資料庫在雲端環境的自我保護

陳靖國*
朝陽科技大學資訊管理系 助理教授
jkchen@cyut.edu.tw

張芳昇
朝陽科技大學資訊管理系 研究生
s10154607@cyut.edu.tw

摘要

雲端概念在工商發展上日益盛行，資料儲存與處理逐漸普級，但也面臨了一些議題與挑戰，例如很受關注的資料安全問題。上傳雲端的資料庫資料如果有統計、探勘、或分析上的應用需要，不能用編碼過的亂碼而必需以明碼方式儲存，則資料安全議題，對企業而言是非常重視的。本論文提出一個自我資料保護方法，將原始資料利用若干種資料順序攪亂手法，加以適當調整。調整過的資料從表面上無法取得真正的意義，不用擔心資料被盜取而洩露隱私或機密，而且調整後的資料不會影響其原來可供統計、探勘、或分析的功能。

關鍵詞: 雲端計算，資料攪亂，隱私保護、資料庫

Abstract

Cloud concept is popular on data store and data process for the development of industry and business. But it also has a number of issues and challenges. One of the most talked about topics is data security. If the data of database for uploading to cloud has the necessity of statistic, mining, or analysis application, the data must be stored in clear form rather than encrypting form. This paper proposes a protection method for the data to be calculated, mined, or analyzed in the cloud. The raw data is confused with several disordered steps before uploading to the clouds. The real data meaning cannot be obtained from the confused data unless using a series of reverse engineering processes. Therefore, we need not worry about the stolen data to disclosure privacy or secret. Besides, the confused data will not affect the original functions for calculating, mining, or analyzing in the cloud.

Keywords: Cloud computing, Data confuse, Privacy protection, Database

1. 前言

雲端概念不僅為資訊科技產業帶來新世紀的革命[9]，更帶來為數龐大的商機與利益，引起許多企業組織的注意。大企業注意到雲端的架設、租賃是有利可圖，而中小型企業則重視雲端的使用可以節省許多管理成本[5]。雲端的特性是資料與應用程式皆放置在雲端服務公司的電腦上，企業組織要使用時，只需簡單的硬體設備，例如智慧型手機、筆記型或平版電腦、PC個人電腦等，利用網際網路連線，便能存取資料或使用應用程式。此外雲端服務是採取使用者付費方式，用多少付多少，對於無法負擔巨額成本，購置軟硬體設備與後續維護的中小企業是一個方便機制。

雲端所帶來的商機非常龐大，2010年

四月底政府宣佈「雲端計算產業發展方案」，撥出高達新台幣240億元的經費推動，代表政府正積極推動雲端計算發展，勢必帶來一個全新的網路應用時代，但相對也衍生出一些問題。根據2008年IDC一份調查發現[5]，雲端計算最受關注的議題之一是資料安全。這是一項重要因素，會讓使用者考量是否願意使用雲端服務。因為使用者將資料庫放到雲端上去處理時，無法控制也不知道資料庫存放在哪裡，更難保證資料庫是否有遭駭客盜取或篡改。個人的隱私資料或商業的機密資料萬一被盜取利用，巨大損失將會使個人或企業難以承受。

雖然雲端服務公司會提供安全措施保護客戶資料，但是客戶本身也應該要有自我防衛意識，畢竟道高一尺，魔高一丈。企業或使用者應該需要有自我保護的辦法[1]，防止因雲端服務公司內賊監守自盜而蒙受巨額損失。本論文研究目的，就是為特定(隱私或機密)資料設計保護方法，將上傳雲端的資料預先加以適當地攪亂處理，使個別資料失去原來意義，但不會影響整體資料的完整性，而讓資料可以順利地在雲端上維持原有的計算、探勘或分析之處理，以供委外處理公司的外包工作順利進行。

2. 文獻探討

2.1 隱私權

隱私權一詞最早在1890年由美國Samuel D. Warren與Louis D. Brandeis所提出[10]。這兩位學者合著的「The Right to Privacy」論文中正式提出隱私權為法律概念，並強調任何人有不受干擾之權利，該文重點放在法律應保障個人隱私之生活利益，並未詳細界定隱私權概念與保護範圍。歐洲於第二次世界大戰時期，因德國政府以個人隱私與身家資料來迫害不同種族，造成慘痛浩劫，引發戰後歐洲對個人隱私保護議題的重視，進而推動個人資料保護法制化。1980年由經濟合作與發展組織 (The Organization for Economic Co-operation and Development, OECD)制訂的《管理保護個人隱私及跨國界流通個人資料指導綱領》，以及歐盟1995年頒訂之《個人資料保護指令》，為保護個人資料建構了法制意涵與遵循原則。亞太經濟合作會議接續了OECD的保護個人資料的核心價值，於2004年制定《APEC隱私保護綱領》，以平衡亞太地區各經濟體間資訊自由流動與隱私保護[2]。

