, G, y\right)$，$y = g^x$，服务器保留 H_2。

承诺阶段：预先协商用户与服务器搜索过程的关键词的模糊度 d。此处我们假设 $d = 1$，当 $d > 1$ 时类似。服务器使用下述的算法对每个关键词 w_j^i 生成相关的模糊关键词集合 $S_{w_j^i} = \left\{w_{j1}^i, w_{j2}^i, \cdots\right\}$，对于每个 w_{jh}^i 计算 $K_{jh}^i = H_1\left(w_{jh}^i\right)^x$，同时选择一个 Hash 函数 H_2，对于每个 i，计算 $p_i = H_2(i)$，生成 $\left(p_1, \cdots, p_n\right)$，随后对所有的文件进行承诺，$E_i = \left(G\left(p_i\right) \oplus c_i\right) \| K_{11}^i \| K_{12}^i \| \cdots \| K_{j1}^i \| K_{j2}^i \| \cdots$，将 $E_1, E_2, \cdots, E_n$ 发送给用户。

传输阶段：在第 l 次传输，用户收到 $E_1, E_2, \cdots, E_n$ 后，选择一个随机数 $r \leftarrow_R Z_p^*$，对于要搜索的关键词 w_l^*，使用模糊关键词构造算法[8]，计算关于 w_l^* 的模糊关键词集合 $S_l^* = \left\{w_{l1}, w_{l2}, \cdots\right\}$，分别计算 $K_{lk} = H_1\left(w_{lk}\right) \cdot g^r$，将 $K_{l1}, K_{l2}, \cdots$ 发送给服务器。服务器分别计算 $K_{lk}' = \left(K_{lk}\right)^x = H_1\left(w_{lk}\right)^x \cdot y^r$，将所有的 K_{lk}' 发送给用户。用户计算 $K_{lk}^* = K_{lk}' \cdot y^{-r}$，得到关于 w_l^* 的模糊关键词的签名 $K_{l1}^*, K_{l2}^*, \cdots$，将收到的 $E_i = E_i' \| K$ 拆分得到对应每个才 c_i 的所有模糊关键词 $K = K_{i1}, K_{i2}, \cdots, K_{l1}, K_{l2}, \cdots$，用 w_l^* 的模糊关键词的签名 $K_{l1}^*, K_{l2}^*, \cdots$ 与上述的 $K_{11}^i, K_{12}^i, \cdots, K_{j1}^i, K_{j2}^i, \cdots$ 进行匹配，对于任意的 i，若有 $K_{lk}^* = K_{jh}^i$，则得到需要的文件 E_i'。

选择阶段：这里采用一个安全的 OT_1^n 协议，用户与服务器进行安全的交互操作。用户提供给服务器想要得到的数据 p_i，服务器从生成的 $\left(p_1, p_2, \cdots, p_n\right)$ 中，选择 p_i 发送给用户。由于使用 OT_1^n 协议，服务器无法得知用户需要那个文件，而用户也无法得到除了 p_i 之外其他的文件。用户得到 p_i 后，计算 $G\left(p_i\right)$，进而计算 $c_i = G\left(p_i\right) \oplus E_i'$ 得到根据模糊关键词搜索得到的文件 c_i。

4.2. OFKS 的安全性

定理 2：假设 OKS 协议是安全的，则 OKFS 协议也是安全的。

证明：OFKS 协议相比 OKS 协议，仅仅增加了关键词的数目，即增加了模糊关键词，因此 OKS 协议的安全性证明同样可以证明 OFKS 协议的安全。

5. 总结

本文首先介绍了基于盲 GDH 签名的 OKS 协议，在承诺和传输阶段，用户和服务器分别对其关键词进行盲签名，然后用户对关键词的签名进行匹配得到所需数据文件的序号，在选择阶段，引入了 OT 协议，使得用户与服务器可以安全的进行交互式运算，通过 OT 协议的安全性质，用户只能得到序号对应的文件，而服务器也无法得到用户的信息。因此该协议能够保密的完成加密搜索。同时，该协议将一个数据文件的多个关键词整合在一起，使用户可以在数据文件同时关联多个关键词的情况下对单个关键词进行加密搜索。另外，基于用户的搜索习惯，增加了对模糊关键词搜索的支持，提出了关于模糊关键词搜索的 OKS 协议，该协议能够有效的进行对模糊关键词的加密搜索，扩展了协议的使用范围。

参考文献 (References)

[1] A. Fox, R. Griffith. Above the clouds: A Berkeley view of cloud

computing. Technical Report No. UCB/EECS-2009-28, University of California, Berkeley, 2009, 28.
[2] D. Boneh, G. Di Crescenzo, R. Ostrovsky, et al. Public key encryption with keyword search. In: Advances in Cryptology-Eurocrypt 2004. Berlin/Heidelberg: Springer, 2004: 506-522.
[3] D. X. Song, D. Wagner and A. Perrig. Practical techniques for searches on encrypted data. Proceedings of IEEE Symposium on Security and Privacy, Berkeley, 14-17 May 2000: 44-55.
[4] Y. C. Chang, M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. ACNS'05 Proceedings of the 3rd International Conference on Applied Cryptography and Network Security, Springer-Verlag, Berlin/Heidelberg, 2005: 442-455.
[5] C. Wang, N. Cao, K. Ren, et al. Enabling secure and efficient ranked keyword search over outsourced cloud data. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(8): 1467-1479.
[6] Z. Chen, C. Wu, D. Wang, et al. Conjunctive keywords searchable encryption with efficient pairing, constant ciphertext and short trapdoor. Intelligence and Security Informatics, 2012, 7299: 176-189.
[7] B. Chor, O. Goldreich, E. Kushilevitz, et al. Private information retrieval. Proceedings of IEEE 36th Annual Symposium on Foundations of Computer Science, Milwaukee, 23-25 October 1995: 41-50.
[8] C. Devet, I. Goldberg and N. Heninger. Optimally robust private information retrieval. Proceedings of the 21st USENIX conference on Security symposium, USENIX Association, 2012: 16.
[9] L. L. Xu, F. G. Zhang. Oblivious transfer with threshold access control. Journal of Information Science and Engineering, 2012, 28(3): 555-570.
[10] W. Ogata, K. Kurosawa. Oblivious keyword search. Journal of Complexity, 2004, 20(2): 356-371.
[11] J. Li, Q. Wang, C. Wang, et al. Enabling efficient fuzzy keyword search over encrypted data in cloud computing. Proceedings of IEEE INFOCOM'10 Mini Conference, San Diego, 2009: 593.
[12] D. Boneh, B. Lynn and H. Shacham. Short signatures from the Weil pairing. Advances in Cryptology—ASIACRYPT 2001, 2001: 514-532.
[13] D. Chaum. Blind signatures for untraceable payments. Advances in Cryptology: Proceedings of Crypto, 1982, 82: 199-203.
[14] A. Boldyreva. Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme. Public Key Cryptography—PKC 2003, 2002, 2567: 31-46.

