Secret Sharing with Multi-cover Steganographic Audio Files

洪胤勛
部門名稱
服務單位
Email吳坤熹
部門名稱
服務單位
Email3rd 作者二
部門名稱
服務單位
Email 4th 作者二
部門名稱
服務單位
Email 5th 作者二
部門名稱
服務單位
Email 6th 作者二
部門名稱
服務單位
Email

摘要

secret sharing[1, 2]為一種分享密文的技術，它會把密文拆成若干份sharing，並且把sharing分給不同的人。若要把原本的密文給解開，則需要湊齊大於或等於(threshold)數量的sharing，否則不能解開原本的密文。本論文將探討要如何安全的把這些sharing分送給不同的人，並導入隱血術，將sharing藏在聲音中，讓這些sharing在不易被察覺的情況下發送給不同的人。

1. 前言

secret sharing為一種分享密文的技術，在這種架構下，它解決了密文掌握在單一個人手上的問題，比如需要多人授權的密碼，像是發射火箭，會需要多可單位的核可，不能由單一人來決定火箭的發射;又或是階層式授權的密碼，比如銀行金庫的密碼，為了應付臨時的緊急狀況，可能需要授權給經理和主管，但也不能讓單一個經理或是單一個主管直接擁有密碼，而是讓一個經理和一個主管，或是三個主管決定要打開金庫時，才能得出密碼。這些問題都可以靠secret sharing來解決。

secret sharing同時也解決了密文掌握在單一個人手上的兩個問題。第一為密文的穩健性，在單一個密文的架構下，當系統被駭客入侵，密文被竊取走，那就表示密文一定被駭客知曉;但在secret sharing的架構下，駭客竊取走單一一份資料他並無法得知密文到底為何，他必去找到一定數量的資料才有辦法破解。第二為密文的可靠性，在單一密文的架構下，密文不小心遺失或式密文被駭客破壞掉了，那就表示再也無法取得密文;但在secret sharing的架構下，即使遺失一份或損壞一份資料仍然有其他的資料可以復原。

secret sharing的解法在1979年相繼被Adi Shamir[1]和George Blakley[2]發表。而他們的解法又常分別被稱為Shamir’s Secret Sharing與Blakley’s Secret Sharing。而本論文將著重在Shamir’s Secret Sharing。

1. 研究動機
2. 相關研究
3. 圖片的[3]
4. 研究目標
5. 技術背景
6. Shamir’s Secret Sharing

Shamir’s Secret Sharing的演算法共有兩個步驟。第一步驟是將原本的密文D拆成n份sharings，並決定需要幾份以上(k)的sharings才能還原密文，且k ≤ n，並以k的值來決定一個二維平面的最高次項的值為k – 1，而零次項的值則為D。

第二步則從這n份sharings中湊得k份sharings，並將它還原為原本的密文，若湊得的份數少於k份，則無法還原。拆成n份sharings與需要k份sharings才能組回去原本密文的情境又稱為(k, n) threshold。

以下將以threshold為(3, 5)，D為34為例。在Shamir’s Secret Sharingk的第一步驟中，將使用二維平面的多項式差值，由於t為3，因此需要產生的二維平面多項式的最高次方為2，而零次項的值為34，其它次項的值則可隨機產生，以公式(1)為例，

$$F\left(x\right)=4x^{2}-21x+34(1)$$

由於n為5，因此需要在這平面上隨機找5個點，F(2) = 8、F(3) = 7、F(4) = 14、F(5) = 29、F(6) = 52。

而在Shamir’s Secret Sharing第二步驟中，使用了拉格朗日差值法。如果收集到了3個sharing，F(3) = 7、F(5) = 29、F(6) = 52，並帶入拉格朗日差執法(2)，並可得出(1)，D為34。

$$\begin{array}{c}\begin{array}{c}F\left(x\right)=7\frac{\left(x-5\right)\left(x-6\right)}{\left(3-5\right)\left(3-6\right)}+29\frac{\left(x-3\right)\left(x-6\right)}{\left(5-3\right)\left(5-6\right)}\\+52\frac{\left(x-3\right)\left(x-5\right)}{\left(6-3\right)\left(6-5\right)}(2)\#\end{array}\end{array}$$

1. Least Significant Bit

Least Significant Bit簡稱LSB。指的是在2進位的數字中，最小的位數。由於最小位數的更動，代表著更動後的值只會相差1，不容易被發現，因此經常被拿當作一種隱血術的方式。

1. 實驗架構
2. 在此實驗架構中，將會有server和client兩個角色的架構。首先server需要決定好D和(k, n)，並產生n份sharings後，等待client的連線。
3. 當有client連上server時，server則可以決定是否要傳送資料。當server決定要開始傳送資料時，server則會開始錄音。在本實驗中，將會以sample rate為8000、channel數1，並用format為unsinged int8來錄製5秒鐘的聲音。
4. 錄完音後，會把n份sharing的其中一份藏進錄音檔裡。使用的方式是用是用LSB藏在sample裡，本實驗使用的format為unsinged int8，所以每個sample就為一個byte，也就是一個byte可以藏入一個bit。以圖1.，sharing為**1-0bf53ca7dd**當範例，先計算D的長度，為12。再把12用一個byte來表示，為00001100。之後把00001100這8個bit藏進前8個sample中。之後把sharing用ASCII來表示，也就是一個字元就是8個bit，所以一個字元會藏在8個sample裡，以上述的範例來說，則需要96個sample來藏。



圖1. 隱藏範例

1. 把已經藏好的音檔，包成WAV檔，並使用TCP把此WAV檔送到client。之後，server又回到第二步，直到所有sharing被領取完畢。
2. client收到WAV檔後，會先把所有的sample從WAV檔取出來，並從前八個sample取得sharing的長度。取得長度之後，就能知道總共要再取幾個sample，並得出sharing。
3. 在n個client中，他們手上都會有不同的sharing。只要聚集k份以上的sharings，就能得出D。
4. 結論

誌謝辭

參考文獻

[1] A. Shamir, "How to share a secret," *Communications of the ACM,* vol. 22, no. 11, pp. 612-613, 1979.

[2] G. R. Blakley, "Safeguarding cryptographic keys," in *Managing Requirements Knowledge, International Workshop on*, 1979: IEEE Computer Society, pp. 313-313.

[3] H.-D. Yuan, "Secret sharing with multi-cover adaptive steganography," *Information Sciences,* vol. 254, pp. 197-212, 2014.