An Asymmetric-key-based Authentication Scheme for Session Initiation Protocol

Shin-Fu Huang Quincy Wu
Department of Computer Science and Information Engineering, National Chi Nan University
No. 1, University Road, Puli, Nantou 545, Taiwan
{s97321539, solomon}@ncnu.edu.tw

***摘要*—本篇論文探討會議初始協定(Session Initiation Protocol)的認證機制。目前定義於標準協定中的認證機制是以使用者密碼為基礎(Password-based)的對稱式金鑰系統，使用者與認證伺服器皆保有相同的認證鑰匙(也就是使用者的密碼)。此認證鑰匙在認證伺服器中是以明文的形式被保存在認證資料庫內，因此一旦管理用戶資料的人員利用職務之便竊取或盜賣用戶資料，造成使用者的身分認證資料流入不明人士的手中，該帳號便可能被冒名使用，造成使用者的權益與個人信用因此受到嚴重的損害。本篇論文提出了以非對稱式金鑰系統改善對稱式金鑰系統並且實作系統驗證此一改善方法的可行性。**

***關鍵詞*—網路電話, 會議初始協定, 身份盜竊, 公開金鑰密碼演算法, 身分認證, 竊取驗證碼攻擊**

一、簡介

在 Internet Protocol (IP)的應用上，IETF 在 RFC 3261 文件中所提出的 Session Initiation Protocol[8, 10]，簡稱 SIP，是繼 H.323 之後，被公認在 Voice over IP(VoIP)[4]的信令(Signaling)及通話控制上，極具潛力的一個通訊協定。相較於 SIP，ITU-T 所制定的標準 H.323 協定過於複雜、擴充性低、規則修改繁雜，因此要在其中開發新的服務或增加新的元件實在是相當不易。SIP 的其中一個優勢在於其擴充性，採用以文字為基礎的標頭格式(Header Format)，它的格式類似 Hypertext Transfer Protocol(HTTP)[11]，易於擴充，因此已被 Third Generation Partnership Project(3GPP) 採用作為其 IP Multimedia Subsystem(IMS)的信令協定。

近年來，由於 SIP 這個 VoIP 的通訊協定被越來越多的人使用，所以如何增加使用者帳號安全性的問題被重新提出來討論。目前 SIP 帳號認證的方式[6]是採用 Challenge - Response 的機制，雙方使用一把共享的固定金鑰（也就是使用者的密碼）去進行認證。但是在管理者的資料庫中，使用者帳號與密碼皆是以明文的方式被保存著，一旦帳號資料庫被非法人士入侵，或者是擁有權限的管理人員離職前監守自盜，都可以在取得使用者帳號密碼後，輕易地冒充其身分撥打長途電話，照成帳務的錯亂與使用者的困擾；若是用於恐嚇與詐欺電話，更可能造成無辜的使用者變成犯罪嫌疑人。由於管理人員監守自盜的事件時有所聞[1, 9]，所以本篇論文的重點即是改善 SIP 的認證方式，將原本使用對稱式鑰匙的認證方式改變成使用非對稱式鑰匙的系統，以提高使用者帳號的安全性。

二、相關工作

相關工作的部分，我們將說明與分析到目前為止專家學者對於會議初始協定所提出的各種加強認證安全性的方法。

在 2005 年時，Yang 等人提出”Secure authentication scheme for session initiation protocol”[3]，此方法的認證機制是基於 Diffie-Hellman Key Exchange[14]，使用者只需要於客戶端輸入自己設定的密碼，而在伺服器端的帳號認證資料庫內則事先會存放使用者所設定的密碼，再進行認證時是將雜湊過後的密碼代入指數公式做運算。在同一年，Durlanik 等人提出”SIP authentication scheme using ECDH”[2]，此認證方法主要是針對 Yang 等人所提出的認證方法做修改，將橢圓曲線密碼學(Elliptic Curve Cryptography) 運用到 Diffie-Hellman Key Exchange 的運算中，原本需要做指數公式的運算變成只需要做乘法公式運算，用更短的金鑰長度與更少的計算量也可以得到相同的安全性，此方法所使用的認證金鑰與 Yang 等人所提出的方法一樣。在 2009 年時，Tsai 提出”Efficient Nonce-based Authentication Scheme for Session Initiation Protocol”[7]，指出 Yang 等人所提出的方法計算成本還是太高了，較不適用於計算能力較低的裝置，所以發表了這篇以 nonce(短時間內具有認證效力的隨機值)為基礎的認證方式。此認證方式是以使用者所設定的密碼為認證鑰匙，僅以雜湊函數與位元運算來做運算，藉由能否取得正確的 nonce 值來驗證身分合法性。此認證方法客戶端與伺服器端皆需要事先共享同一把認證金鑰，並且在伺服器上是直接將認證金鑰(使用者的密碼)以明文模式存放在認證資料庫內。