OECD的《管理保護個人隱私及跨國界流通個人資料指導綱領》有八大原則[2]：

(1) 限制蒐集原則 (Collection Limitation Principle)
(2) 資料品質確保原則 (Data Quality Principle)
(3) 目的明確原則 (Purpose Specification Principle)
(4)利用限制原則(Use Limitation Principle)
(5) 安全保護原則 (Security Safeguards Principle)
(6)公開原則(Openness Principle)
(7) 個人參與原則 (Individual Participation Principle)
(8) 責任義務明確原則 (Accountability Principle)

亞太經濟合作會議(The Asia-Pacific Economic Cooperation, APEC)，於2004年訂定《APEC隱私保護綱領》(APEC Privacy Framework)。APEC的九大隱私權保護原則

如下[2]：
(1)避免損害原則(Preventing Harm)
(2)告知原則(Notice)
(3)限制蒐集原則(Collection Limitation)
(4)利用個人資料原則(Uses of Personal Information)
(5)當事人選擇原則(Choice)
(6)個人資料完整原則(Integrity of Personal Information)
(7)安全維護原則(Security Safeguards)
(8)當事人查詢及更正原則(Access and Correction)
(9)責任原則(Accountability)

2.2 資料攪亂方法

關聯式資料庫基本上是由若干個關聯資料表組成，每一個關聯資料表又是由若干個欄位組成。例如一家製造業公司的資料庫可能包含員工、商品、訂單、原料等關聯資料表，而員工資料表可能包含代號、姓名、職稱、薪資等欄位。一個表格如果包含個人隱私資料(例如薪資)或企業機密資料(例如配方)，當需要上傳雲端給委外廠商做業務處理時，例如員工薪資轉帳，如果擔心隱私/機密資料被盜取，最保險的做法就是將表格欄位資料攪亂，亦即將幾筆完整資料的某些欄位值調換，改變欄位之間原本的順序關係，而不是用資訊安全編碼方式將資料變成亂碼，因為亂碼是無法直接供委外廠商使用。

最簡單的資料攪亂方法是資料交換(Data swapping) [6]。這個方法主要功能在於將表格裡的資料列欄位值兩兩互相交換，攪亂原始欄位資料之間的順序，進而達到保護原始資料內容雖然是明碼，但是正確完整資料列卻不易被發現。資料交換方法有下列的優缺點[8]。

優點：
1. 可以準確的遮蔽每一筆資訊。
2. 可以移除欄位與欄位之間資料的正確對應關係。
3. 只需一個亂數產生程序即可實作，程式非常簡單。
4. 資料交換程序可以設定一個或多個變量，不會影響非敏感或非識別資料。
5. 使用連續性的變量和交換公式頻繁的變化，讓資料不易被破解。
6. 不僅限於單一欄位，也可以多個欄位，例如姓名、職業、生日等欄位互換，使原本的資料看起來失去真實性，創造出更多種奇怪的組合。

缺點：
1. 任意的多重欄位互換可能產生怪異的組合，變成不尋常的紀錄。例如男性的性別欄位被換成女性的名字。
2. 可能需要大量的計算時間和資源來進行資料交換和儲存交換後的資料版本。
3. 降低資料可分析的價值。例如將資料類型沒有相容的兩個欄位任意互換，會無法分析想獲得的資訊。

3. 特定欄位資料保護方法

本論文提出一個資料旋轉(Data Rotation)方法，參考資料交換方法的優點並避開其缺點，對資料庫的指定關聯表做加工處理。假設一個關聯表有 m 個欄位，有需要保護的 n 個敏感欄位，$n \leqq m$。利用旋轉原理將所有資料列的 n 個敏感欄位，採取一系列往上或往下旋轉動作。每一輪旋轉時，某一個欄位的資料會有旋轉的動作，經過 n 輪之後，有 n 個欄位先後旋轉即告完成。以下舉例說明敏感資料如何旋轉。假設有三個敏感欄位 a_1、a_2、a_3，其

