一個於辮群上的可比較之可搜尋加密演算法

陳國璋　　洪國寶　　陳昱圻　　劉兆樑
中興大學資訊科學與工程學系　　亞洲大學資訊多媒體應用學系
{s9756013, gbhorng, s9756034}@cs.nchu.edu.tw　　jliu@asia.edu.tw

摘要

在傳統關鍵字可搜尋加密法中，大多無法支援關鍵字的範圍搜尋。這也意味使用者無法依照自己所需之條件來取回符合的文件，在彈性上略有不足。最早由 Song 等學者提出範圍查詢於可搜尋加密法，利用樹(Tree)之結構跟超矩陣(Hyper-Rectangle)的觀念來架構。之後，Boneh 等學者進一步的將範圍查詢擴展為條件搜尋，這包含三種條件：範圍查詢、子集查詢與比較查詢，並提出隱藏向量加密法，使用者將所需條件建立一個預測向量，讓伺服器進行想要的查詢。本文提出一種支援比較查詢的可搜尋加密系統，透過辮群(Braid group)與順序可嵌性函數(Order-embedding function)的特性，來建構一種可支援比較查詢的可搜尋加密系統。我們改進前人的向量轉換方式與其複雜的設計，減少計算與空間成本，並提供一個更有效的方法。

關鍵詞：可搜尋加密法、辮群、範圍搜尋、條件搜尋。

Abstract

Most keyword-searchable encryption schemes are unable to provide range query of keywords. Song *et al.* first proposed a keyword-searchable encryption based on the conception of tree structure and hyper-rectangle. Boneh *et al.* proposed the hidden vector encryption (for short, HVE) which has ability to supply the search for range query, subset query, and comparison query. A user needs to construct the predicate vector for the server to test. This paper presents a comparable keyword-searchable encryption scheme based on braid group and order-embedding function. The proposed scheme has the capability of searching for comparison query. Construction is more efficient than previous schemes, and needs less computation and communication.

Keywords: Searchable Encryption, Braid group, Range Query, Conditional Query.

1. 前言

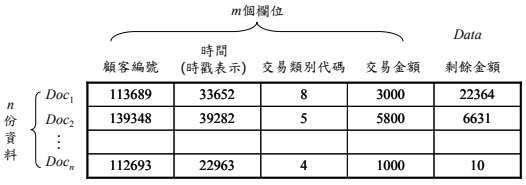
近年來資訊與網路相關研究進步神速，以及網路的普及性與方便性，漸漸融入人們日常生活。因此，大家都習慣在個人電腦上進行資料的處理與儲存以及在網路上發表，這可以保有資料的完整性，也方便人們取得。最近個人網路平台崛起，如 Facebook、無名小站、Plurk 等，大家不自覺地將含有個人隱私資料在網路上公開。因此，保護資料隱私性與易取得性之重要性日益遽增。

最簡單的保護方式，就是將資料作加密。加密後的資料儲存在遠端的公開伺服器上。當使用者要取回資料時，透過伺服器把加密資料取回後，再解密所有資料，進而從中挑選使用這想要的部份。但是，這取回的數量過大且非常不便利挑選。雖然這方式相當有效地防範惡意伺服器與非法使用者的攻擊，不過隨著時間的過去，存放在伺服器上的資料也會快速增長，一次取回所有文件的作法相當浪費網路成本與暫存空間，特別是在使用者端能用資源是極度有限設備下，如手機、PDA 等，這作法是相當不明智的。比較好的方式，讓使用者在加密資料上進行搜尋，再取回想要的資料。

最早由 Song 等學者提出在加密資料上作搜尋的方法，利用對稱式金鑰來建構，使用者利用關鍵字建立屬於自己的暗門，利用暗門在伺服器上進行加密資料搜尋。這方面的研究，我們稱為關鍵字搜尋。