表一 各方法所使用認證金鑰表

	認證金鑰	伺服器端所保存的認證金鑰形式	Stolen-Verifier Attack
Digest-MD5[6]	*PW*	*PW*	*YES*
Yang[3]	*PW*	*PW*	*YES*
Durlanik[2]	*PW*	*PW*	*YES*
Tsai[7]	*PW*	*PW*	*YES*

註: *PW* :為使用者所設定之密碼，未經過任何的轉換。

H() :為雜湊函數。

表一是將上述各種認證方法整理過後的金鑰分析表，每個認證的方法都有其論文內所宣稱安全性上的優點，可是如何保護存放在認證伺服器內的認證金鑰(使用者密碼)卻未被同等的重視，甚至絕大多數的認證方法直接就將使用者密碼以明文的模式存放在認證資料庫中。這樣一來如果整個密碼表被有心人士所竊取或是管理人員利用職務之便盜賣用戶資料，那麼即使認證機制被設計得可以抵禦再多的攻擊、或者是認證公式的運算可以達到如何的輕巧，屆時都於事無補。尤其是近年來網路帳號被盜用的事件層出不窮，所以如何預防竊取認證金鑰攻擊(Stolen-Verifier Attack)維護使用者的權益是一個很重要的課題，值得我們去加以重視。

三、對稱與非對稱式系統

(一) 現行的 Shared-Key System

現行的 SIP 通訊協定中，使用者代理(User-Agent，簡稱 UA)與 SIP 的註冊伺服器，以下簡稱 Registrar，皆是使用相同的一把鑰匙(也就是使用者密碼)做 SIP 認證的計算。為了安全起見，這個密碼不在公開的網路中傳輸。在 Registrar 上有一個資料庫專門保存使用者的帳號以及密碼，當 Registrar 在做 SIP 認證計算的時候，可以直接從資料庫內取出使用。但是這種機制的假設是伺服器的上的資料絕對不會遭人竊取。對於使

用者來說，自己所擁有的帳號與密碼未被加密而是以明文形式保存在 Registrar 的資料庫中，如果帳號管理的資料庫被不法人士入侵竊取資料、或者是管理人員離職前監守自盜的行為，皆會對帳號使用者造成嚴重的損害。