隨機處理的先後順序為 a_2(+*c1*)、a_3(-*c2*)、a_1(+*c3*)，括號內的'+'、'-'符號分別表示往下、往上的旋轉方向，而'*c1*'、'*c2*'、'*c3*'表示旋轉時的次數。第一輪處理時，a_2 欄位將往下旋轉 *c1* 次。第二輪處理時，a_3 欄位將往上旋轉 *c2* 次。到第三輪處理時，a_1 欄位則往下旋轉 *c3* 次。敏感欄位資料旋轉順序、旋轉方向、與次數由亂數產生器決定，主要是考量這些數據會影響資料的攪亂度，所以由電腦產生，避免由人工輸入而容易被破解的情形發生。

考慮欄位之間資料型態可能不同，本資料旋轉方法不會像一般資料交換方法那樣，將不同欄位的資料互相調換，而是針對敏感欄位的資料作個別處理，盡量保持整份資料的完整性，使其具有統計、分析等價值。另外，本資料旋轉方法比資料交換方法簡單又容易操作，在資料處理上不需欄位兩兩互換，而是用欄位垂直旋轉方式來改變欄位資料的位置。目的是要去除一筆資料內敏感欄位與其他普通欄位資料之間的正確對應關係，但又不會影響原來資料的整體性。

以下介紹本資料旋轉方法的演算法 DataRotation。輸入的參數有兩個，原始的資料關聯表 DataRel 與需要旋轉的欄位群陣列 AttArr。輸出參數有三個，處理後的新關聯表 NewDataRel、需要旋轉欄位的新順序陣列 NewAttArr、和旋轉次數的陣列 CntArr。NewAttArr 代表 AttArr 的欄位順序隨機重新排序後的結果。CntArr 包含之數值代表每一輪要旋轉欄位的旋轉次數，皆為亂數產生。公式為 $r_i = \pm$ (全部資料筆數/2)，$r_i \neq 0$。'+'代表向下旋轉，'-'則代表向上旋轉，除以 2 的原因在於減少處理時間。假設有 10 筆資料，要將某個欄位的第一個資料旋轉至第 10 個位置，可以用向下+9 的方式旋轉或者向上-1 的方式旋轉，都可以達到同樣結果，然而從旋轉距離來看，向上-1 的旋轉較為快速。演算法內容如下。

Algorithm DataRotation (DataRel, AttArr)
//Input parameters:
DataRel(a_1, a_2, a_3,..., a_m): data relation;
AttArr(c_1, c_2, …, c_n): rotation attribute array;
Output parameters:
NewDataRel(a_1, a_2, a_3,..., a_m): new data relation;
NewAttArr(c'_1, c'_2, c'_3, ..., c'_n): reordered rotation attribute array;
CntArr(r_1, r_2, r_3, ..., r_n): rotate count array;//
1. NewAttArr ← random reorder of AttArr;
2. CntArr ← n random numbers generation;
3. NewDataRel ← DataRel;
4. for (i=1; i≦n; i++)
5. rotate CntArr[i] times the values of attribute NewAttArr[i] in NewDataRel;
6. end for;
7. return (NewDataRel, NewAttArr, CntArr);
8. end DataRotation.

4. 實例說明

以下用一個製造業資料庫的員工資料為範例，說明本論文所提出的方法在敏感資料保護的做法。假設某製造業有使用雲端服務的 DaaS (Data storage as a Service) 服務[5]，需要將員工基本資料關聯表定期上傳至雲端服務公司，並委託該雲端機構做某些相關資料的統計分析，例如員工的平均薪資、罹患那些疾病等。表 1 是員工基本資料關聯表原始內容，因為篇幅有現，只呈現 10 筆重要敏感欄位資料以方便

解說。這些資料是經過特別設計，例如身分證 ID 欄位值的第一個英文字母分別從 A 到 J，姓名欄位值的第二個字分別從一到十，依此類推，好方便查閱。上傳之前務必使用本論文所提出的方法將局部資料攪亂，以免員工的隱私或敏感資料洩露，造成個人或公司的損失。