在傳統關鍵字可搜尋加密法中[1, 6, 12, 13, 14]，大多無法支援關鍵字的範圍搜尋。這也意味使用者無法依照自己所需之條件來取回符合的文件，在彈性上略有不足。當一個加密演算法有支援條件式搜尋，這讓應用空間變得更廣泛，如應用在電子信件伺服器(E-mail server)上或金融機構。以金融交易記錄(Financial audit log)為例，一筆紀錄有四個欄位，分別是顧客編號、時間、交易類別編號與交易金額，資料為剩餘金額。以金融交易記錄為例，如圖一，當帳目管理人員想要知道特定顧客(等於某顧客編號)在某個時間點過後(小於某個時戳值)，進行提款服務項目(等於某個交易類別編

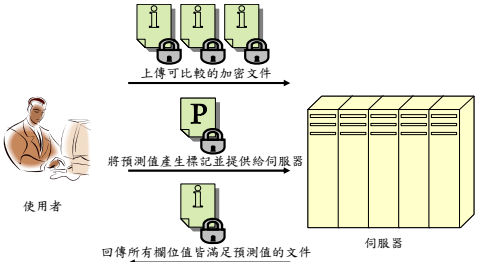
號)，其交易金額為小額服務(小於某特定金額)的資料。有了我們所提出的加密機制，可以讓管理人員方便且安全地取得資料。



圖一：以金融交易記錄為例。

首先提出這方面研究的 Song[3, 14]等學者，他們利用超矩形(Hyper-Rectangle)的想法，將多個欄位抽象化成一個多維度的矩形，資料的欄位值抽象化成多維度空間裡的一個點，伺服器在搜尋時透過超矩形與點來判斷這份資料是否有滿足使用者需求。將範圍搜尋的研究一般化以及更廣泛應用，即為條件式搜尋，這提供更多選擇給使用者。此概念是由Boneh[2]等學者提出隱藏向量加密法(Hidden Vector Encryption, HVE)，應用範圍相當廣泛。他們將條件化搜尋分成三種，分別是範圍式查詢、子集式查詢、比較式查詢。使用者將想要的預測條件轉換成預測向量，資料的實際欄位值也轉換成向量，伺服器透過這兩向量來進行查詢。

我們的目的，讓使用者上傳加密資料到伺服器，當使用者想取回某些特定欄位值的資料，利用預測值產生標記(Token)。伺服器收到使用者提供的標記，開始進行比對，並將所有滿足預測值的資料傳回給使用者。其互動流程參照圖二。一般而言，當資料經過加密後，會喪失數值的順序性。而我們提出的機制，必須要有下列條件：一份未加密且所有欄位皆滿足使用者預測的資料，經過加密後，加密資料的欄位值仍需滿足加密後的預測值。我們必須確保這機制的正確性與安全性。在本文中，我們著重在條件搜尋中的比較式查詢。資料加密時，實際欄位值透過順序可嵌性函數轉換；當使用者要搜尋時，要決定各欄位的預測值，同樣透過順序可嵌性函數轉換，這轉換過程中並不會遺失掉數值的順序性。這種作法可以很明顯地降低計算成本，也大幅度降低儲存空間。



圖二：使用者與伺服器的互動流程。

本文第二部分簡單描述問題定義與系統架構，第三部分簡單介紹辮群與順序可嵌性函數等基礎知識，第四部分是我們的方法。在第五部分進行安全性分析。最後為本文結論。

2. 系統架構

假設使用者的儲存資料格式：$Doc = (d_1, d_2, \ldots, d_m \| M)$。一筆資料包含 m 個欄位，每個欄位皆可以編碼化成一個整數區間$[1,T_i]$，$\forall\, i = 1, 2, \ldots, m$。每個欄位值 d_i 皆介在該欄位編碼後的整數區間$[1,T_i]$中，$\forall\, i = 1, 2, \ldots, m$。$E(M)$是訊息 M 經過一個安全的對稱式加密法產生的密文。

在產生標記階段，使用者提出的預測格式：$Pred = (co_1 \| p_1, co_2 \| p_2, \ldots, co_m \| p_m) = (p_1, p_2, \ldots, p_m) \| Comp$。一份預測包含 m 個比較條件與 m 個預測值，其中比較條件有四種情況：大於(>)、小於(<)、等於(=)、不考慮(×)；預測值 p_i 皆介在所屬欄位編碼後的整數區間$[1,T_i]$中，$\forall\, i = 1, 2, \ldots, m$。