(二) 非對稱式金鑰系統的概念

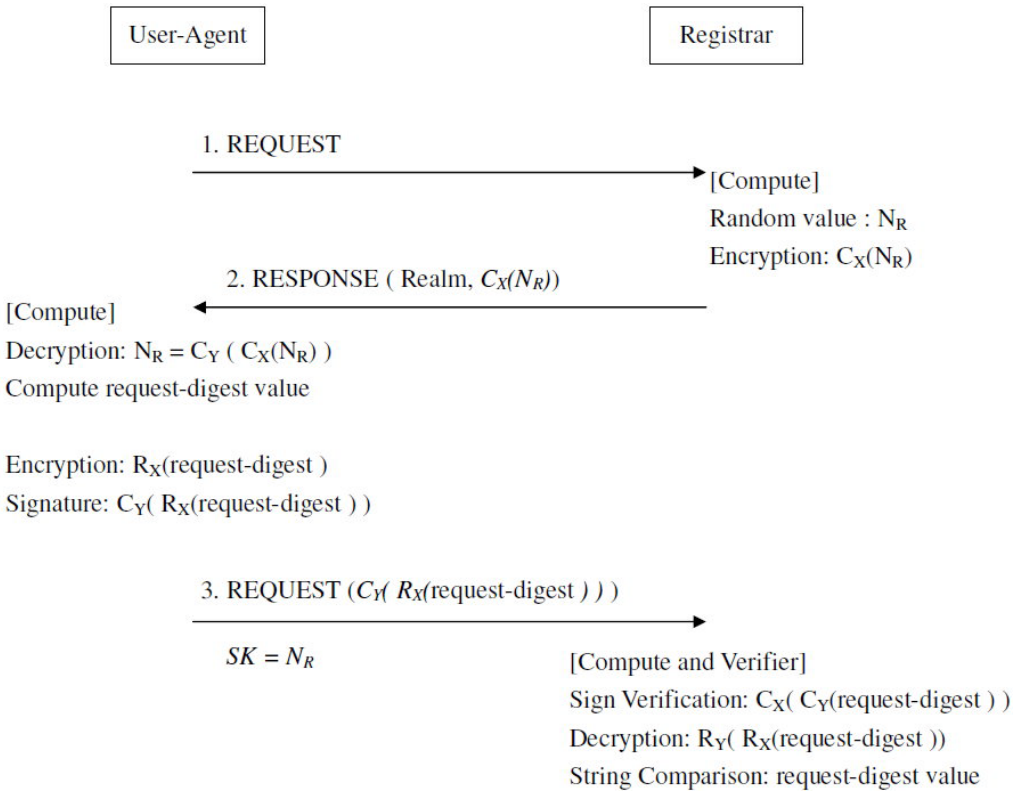
非對稱式金鑰系統也就是公開金鑰密碼系統(Public-Key Cryptosystem)，在公開金鑰密碼系統中所產生的金鑰是一對的，分別是祕密鑰匙(Private Key) 跟公開鑰匙(Public Key)，祕密鑰匙是被用來做解密與簽名，公開鑰匙則是被用來做加密與驗證簽名。其中公開鑰匙是可以被散佈的，並且經由公開鑰匙很難推導出祕密鑰匙。目前比較常見的公開金鑰密碼演算法(Public-Key Cryptography)有 ElGamal[13]、Schnorr、RSA 與 DSA。

(三) 我們提出的改善方法

首先，每個 UA 都有一組公開與祕密鑰匙，而 Registrar 也有自己的公開與祕密鑰匙；UA 與 Registrar 都只知道對方的公開鑰匙，而自己的祕密鑰匙只有自己才知道，並未被公布出來；並且在 Registrar 的認證資料庫內存放的是使用者的帳號名稱與該帳號的公開鑰匙。以下我們將詳述我們所提出改善的方法，表二是我們方法所會使用到的符號說明表。

表二 符號說明表

Symbol	Description
$H( )$	A one-way hash function
N_R	A nonce value be generated by Registrar
C_X	UA's public key
C_Y	UA's private key
R_X	Registrar's public key
R_Y	Registrar's private key
SK	Session key



圖一 我們所提出改善方法的認證流程

Step.1 ***UA → Registrar:*** UA 向 Registrar 發出一個 method 為 REGISTER 的 SIP 註冊請求訊息，告知 Registrar 它要進行註冊。

Step.2 ***Registrar →UA:*** Registrar 收到 UA 的註冊請求之後，使用與 [8] 相同的 Challenge-Response 機制。不同之處在於，Registrar 隨機產生 48 bytes 的 N_R 值後，並使用 UA 的公開鑰匙對 N_R 值做加密，也就是 $C_X(N_R)$。這個經由 UA 公開鑰匙加密後的 $C_X(N_R)$值，即使在公開網路上傳送時被截取，被看到的 $C_X(N_R)$ 值也只會是加密過後的亂碼而已，安全性可以得到保障。$C_X(N_R)$值會夾帶在 WWW-Authenticate 標頭欄位中，接著回覆 401 Unauthorized 的 SIP 訊息給 UA，請 UA 先做認證的計算。如果在 WWW-Authenticate 標頭欄位中沒有使用特別的參數說明，那麼在 SIP 註冊認證演算法中預設的 Hash Function 是採用在 RFC 1321 文件中所定義的"The MD5 Message-Digest Algorithm"[12]。如圖二(1)所示，在這個改良的 SIP 認證方法中，A1'字串裡不需要有 password 參數(也就是使用者密碼)，只需要包含使用者的帳號與 realm 值；A2 字串和[8]一樣，由 Request-Line 所提供的 SIP Method 與 Request-URI 合併成為，如(2)；然後再將雜湊過後的 A1'字串，加上先前產生的 N_R 值與雜湊過後的 A2 字串，合併成一個新字串，再執行一次雜湊函數，如 (3)，即求得 request-digest 值。