表 1. 關聯表原始資料

身分證 ID	姓名	生日	薪資	病史
A1…	陳一	61/01/01	31k	心臟病
B2…	林二	62/02/02	32k	糖尿病
C1…	張三	63/03/03	33k	高血壓
D1…	李四	64/04/04	34k	肝炎
E1…	王五	65/05/05	35k	糖尿病
F2…	毛六	66/06/06	36k	氣喘
G2…	周七	67/07/07	37k	心臟病
H1…	汪八	68/08/08	38k	高血壓
I1…	馬九	69/09/09	39k	糖尿病
J2...	吳十	70/10/10	40k	肝炎

關聯表欄位可分為兩大類[3]：基本資料欄位與隱私資料欄位。可以識別個人身份的敏感欄位，例如身分證 ID 或姓名，必須加以保護以免洩露個人隱私資料。為了確保資料的完整性，以供統計或研究分析用，隱私資料欄位將保留不刪除，利用攪亂資料欄位原本位置的手法，讓實際資料列中的基本資料欄位與隱私資料欄位無法正確銜接起來，以防止資料洩露。如此，就算資料被盜取，竊取者也無法從中獲得一份完整的個人資料。

表 1 包含的身分證 ID、姓名、生日、薪資、病史，都是需要保護的敏感欄位。因為身分證 ID 與姓名具有識別個人身份的功能，生日算是個人敏感資料，如果與身分證 ID 或姓名配合使用，很容易查出個人隱私，所以也必須保護。薪資也是高所得個人的敏感資料，如果被同事知道會尷尬，被敵對公司知道會擔心被挖角，所以必須保護。至於病史，則是一般人極為重視的隱私，絕對不願意資料外洩。原則上欲保護的欄位愈多，資料攪亂程度愈高，保護效果自然愈好。

決定這五個欄位有需要攪亂資料之後，再來就是決定攪亂的先後順序。假設利用亂數產生器，隨機產生欄位旋轉的順序與資料旋轉的次數為：姓名(+3)、病史(-2)、身分證 ID (-4)、薪資(+5)、生日(-1)。旋轉處理過程描述如下。第一輪時，首先姓名欄位的資料往下旋轉三個位置，例如陳一從第 1 格降至第 4 格，結果如表 2。

表 2. 第一輪旋轉結果

身分證 ID	姓名	生日	薪資	病史
A1…	汪八	61/01/01	31k	心臟病
B2…	馬九	62/02/02	32k	糖尿病
C1…	吳十	63/03/03	33k	高血壓
D1…	陳一	64/04/04	34k	肝炎
E1…	林二	65/05/05	35k	糖尿病
F2…	張三	66/06/06	36k	氣喘
G2…	李四	67/07/07	37k	心臟病
H1…	王五	68/08/08	38k	高血壓
I1…	毛六	69/09/09	39k	糖尿病
J2...	周七	70/10/10	40k	肝炎

第二輪時，病史欄位往上旋轉兩個位置，例如心臟病從第 1 格旋轉至第 9 格，結果如表 3 所示。

表 3. 第二輪旋轉結果

身分證 ID	姓名	生日	薪資	病史
A1…	汪八	61/01/01	31k	高血壓
B2…	馬九	62/02/02	32k	肝炎
C1…	吳十	63/03/03	33k	糖尿病
D1…	陳一	64/04/04	34k	氣喘
E1…	林二	65/05/05	35k	心臟病

F2…	張三	66/06/06	36k	高血壓
G2…	李四	67/07/07	37k	糖尿病
H1…	王五	68/08/08	38k	肝炎
I1…	毛六	69/09/09	39k	**心臟病**
J2...	周七	70/10/10	40k	糖尿病

第三輪時，身分證 ID 欄位的資料往上旋轉四個位置，結果如表 4 所示。

表 4. 第三輪旋轉結果

身分證 ID	姓名	生日	薪資	病史
E1…	汪八	61/01/01	31k	高血壓
F2…	馬九	62/02/02	32k	肝炎
G2…	吳十	63/03/03	33k	糖尿病
H1…	陳一	64/04/04	34k	氣喘
I1…	林二	65/05/05	35k	心臟病
J2...	張三	66/06/06	36k	高血壓
A1…	李四	67/07/07	37k	糖尿病
B2…	王五	68/08/08	38k	肝炎
C1…	毛六	69/09/09	39k	心臟病
D1…	周七	70/10/10	40k	糖尿病