我們提出一個使用辮群與順序可嵌性函數來達成可比較的可搜尋式加密演算法，這架構在非公開金鑰加密系統之下。此演算法具有四個隨機多項式時間演算法：

(1) SETUP(λ)：輸入安全參數 λ，產生公開參數 PP 與私密參數 SP。
(2) ENC(PP, Doc)：輸入公開參數 PP 與資料 Doc，產生密文 C。
(3) TOKEN(SP, $Pred$)：輸入私密參數 SP 與使用者的預測 $Pred$，產生可讓伺服器進行比對的標記(Token)T。
(4) QUERY(C, T)：輸入密文 C 與標記 T。當密文 C 的欄位值都符合標記 T 的預測值，則輸出"Yes"並回傳 $E(M)$；否則，輸出"No"並結束查詢。

3. 基礎知識

辮群最早是由 Artin 學者，在 1925 年所提出，在 1974 年之後才廣泛地被運用。在物理、數學、計算機科學等領域，皆有相當多的應用。以下簡單介紹辮群的基本定義、特性[11]與已知難題[9]。

定義一：辮群，B_n。辮群是個無限非交換群。B_n，$n \geq 3$，為一辮群，n 為辮標，則 B_n 是由$(n\text{-}1)$個初等辮所組成，分別記作 $\sigma_1, \sigma_2, \ldots, \sigma_{n-1}$。須滿足該性質：$B_n = \{\sigma_1, \sigma_2, \ldots, \sigma_{n-1} : \sigma_i\sigma_j = \sigma_j\sigma_i \text{ , } |i - j| \geq 2;\ \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1},\ o.w.\}$。

定義二：正辮。一個辮子 b 稱作正辮，若

且唯若，b 可以由某些初等辮的乘積所表示，且乘積中所有的初等辮皆沒有負冪次方。單位辮(Identity)ε，定義為一正辮。所有正辮的集合為辮群的半群(Semi-group)，記作 B_n^+。

定義三：(左/右)偏序關係($\leq, \leq_L, \leq_R$)。假設兩正辮 $v, w \in B_n$，如果 $v \leq w$ ($v \leq_L w$ 或 $v \leq_R w$)，則存在兩個正辮 $a, b \in B_n$，使得，$w = avb$ ($w = vb$ 或 $w = av$)。

定義四：左辮群，LB_n。

$LB_n = \{\sigma_1, \sigma_2, ..., \sigma_{\lfloor n/2 \rfloor - 1}\}$，為 B_n 子群。

定義五：右辮群，RB_n。

$RB_n = \{\sigma_{\lfloor n/2 \rfloor + 1}, ..., \sigma_{n-2}, \sigma_{n-1}\}$，為 B_n 子群。

特性一：當 $a \in LB_n$，$b \in RB_n$，則 a 跟 b 具有交換性(Commutativity)，意即，$ab = ba$。

定義六：辮群上的已知難題。Root Problem(RP)與 Variant Root Problem(VRP)。

(1) RP：給定 $\beta \in B_n$，$c \in N$，c>1，要找 $\gamma \in B_n$，使得 $\beta = \gamma^c$ 是困難的。

(2) VRP：給定 $\beta \in B_n$，$c \in N$，c>1，要找 $\alpha \in B_n$，$x \in N$，使得 $\beta = \alpha^{xc}$ 是困難的。

有相關研究指出可以在多項式時間內破解其共軛特性[5, 7, 8]，而且我們提出的機制並沒使用共軛特性，在此不介紹辮群上有關共軛特性與相關難題。

簡單介紹函數與順序關係。

定義七：函數與順序關係。假設 $f: P \rightarrow Q$。

(1) 單調或順序保持性。如果在 P 中，$a \leqq b$，則在 Q 中，$f(a) \leqq f(b)$。

(2) 順序反身性。如果在 Q 中，$f(a) \leqq f(b)$，則在 P 中，$a \leqq b$。

(3) 反調或順序反轉性。如果在 P 中，$a \leqq b$，則在 Q 中，$f(b) \leqq f(a)$。

(4) 順序嵌入性。函數 f 是單調且順序反身性函數。如果在 P 中，$a \leqq b$，若且唯若，在 Q 中，$f(a) \leqq f(b)$。一般而言，函數 f 必須要為一對一函數。