A1' = username-value ":" realm-value	(1)
A2 = Method ":" Request-URI	(2)
request-digest = H(H(A1') ":" N_R ":" H(A2))	(3)

圖二 Formulas in generating the request-digest

Step.3 ***UA → Registrar:*** 當 UA 收到由 Registrar 發出的 challenge (401 Unauthorized)，UA 會從此 SIP 訊息中的 WWW-Authenticate 標頭欄位取出 $C_X(N_R)$值，並使用 UA 的私密鑰匙解密，也就是 $N_R = C_Y(C_X(N_R))$。依圖二所示次序，將使用者帳號與 realm 值合併，再進行 hash 後，就得到 A1' 字串；將我們送出註冊訊息中 Request-Line 的 SIP Method 與 Request-URI 合併成為 A2 字串，如(2)；然後再將 hash 後的 A1' 字串，加上先前從 Registrar 回應訊息中解密後取得的 N_R 值與 hash 後的 A2 字串合併，將這個新字串再做一次 hash，如(3)。由於 SIP 的註冊訊息是在公開的網路上傳輸，封包有被截取的可能性，為了避免 request-digest 這個重要的認證訊息被竊聽到，所以我們先使用 Registrar 的公開鑰匙將其加密為 R_X(request-digest)。加密過後，除了 Registrar 的祕密鑰匙可以解密以外，其他中間竊聽者皆無法解密成功。接著，再使用 UA 的祕密鑰匙做一個簽名的動作，用以證明這個加密過後的值確實是由 UA 所發出的。這樣一來，我們就可以確認幾個重要的 SIP 標頭欄位在傳輸過程中有無被中間傳輸者竄改過，以及這份 SIP 註冊訊息是否由該帳號的真正使用者所發出。最後就讓這個簽名過後的值 (也就是 $C_Y(R_X(\text{request-digest}))$)置入 Authorization 標頭欄位中的 response 參數，送出 method 為 REGISTER 的 SIP 註冊訊息給 Registrar。

Step.4 ***Registrar →UA:*** Registrar 接收到由 UA 送出 method 為 REGISTER 的 SIP 訊息後，先用使用者的公開鑰匙 C_X 去驗證這封 SIP 訊息是否由使用者本人所發出，計算 $R_X(\text{request-digest}) = C_X(C_Y (R_X(\text{request-digest})))$。驗證失敗的話，Registrar 會重新回到圖一中的第二個步驟，重新隨機產生 nonce 值，然後再次發送 401 Unauthorized 的 SIP 訊息給 UA，請 UA 重新再作一次認證；若驗證成功，Registrar 就會用自己的祕密鑰匙去解密得到 UA 計算出來的 request-digest 值，檢驗 request-digest =

R_Y(R_X(request-digest))，接著 Registrar 也同樣如圖二內公式所示計算出 request-digest 值。兩者互相比較，如果兩者是完全相同的，那麼 Challenge-Response 即為成功，Registrar 會送出一個 200 OK 的訊息告訴 UA 認證成功。此次認證通過之後，Registrar 在圖一(步驟二)中所產生的隨機字串 N_R 值會被用來當作此次會議的鑰匙(Session Key)；如果兩者比對結果是不同的，Challenge-Response 則是失敗，Registrar 會重新回到圖一的步驟二，重新隨機產生 N_R 值，然後再次發送 401 Unauthorized 的 SIP 訊息給 UA，請 UA 重新再作一次認證。