第四輪時，薪資欄位的資料往下旋轉五個位置，結果如表 5 所示。

表 5. 第四輪旋轉結果

身分證 ID	姓名	生日	薪資	病史
E1…	汪八	61/01/01	36k	高血壓
F2…	馬九	62/02/02	37k	肝炎
G2…	吳十	63/03/03	38k	糖尿病
H1…	陳一	64/04/04	39k	氣喘
I1…	林二	65/05/05	40k	心臟病
J2...	張三	66/06/06	31k	高血壓
A1…	李四	67/07/07	32k	糖尿病
B2…	王五	68/08/08	33k	肝炎
C1…	毛六	69/09/09	34k	心臟病
D1…	周七	70/10/10	35k	糖尿病

最後一輪，生日欄位的資料往上旋轉一個位置，結果如表 6 所示。

表 6. 第五輪旋轉結果

身分證 ID	姓名	生日	薪資	病史
E1…	汪八	62/02/02	36k	高血壓
F2…	馬九	63/03/03	37k	肝炎
G2…	吳十	64/04/04	38k	糖尿病
H1…	陳一	65/05/05	39k	氣喘
I1…	林二	66/06/06	40k	心臟病
J2...	張三	67/07/07	31k	高血壓
A1…	李四	68/08/08	32k	糖尿病
B2…	王五	69/09/09	33k	肝炎
C1…	毛六	70/10/10	34k	心臟病
D1…	周七	61/01/01	35k	糖尿病

最終結果顯示，每一筆完整的原始資料，確實有被攪亂的效果。例如，陳一這位員工的資料中，身分證 ID 欄位值，從”A1…”變成”H1…”，生日欄位值從”61/01/01”變成”65/05/05”，薪資欄位值從 31k 變成 39k，病史欄位值從心臟病變成氣喘。可見本論文提出的方法確實可行。

5. 結論

雲端概念的盛行，使得資料安全問題成為研究焦點。本論文針對需要雲端儲存、探勘、或分析的資料庫，提出一個簡單的旋轉方法來保護敏感資料。從原始關聯表中選定幾個敏感欄位，然後將每一個欄位的所有資料值視為一個環狀連結，以向下或向上方式加以旋轉調整其位置。使得一筆資料的若干欄位值變成其他筆資料相同欄位的值。調整過後的資料無法辨識出個人真正的完整資料，就算資料洩露也不會影響個人隱私與權益，而且整體資料

並沒有修改，因此在雲端上的原來處理功用不會受到影響。考慮保護方法的功能性與資料使用便利性之間的平衡，未來希望以旋轉方法為基礎，結合其他方法，例如 Swapping、Perturbation、Aggregation [4] 等，發展功能更強而不易被破解的資料保護方法。

參考文獻

[1] 劉家驊、洪士凱(2010)，「雲端運算資料安全防護機制之研究」，2010 電腦視覺、影像處理與資訊技術研討會，桃園，第 100-109 頁。

[2] 林亞萱(2013)，新版個資法對於企業客戶關係管理活動之研究，碩士論文，國立中山大學企業管理學系，高雄。

[3] 江育誠(2000)，公開資料庫之個人隱私保護，碩士論文，台灣大學，台北。

[4] V. S. Verykios, K. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin and Y. Theodoridis, “State-of-the-art in Privacy Preserving Data Mining,” ACM SIGMOD Record, Vol. 33, No. 1, 2004, pp. 50-57.

[5] Tharam Dillon, Chen Wu, and Elizabeth Chang, “Cloud Computing: Issues and Challenges,” IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 27-33.

[6] Stephen E. Fienberg and Julie McIntyre, “Data Swapping: Variations on a Theme by Dalenius and Reiss,” Journal of Official Statistics, vol. 21, no. 2, pp. 309-323, 2005.

[7] Siani Pearsonm, “Taking Account of Privacy when Designing Cloud Computing Services,” ICSE Workshop on Software Engineering Challenges of Cloud Computing, pp. 44-52, 2009.

[8] Richard A. Moore, Jr., “Controlled Data-Swapping Techniques for Masking Public Use Microdata Sets,” SRD Report RR 96-04, U.S. Bureau of the Census, 1996.

[9] Jian Wang, Yan Zhao, Shuo Jiang and Jiajin Le, “Providing Privacy Preserving in Cloud Computing,” International Conference on Test and Measurement, 2009, pp. 213-216.

[10] Samuel D. Warren and Louis D. Brandeis, The Right to Privacy, 1890.