4. 我們的方法

我們提出一個基於辮群與順序可嵌性函數的可比較可搜尋加密法。4.1.節為提出的機制，4.2.節為安全性分析與討論。

- 4.1. 提出的機制

1. SETUP(λ)：

(1) 產生 2 個辮群 B_n, B_p，其中 $n \leqq p$，則 $B_n \subseteq B_p$。n 跟 p 皆夠大到可以抵抗辮群上的長度攻擊。

(2) 產生一個順序嵌入性函數 $OE: N \rightarrow B_n$ 定義為 $OE(x) = \delta^x$，其中 δ 為一隨機辮子，保持 δ 私密。

(3) 產生一個單向函數 $F: B_n \rightarrow B_n$ 定義為 $F(\gamma) = \gamma^c$，其中 c 為一隨機整數，保持 c 私密。

(4) 合成 OE 與 F，得 $F \circ OE: N \rightarrow B_n$ 為 $F(OE(x)) = F \circ OE(x) = (\delta^x)^c$，此為順序嵌入性單向函數，記作 $FOE(x)$。

(5) 產生左基本雜訊辮(Left base noise braid)，記作 $LBN \in LB_p$。產生方式如下：在 LB_p 中找一條由連續的 p-braid 相乘的正辮，長度為 $r \geq (m+1)$。表示法為 $LBN = \sigma_i \sigma_{i+1} \ldots \sigma_{i+r}$, 其中 σ_i, $\sigma_{i+1}, \ldots, \sigma_{i+r} \in LB_p$。

(6) 產生右基本雜訊辮(Right base noise braid)，記作 $RBN \in RB_p$。產生方式如下：在 RB_p 中找一條由連續的 p-braid 相乘的正辮，長度為 $t \geqq (m+1)$。表示法為 $RBN = \sigma_j \sigma_{j+1} \ldots \sigma_{j+t}$, 其中 σ_j, $\sigma_{j+1}, \ldots, \sigma_{j+t} \in RB_p$。

(7) 根據 LBN，在 LBN 之中的任意相鄰的 p-braid 進行交換，作為左欄位雜訊(Left field noise)，此步驟進行 m 次，依序得到 $LFN_1, LFN_2, \ldots, LFN_m$。表示法為 $LFN_i = \sigma_i \sigma_{i+1 \ldots} \sigma_{ik+1} \sigma_{ik \ldots} \sigma_{i+r}$, 其中 $i \leqq i_k \leqq i+r$。

(8) 根據 RBN，在 RBN 之中的任意相鄰的 p-braid 進行交換，作為右欄位雜訊(Right field noise)，此步驟進行 m 次，依序得到 $RFN_1, RFN_2, \ldots, RFN_m$。表示法為 $RFN_i = \sigma_j \sigma_{j+1 \ldots} \sigma_{jl+1} \sigma_{jl \ldots} \sigma_{j+t}$, 其中 $j \leqq j_l \leqq j+t$。

(9) 隨機挑選 q 個 n-braid，$\mu_1, \mu_2, \ldots, \mu_q$。

(10) 計算 $\mu = \mu_1 \mu_2 \ldots \mu_q$；計算 $\mu^{-1} = \mu_q^{-1} \ldots \mu_2^{-1} \mu_1^{-1}$。

(11) 隨機挑選 u 個 p-braid，$\theta_1, \theta_2, \ldots, \theta_u$。

(12) 計算 $\theta = \theta_1 \theta_2 \ldots \theta_u$；計算 $\theta^{-1} = \theta_u^{-1} \ldots \theta_2^{-1} \theta_1^{-1}$。

(13) 隨機挑選 m 個整數，$a_1, a_2, \ldots, a_m$。

(14) 計算公開參數 $\alpha_i = \mu^{a_i}, \forall i = 1, 2, ..., m$。

(15) 計算私密參數 $\beta_i = (\mu^{-1})^{a_i}, \forall i = 1, 2, ..., m$。

(16) 產生 m 個 noise function，記作 N_i：$N \rightarrow B_p$，$\forall i = 1, 2, ..., m$。定義為 $N_i(x) = \theta^x \cdot \alpha_i$。