四、系統實作與效能評估

為了檢視它的執行效能與可行性，我們在一套 Open Source 的 SIP 伺服器上實作我們上述所提出的非對稱式金鑰系統。在我們的實驗環境中，SIP 認證伺服器的硬體部份中央處理器是 Intel Pentium 4 - 3.40GHz，記憶體大小是 DDR400 - 1GB；軟體部分，作業系統是採用 FreeBSD 7.0 [15]，SIP server 是架設 Open SIP Server(Open SIPS 1.4.4)[18]，它是一套成熟並且開放原始碼可以提供我們自行修改系統。關於非對稱式金鑰密碼系統這部分我們是使用 GNU Privacy Guard(GnuPG) [5, 16]，這套軟體提供一些非對稱式加解密演算法可以讓我們建立金鑰對(我們設定的金鑰長度 1024 個位元)，並且執行加密、解密、簽名與驗證簽名的功能，它也是免費就可以取得的。此外，為了可以讓我們直接利用 GnuPG 已經包裝好的功能來快速且方便的開發我們的認證程式，我們安裝 GPGME(GnuPG Made Easy)[17]這套軟體，它讓我們可以直接呼叫並且使用 GnuPG，例如 gpgme_op_encrypt()是被用來做加密的、gpgme_op_decrypt()是做解密、gpgme_op_sign()是做簽名與 gpgme_op_verify()則是驗證簽名。表三顯示這四個函式各執行十萬次所需的秒數平均值，在我們運算這些函式十萬次過程中，我們發現在呼叫這些函式時，它們所用的記憶體有逐漸增多的現象，並且 CPU 的使用率也有上升的趨勢。這些現象導致了最後取得的總秒數與秒數平均值的增加。我們所提出的非對稱式金鑰認證方法，以一個完整的認證流程來說，Registrar 總共會執行了一次的加密、一次解密與一次的驗證簽名；而 UA 會執行了一次加密、一次解密與一次簽名的動作。

表三 GnuPG 上執行各功能所需花費時間

Operation	Execute Time(ms)
Encryption	2.261
Decryption	0.669
Signature	1.492
Verification	0.596

圖三是 UA 向 Registrar 要求進行註冊的訊息內容。Registrar 的 IP Address 是 10.10.59.150，UA 的 IP Address 是 10.10.59.115，此 SIP 訊息的 method 為 REGISTER，要求進行註冊的使用者帳號名稱為 1000。其中 Supported 欄位內的參數值為 AAS（Asymmetric Authentication Scheme），代表本論文所提出的認證方法。Supported 欄位是 SIP 一項非常有彈性的設計，方便 UA 向 Registrar 表明它所支援的衍生功能清單，讓 Registrar 可以選用 UA 所支援的認證方法進行驗證。

```
REGISTER sip:10.10.59.150 SIP/2.0
Via: SIP/2.0/UDP 10.10.59.115:5060
Max-Forwards: 70
From: <sip:1000@10.10.59.150>
To: <sip:1000@10.10.59.150>
CSeq: 1470 REGISTER
Contact: <sip:1000@10.10.59.115:5060>
Expires: 3600
Call-ID: 145236985-58112
Supported: AAS
Content-Length: 0
```

圖三 要求註冊的 SIP 訊息內容

表四 認證欄位說明表

參數名稱	參數的意義
username	要被認證的帳號名稱。
realm	Registrar 提供 UA 認證挑戰之一，代表此認證伺服器所屬的領域。
nonce(圖四)	Registrar 提供 UA 認證挑戰之一，也就是圖一步驟二使用 UA 的公開鑰匙加密過後的 N_R 值
nonce(圖五)	圖一步驟三被 UA 解密過後得到的 N_R 值；其值將等同於 Registrar 在圖一步驟二中隨機所產生 48bytes 的 N_R 值。
response	UA 接受 Registrar 的挑戰的回應值。也就是被 Registrar 的公開鑰匙加密與 UA 的祕密鑰匙簽名的 request-digest 值。
algorithm	所採用的雜湊函數(預設為 MD5)。