(17) 產生 m 個 partial noise inverse function，記作 IN_i：$N \rightarrow B_p$，

$\forall i=1,2,...,m$。定義為 $IN_i(x)=\alpha_i\cdot(\theta^{-1})^x$。

(18) 產生 m 個 noise function for ENC，記作 $EN_i: B_n\times B_n\rightarrow B_p$，$\forall i=1,2,...,m$。定義為 $EN_i(a,b)=\alpha_i\cdot a\cdot LFN_i\cdot b$。則 $EN_i(a,FOE(x))\cdot N_i(x)=\alpha_i\cdot a\cdot LFN_i\cdot FOE(x)\cdot N_i(x)$，我們將此函數記為 $ENC_i(a,x)$。

(19) 產生 m 個 noise function for TOKEN，記作 $TN_i: B_n\times B_n\rightarrow B_p$，$\forall i=1,2,...,m$。定義為 $TN_i(a,b)=\beta_i\cdot a\cdot LFN_i\cdot RFN_i\cdot b$。則 $TN_i(a,FOE(x))=\beta_i\cdot a\cdot LFN_i\cdot RFN_i\cdot FOE(x)$，我們將此函數記為 $TKN_i(a,x)$。

(20) 公開參數 $PP=(ENC_1, \ldots, ENC_m, IN_1, \ldots, IN_m, \alpha_1,\ldots, \alpha_m)$。

(21) 私密參數 $SP=(TKN_1, \ldots, TKN_m, \beta_1,\ldots, \beta_m, RFN_1, \ldots, RFN_m)$。

2. ENC(*PP*, *Doc*)

(1) 在 LB_n^+ 中，隨機挑選 m 個正辮，這 m 個正辮都互相不滿足辮群上的偏序關係。分別記作 $s_1, s_2, \ldots, s_m$。

(2) 計算第一部份可比較密文 $D_i=ENC_i(s_i,d_i),\forall i=1,2,...,m$。

(3) 計算第二部份可比較密文 $SA_i=s_i\cdot\alpha_i,\forall i=1,2,...,m$。

(4) 計算部分解雜訊 $EPN_i=IN_i(d_i),\forall i=1,2,...,m$。

(5) 令 $CC_i=D_i\,||\,SA_i\,||\,EPN_i,\forall i=1,2,...,m$。

(6) 對資料 M 進行加密，加密後資料記作 $E(M)$。

(7) 輸出密文 $C=CC_1\,||\,CC_2\,||\,\ldots\,||\,CC_m\,||\,E(M)$。

3. TOKEN(*SP*, *Pred*)

(1) 在 RB_n^+ 中，隨機挑選 m 個正辮，這 m 個正辮都互相不滿足辮群上的偏序關係。分別記作 $t_1, t_2, \ldots, t_m$。

(2) 計算第一部份可比較標記 $P_i=TKN_i(t_i,p_i),\forall i=1,2,...,m$。

(3) 計算第二部份可比較標記 $TA_i=t_i\cdot RFN_i\cdot\beta_i,\forall i=1,2,...,m$。

(4) 計算部份解雜訊 $TPN_i=\beta_i^2,\forall i=1,2,...,m$。

(5) 令 $CT_i=P_i\,||\,TA_i\,||\,TPN_i,\forall i=1,2,...,m$。

(6) 輸出標記 $T=CT_1\,||\,CT_2\,||\,\ldots\,||\,CT_m\,||\,Comp$。

4. QUERY(*T*, *C*)

(1) 計算 $Cipher=TA_iD_iTPN_iEPN_i$，$\forall i=1,2,...,m$。

(2) 計算 $Token=SA_iP_i$，$\forall i=1,2,...,m$。

(3) 根據 *Comp* 的條件，比較 *Cipher* 與 *Token* 兩者之間，在辮群上的偏序關係，$\forall i=1,2,...,m$。若比較條件為不考慮，則無須進行比較的動作。

(4) 如果所有 *Cipher* 與 *Token* 兩者之間皆滿足 *Comp* 的條件，則輸出"*Yes*"並回傳 $E(M)$；否則，輸出"*No*"並結束查詢。

5. 正確性：

定理一：若欄位值 d_i 小於(或大於)預測值 p_i 時，若且唯若，根據辮群左偏序關係，$Cipher=TA_iD_iTPN_iEPN_i\leq_L Token=SA_iP_i$(或 $Cipher\geq_L Token$)。

證明：

不失一般性假設，$d_i\leq p_i$

$\Leftrightarrow OE(d_i)\leq_L OE(p_i)$

$\Leftrightarrow$ 根據辮群左偏序關係，存在正辮 b 使得 $OE(p_i)=OE(d_i)\cdot b$。

$\Leftrightarrow \delta^{p_i}=\delta^{d_i}\cdot b$

$\Leftrightarrow \delta^{p_ic}=\delta^{d_ic}\cdot b^c$

$\Leftrightarrow$ 在 QUERY 的過程，最後計算 *Cipher* 與 *Token*。

$$\begin{aligned}Cipher&=TA_i\cdot D_i\cdot TPN_i\cdot EPN_i\\&=t_i\cdot RFN_i\cdot\beta_i\cdot\alpha_i\cdot s_i\cdot LFN_i\cdot\delta^{d_ic}\cdot\theta^{d_i}\cdot\alpha_i\cdot\beta_i^2\cdot\alpha_i\cdot(\theta^{-1})^{d_i}\\&=t_i\cdot RFN_i\cdot s_i\cdot LFN_i\cdot\delta^{d_ic}\\&=s_i\cdot t_i\cdot LFN_i\cdot RFN_i\cdot\delta^{d_ic}\end{aligned}$$

$$\begin{aligned}Token&=s_i\cdot\alpha_i\cdot\beta_i\cdot t_i\cdot LFN_i\cdot RFN_i\cdot\delta^{p_ic}\\&=t_i\cdot s_i\cdot LFN_i\cdot RFN_i\cdot\delta^{p_ic}\\&=t_i\cdot s_i\cdot LFN_i\cdot RFN_i\cdot\delta^{d_ic}\cdot b\\&=Ciphet\cdot b\end{aligned}$$

$\Leftrightarrow Cipher\leq_L Token$. *Q.E.D*

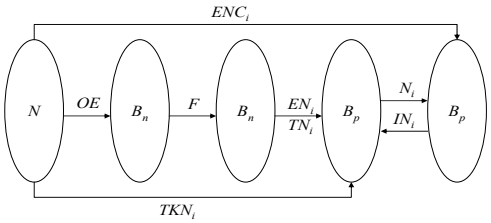
反向，同理可證。

● 4.2. 安全性分析與討論

我們提出的機制底下，必須確保其正確性與安全性。正確性是指當欄位值與預測值之間的大小關係，經過我們的機制運算之後，仍會保持兩者之間的關係。安全性是指伺服器或攻擊者無法區分出兩份文件或兩份預測的各欄位值的大小關係。

1. 安全性：

(1) 函數特性



圖三：我們提出的機制所用的函數。

OE：順序可嵌性函數，將整數轉換到辯群上，並保有其順序性。

F：單向函數，保持順序性私密。此函數架構在辯群的 Root Problem 上。

EN_i、TN_i：欄位雜訊函數，用來打亂不同文件中各欄位之間的順序性。

N_i：雜訊函數，用來打亂同一文件中同一欄位之間的順序性。

IN_i：解雜訊函數，用來還原同一文件中同一欄位之間的順序性。

(2) ENC_i 為單向函數。

$$\begin{aligned} ENC_i(a,x) &= EN_i(a, FOE(x)) \cdot N_i(x) \\ &= \alpha_i \cdot a \cdot LFN_i \cdot FOE(x) \cdot N_i(x) \\ &= \alpha_i \cdot a \cdot LFN_i \cdot \delta^{xc} \cdot \theta^x \cdot \alpha_i \\ &= \mu^{a_i} \cdot a \cdot LFN_i \cdot \delta^{xc} \cdot \theta^x \cdot \mu^{a_i} \end{aligned}$$