圖四是 Registrar 請 UA 進行認證的訊息(Challenge)。表四為圖四中所提供認證欄位內各個參數所代表的意義。

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.10.59.115:5060
From: <sip:1000@10.10.59.150>
To:<sip:1000@10.10.59.150>;tag=0b1e5a175178ff79c4a68bec44cb7f10.e7e4
CSeq: 1470 REGISTER
Call-ID: 145236985-58112
WWW-Authenticate: Digest realm="10.10.59.150",
nonce="hQIOA6YI++Y4MCo8EAf+OwS4pqVlBtyh5s22A7LCyfClG/c3olidPO687/Ozf2LiK9PqWkiXsTO0vGLwmVSYGth77L9yjjzBMBGTf4HpSwj8kYgP7iAdwbEpy0/gLeiKvlH6MWVAPhP5pCdDcb86jFAAZzBYASE/9W3yj5VGnkzkWU6EMkKB4iOHt9FzvLF/NmzagSDw6I+mvE4GRnHfbZXoasy4DH3ODhiE8kHHossHya07PLb4JHQl4MDQkij/D4XPXuVvrYyAwQilFZcy/EQgdTkzHhkQyqbzfrcO+DTZqVOQDy6sgOguws6QrnFRShniX3VKY8ws1R7zmH7UafJuiC3P4F7vwbH21SqS0wf5Ab/mK3n1fH0YzK1dFtEWJUkreIYjek1jh6YkSw3Zpc12X9KJTMJFoM4VoVJglssiKJwKkbZfy4Sk89FBcL/NcEQPWvf0LhuBsoaDk0EcsRf4O0RuMLMj8yxE9EvhQ8MFbcZGdoRdzcQxI0vuJmSN/jCB0MwJCwF5EfvGZhdzx0FwYqe/4gu61tXPPhakrH4L5TX5tJSfwPFU+SFaXL6suPMWTFt03CLV15BS0iK437pP+KmKrb5XuUwRnnEXvkiulaOFK8zSQjaLrYCo1NJ4GI5pCxfCUfCv+eg7s5YpIDrufbhvhfLUd0K3uQDGcM7YhX10nvrJOrVLW09GCcs3QNJmASmy3Z0cn6R0YrW3hmeJ0UsIa5MMvqeQmqxBAgoXusgxPjY1LXKGnQpJk8t5xkLBxWBMZg/BTP1VibO17x8R2A/Crq5jFeswNiRKPBkXNWd1dj0xvgHK/OsLuGcY2IwO27esjz7E=", algorithm=MD5
Server: OpenSIPS (1.4.4-notls (i386/freebsd))
Content-Length: 0
```

圖四 要求 UA 進行身分認證的訊息內容

圖五是 UA 向 Registrar 提供認證資料的訊息。Authorization 欄位的各參數值意義說明，如表四。

```
REGISTER sip:10.10.59.150 SIP/2.0
Via: SIP/2.0/UDP 10.10.59.115:5060
Max-Forwards: 70
From: <sip:1000@10.10.59.150>
To: <sip:1000@10.10.59.150>
CSeq: 1472 REGISTER
Contact: <sip:1000@10.10.59.115:5060>
Expires: 3600
Call-ID: 145236985-58112
Authorization: Digest username="1000", realm="10.10.59.150",
 nonce="4a99f26600000000fa6df12a127209f73aa3ec47bed3d569"
, uri="sip:10.10.59.150",response="hQIOA8dBMQV1NNhaEAf/fvQJLfFlC7amQQu9VK36mgKQotWg7mtm38cYcWZMaIdkx79u7SHJnkVtm5xAbUyvClexZOoiR4coL1hw7tpZjoRhfmtwrAqVcEcnLjAlCY/DYdCaV2tGizVwv2LKlgh3mkcyRwnZ92pM5Vns8E1XNrlZ9nTqR8GVjb2Ex/CzfCg9CcROsvBEkzGyb+5QAYUT6u+2osXRbI0BQvCOdhFcFkD3NgVN+MevUZeUpnXClpz+UmuVNEcFsG129Do2JeQzMwpAX6UkIW/TBLnp4gdBbmDErPPoFftmrOHlauFi8uelFwkJT/AFw/NE8svDM860sQK1A4vZOl/wR1QNHPtUcQgAjCtWrkpLN4JxVBJr3o3dxuO5gtIwCKyIz5ia5wp7z8GMTUuTRAwR4W8lliUNLKu1H0pqf+VUnvngvJJt47qauAygmmC8CVu6X0mCS4uvIX6z+Z8fPRBbRsLM5tGCHe4m07RG2kOoRnwlVmsNs976rRqNwDSOy/JftN2M9rdRx0rXilSvtYg+ewCCrCh+0Oz72/s1HSv2Ji5dkA7qLumYm81zG7klXAVElUtidNjoCK6loTEpTtlWt7BhmAdV44dvUGKHAJeAS/54LJzbIADkqOqvoiNX166RxdN7EhOc7rUUn/NqxBIRQ5lUPXuo6QLu9BZrQ5YIt/gC/Elct4TDNNKqASNB9METPRAz3GDRi32JSxylh5pybczmbkpNsjJXTenhDxw0cP9OAM0rp2+yaKNme6qKp+lwsA2/io06mc5/lhkGInUQIbF7ZWdYrfQITupDTCb2N7Q46eip4ila5Q4Zov2cp5IJrbWQW2NOhpJCePLYGM8KSIgX3zzTe+tXzVahZuu3QWKAi+JRK6m3eVezpl2VpxCC34l+uEBbXAb4GpldduW688o6GeZ8==K9nL", algorithm=MD5
Content-Length: 0
```

圖五 UA 提供認證資料的訊息內容

圖六為 Registrar 傳送給 UA 的 SIP 訊息，表示其帳號認證成功(200 OK)。

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.10.59.115:5060
From: <sip:1000@10.10.59.150>
To:<sip:1000@10.10.59.150>;tag=0b1e5a175178ff79c4a68b
ec44cb7f10.e7e4
CSeq: 1472 REGISTER
Call-ID: 145236985-58112
Contact: <sip:1000@10.10.59.115:5060>;expires=3600
Server: OpenSIPS (1.4.4-notls (i386/freebsd))
Content-Length: 0
```