此函數可以視為兩個基於 RP 與一個基於 VRP 的單向函數合成。

(3) TKN_i 為單向函數。

$$\begin{aligned} TKN_i(a,x) &= TN_i(a, FOE(x)) \\ &= \beta_i \cdot a \cdot LFN_i \cdot RFN_i \cdot FOE(x) \\ &= \beta_i \cdot a \cdot LFN_i \cdot RFN_i \cdot \delta^{xc} \\ &= (\mu^{-1})^{a_i} \cdot a \cdot LFN_i \cdot RFN_i \cdot \delta^{xc} \end{aligned}$$

此函數可以視為一個基於 RP 與一個基於 VRP 的單向函數合成。

(4) 使用者產生的標記，只對該使用者的加密文件有查詢能力。

在 QUERY 時，伺服器去計算 *Cipher* 跟 *Token* 之間的關係。如果兩者之間有偏序關係存在，必須滿足 SA_i 與 P_i、TPN_i 與 EPN_i 且 TA_i 與 D_i 這三組相乘後必須把公開參數 α_i 抵銷。要抵銷公開參數 α_i 的關鍵在於只有該使用者的私密參數 β_i 才可以，這視同破解 RP。故，只有該使用者提供的標記，才會對該使用者的加密文件才有查詢能力。

(5) 一份文件同時滿足多份預測，而伺服器無法區分多份預測之間各預測值大小關係。

定理二：若一份文件 *Doc* 產生的密文為 $(D_i = ENC_i(s_i, d_i), SA_i = s_i \cdot \alpha_i, EPN_i = IN_i(d_i))$，同時滿足兩份預測 $Pred_1$ 與 $Pred_2$，兩份預測產生的標記分別為 $(P_i^{(1)} = TKN(t_i, p_i^{(1)})$，$TA_i^{(1)} = t_i^{(1)} \cdot RFN_i \cdot \beta_i$，$TPN_i = \beta_i^2)$與 $(P_i^{(2)} = TKN(t_i, p_i^{(2)})$，$TA_i^{(2)} = t_i^{(2)} \cdot RFN_i \cdot \beta_i$，$TPN_i = \beta_i^2)$，則 $Token_1$ 與 $Token_2$ 之間大小關係無法區分。

證明：

$$\begin{aligned} Token_1 &= SA_i \cdot P_i^{(1)} \\ &= s_i \cdot t_i^{(1)} \cdot LFN_i \cdot RFN_i \cdot FOE(p_i^{(1)}) \\ Token_2 &= SA_i \cdot P_i^{(2)} \\ &= s_i \cdot t_i^{(2)} \cdot LFN_i \cdot RFN_i \cdot FOE(p_i^{(2)}) \end{aligned}$$

不失一般性，假設 $Token_1 \le_L Token_2$

$\Rightarrow$ 一正辯 γ 使得 $Token_2 = Token_1 \cdot \gamma$

$$\begin{aligned} \Rightarrow Token_2 &= s_i \cdot t_i^{(2)} \cdot LFN_i \cdot RFN_i \cdot FOE(p_i^{(2)}) \\ &= s_i \cdot t_i^{(1)} \cdot LFN_i \cdot RFN_i \cdot FOE(p_i^{(1)}) \cdot \gamma \end{aligned}$$

$$\Rightarrow \gamma = \left[FOE(p_i^{(1)})\right]^{-1} \cdot \left[RFN_i\right]^{-1} \cdot \left[LFN_i\right]^{-1} \cdot \left[t_i^{(1)}\right]^{-1} \cdot t_i^{(2)} \cdot LFN_i \cdot RFN_i \cdot FOE(p_i^{(2)}) \rightarrow\leftarrow$$