圖六 認證成功的訊息

五、結論

在本篇論文中，我們提出以非對稱式金鑰的特性來改善現行SIP認證方法中使用對稱式金鑰的缺失，目的是為了要加強使用者帳號管理的安全性，防止管理人員離職前的監守自盜或是認證資料庫遭到不明人士入侵竊取使用者密碼後冒名頂替的狀況，藉以保障帳號使用者個人的身分隱私與財務系統的安全。我們所提出的非對稱式金鑰系統與 SIP 原本的對稱式的認證方式相比，SIP 信令個數是相同的，而加解密的運算次數略增，但是加解密的對象僅是部分較重要的認證字串，而不是對整個 SIP 訊息做加解密。實驗過後發現其所增加的時間低於 10ms，但卻可以提高對使用者安全性的保障。因此將 SIP 改用非對稱式的驗證方式，可避免認證資料庫受到竊取認證金鑰攻擊後，使用者的權益受損，是值得考慮的改進方向。

六、參考文獻

[1] YAHOO!奇摩新聞，"離職工程師「好玩」惡搞 成台哥大跨年夜當機元兇", http://tw.news.yahoo.com/article/url/d/a/090901/17/1q8hs.html，2009 年 9 月。

[2] A. Durlanik, and I. Sogukpinar, "SIP authentication scheme using ECDH", World Enformatika socity Transaction on Engineering computing and technology, Vol.8, pp. 350-353, 2005.

[3] C. C. Yang, R. C. Wang, and W. T. Liu, "Secure authentication scheme for session initiation protocol", Computers and Security, Vol. 24, pp. 381-386, 2005.

[4] Daniel Collins, "Carrier Grade Voice over IP", 2nd Edition, McGraw-Hill, Sep. 2002.

[5] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer, "OpenPGP Message Format", IETF, RFC 4880, Nov. 2007.

[6] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", IETF, RFC 2617, Jun. 1999.

[7] J. L. Tsai, "Efficient Nonce-based Authentication Scheme for Session Initiation Protocol", International Journal of Network Security, Vol. 9, No. 1, pp. 12-16, 2009.

[8] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, " SIP: Session Initiation Protocol", IETF, RFC 3261, Jun. 2002.

[9] KOAA.com, "Government computers hacked; personal info stolen", http://www.koaa.com/aaaa_top_stories/x407184677/Government-computers-hacked-personal-info-stolen, Feb. 2009.

[10] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: Session Initiation Protocol", IETF, RFC 2543, Mar. 1999.

[11] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", IETF, RFC 2616, Jun. 1999.

[12] R. Rivest, "The MD5 Message-Digest Algorithm", IETF, RFC 1321, Jun. 1992.

[13] Taher El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", Proceedings of CRYPTO 84 on Advances in cryptology, pp. 10-18, 1985.

[14] W. Diffie, and M. Hellman, "New directions in cryptology", IEEE Transaction on Information Theory, Vol. 22, no. 6, 1976.

[15] FreeBSD [http://www.freebsd.org/]

[16] GNU Privacy Guard [http://www.gnupg.org/]

[17] GPGME [http://www.gnupg.org/related_software/gpgme/index.en.html

[18] OpenSIPS [http://www.opensips.org/]