故，$Token_1$ 與 $Token_2$ 在辯群上不存在左偏序關係。

(6) 多份文件同時滿足一份預測，而伺服器無法區分多份文件之間各欄位值大小關係。

分析方式與定理二相同。

(7) 與 HVE 機制比較

我們的機制讓使用者無須設計複雜的預測向量產生方式，也無須將文件經過預先處理。

(8) 時間複雜度

根據[4]，辯群上的計算與比較，皆可在多項式時間內達成。

(9) 空間複雜度

根據[4]，空間成長的速度為常數或線性。

5. 結論

本文提出的機制，主要特色在於提供一個更有效的讓使用者搜尋自己所需條件的文件。這方法比 Boneh 等學者提出的 HVE 要來的簡單，利用辯群的偏序關係與順序可嵌性函數，讓使用者無須設計複雜的預測向量產生方式，也無須將文件經過預先處理，即可比較文件與預測之間的關係。在辯群上的計算與比較，可在多項式時間內達成；其空間成長速度成常數或線性。透過這特點，提升範圍搜尋的便利性。

致謝

本研究由國科會輔助完成，計畫編號: NSC96-2628-E-005-076-MY3 與 NSC98-2221-E-468-012

參考文獻

[1] D. Boneh, G. D. Crescenzo, R. Ostrovsky and G. Persiano, "Public Key Encryption with Keyword Search.", ***Proceedings of IEEE Symposium on Security and Privacy***, pp.44-45 IEEE, Apr 2004.

[2] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data.", ***proceedings of TCC'07***, ***LNCS 4392***, pp. 535-554, 2007.

[3] J. Bethencourt, H. Chan, A. Perrig, E. Shi, and D. Song, "AnonymousMulti-Attribute Encryption with Range Query and Conditional Decryption.", ***Technical report***, ***C.M.U***, 2006.

[4] J. C. Cha, K. H. Ko, S. J. Lee, J. W. Han and J. H. Cheon, **"**An Efficient Implementation of Braid Groups**."**, ***LNCS 2248***, Springer-Verlag, pp. 144-156, 2001.

[5] J. H. Cheon, and B. Jun, **"**A polynomial time algorithm for the braid Diffie-Hellman conjugacy problem**."**, ***Advances in cryptology–CRYPTO***, 2003.

[6] P. Golle, J. Staddon, and B. Waters, "Secure Conjunctive Keyword Search over Encrypted Data.", ***Proceedings of Applied Cryptography and Network Security Conference***, ***LNCS 3089***, Springer-Verlag, pp.31–45, 2004.

[7] J. Hughes, "A Linear Algebraic Attack on the AAFG1 Braid Group Cryptosystem.", ***LNCS 2384***, pp. 176-189, 2002.

[8] A. G. Kalka, "Representation Attacks on the Braid Diffie-Hellman Public Key Encryption.", ***Applicable Algebra in Engineering, Communication and Computing Vol. 17, No. 3-4***, pp. 257-266, 2006.

[9] K**.** H**.** Ko, S**.** J**.** Lee, J**.** H**.** Cheon, J**.** W**.** Han, J**.** S**.** Kang, and C. Park, **"**New public-key cryptosystem using braid groups.**"**, ***LNCS 1880,*** Springer-Verlag, pp.166–183, 2000.

[10] E. Lee, S. J. Lee, and S. G. Hahn, **"**Pseudorandomness from braid groups**."**, ***LNCS 2139***, Springer-Verlag, pp. 486-502, 2001.

[11] K. Murasugi and B. I. Kurpita, ***A study of braids***, Kluwer Academic Publishers, 1999.

[12] D. J. Park, K. Kim, and P. J. Lee, "Public key encryption with conjunctive field keyword search.", ***Proc. WISA 2004***, pp. 73-86, Springer-Verlag, 2004.

[13] E. K. Ryu and T. Takagi, "Efficient Conjunctive Keyword-Searchable Encryption.", ***Advanced Information Networking and Applications Workshops***, pp. 409-414, 2007.

[14] E. Shi, J. Bethencourt, H. Chan, D. Song, and A. Perrig., "Multi-Dimensional Range Query over Encrypted Data.", ***IEEE Security and Privacy Symposium***, May 2007.

[15] D. Song, D. Wagner and A. Perrig, "Practical Techniques for Searches on Encrpted Data.", ***Advances in Cryptology - EUROCRYPT 2004***, pp.506-522, May 2000.