國立交通大學

理學院科技與數位學習學程

碩士論文

以資訊隱藏之圖片進行使用者身分驗證之研究

SSL(Simple Steganographic Login)

A Study on Authenticating User Identity

研究生：黃振燊

指導教授：蔡文能　教授

中華民國　九十九　年六月

以資訊隱藏之圖片進行使用者身分驗證之研究

SSL(Simple Steganographic Login)
A Study on Authenticating User Identity

研 究 生：黃振燊　　Student：Chen-Shen Huang

指導教授：蔡文能　　Advisor：Wen-Nung Tsai

國 立 交 通 大 學

理學院科技與數位學習學程

碩 士 論 文

A Thesis

Submitted to Degree Program of E-Learning

College of Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Degree Program of E-Learning

June 2010

Hsinchu, Taiwan, Republic of China

中 華 民 國　九 十 九　年 六 月

以資訊隱藏之圖片進行使用者身分驗證之研究

學生：黃振燊　　指導教授：蔡文能

國立交通大學理學院科技與數位學習學程

摘　要

隨著網際網路的日益蓬勃，各項服務陸續發展，便伴隨著網路攻擊威脅的與日俱增。因此，如何在複雜的網路環境之中驗證遠端使用者的身分，對於服務的提供者及使用者就成了一個非常重要的問題。

目前最常見驗證遠端使用者的方法，都是藉著使用者的帳號和密碼來進行驗證。這種方法的安全性，最重要便是依靠密碼的強度，只要密碼足夠長且具有亂數特性，則使用者帳號與密碼的安全性便會越高。但要使用者將此類型的密碼熟記，是非常困難的。因為大多數人通常都傾向於使用有意義或和個人資料有關的密碼，雖然這樣的密碼易於記憶，卻也容易遭到不懷好意的駭客破解。

因此在本研究中，提出一個利用圖形檔案及資訊隱藏學的方法來進行使用者驗證，讓使用者無需記憶複雜又無意義的長密碼，並同時增加驗證的安全性，也降低使用者帳號和密碼被竊取的風險。

關鍵字：驗證、資訊隱藏、安全性、密碼、密碼破解

SSL(Simple Steganographic Login)

A Study on Authenticating User Identity

Students：Chen-Shen Huang　　Advisor：Dr. Wen-Nung Tsai

Degree Program of E-Learning
College of Science
National Chiao Tung University

ABSTRACT

With the ever expanding internet, there comes the availability of all manner of goods and services. What accompanies these conveniences is the increasing threat of Cyber attacks. How a server authenticates a remote user then becomes a critically important issue for both the servers and the users.

The most common way to authenticate a remote user is through the clients account and password. The most important part of this method typically relies on the strength of the password. As long as the password length is sufficient with random numbers, the security of user account and password will be maximized. However, it is difficult for most users to memorize a password of this type. The most common type of password is usually meaningful or related to the user and thus is easy to be predicted. Unfortunately a password that is easy to remember is also easily hacked by people with bad intentions.

Therefore, in our research, we proposed a preferred method of utilizing graphics and steganography for user authentication. With this method, the user can still have secured Internet services without memorizing long and meaningless passwords. This also reduces the risk of the user account and password being compromised.

Keywords：Authentication、Graphical Passwoard、Steganography、Safety、Compromise

誌　謝

能夠順利的完成這篇論文首先要感謝蔡文能教授不時的指點與啟發，使我能在與眾人討論中探尋出論文研究正確的方向，讓我得以一窺資訊領域的深奧，在研究過程中，從蔡教授的指引下讓我學到了嚴謹的研究方法和良好的學習態度，實在獲益匪淺！

一路跌跌撞撞的走來，亦得感謝的好友文洋在程式設計上的解惑與大力相助，提供相關的經驗，使我能獲得不少寶貴設計方法，另外也要感謝同實驗室中茂南、興能、瑛旗、丁榮等同學的協助，總能夠相互扶持、勉勵，讓我在研究遇到瓶頸時得以努力向前。

最後當然要感謝身旁親友的支持與鼓勵，更是我研究得以繼續前進的最大助力，讓我無後顧之憂的浸淫於資訊學術領域中；若沒有眾人的體諒及協助，相信我的研究所生涯將會是很不同的光景，再次感謝所有給我協助與支持的每一位，感謝您。

目　　錄

表目錄

圖目錄

第一章 緒論

由於密碼記憶的困難和電子設備的輸入方式不便利，所導致的結果便是使用者通常都選擇安全性較弱但容易記住的密碼來使用，例如常用詞語或簡短的個人相關資訊來作為密碼，以上因素都使得服務系統以及個人重要資訊暴露在可怕的安全性威脅之下。

因此，如何建構一個安全而有效的認證系統，在現今網際網路世界上，不論是網路服務提供者或是對於使用者而言都是很重要的。安全性高的認證系統早已誕生，如虹膜辨識系統或指紋辨識系統，但其昂貴的造價卻讓許多服務提供者卻步；安全性高且花費較便宜的認證系統，也使用已久，如智慧卡或圖形化密碼認證，但其使用的便利性，卻造成使用者無意變更熟悉已久的帳號密碼式認證系統。於是，如何讓服務提供者花費降低，有足夠的便利性讓使用者願意改變，有夠高的安全性效能可以確認雙方身分，便是本研究的主要研究重點。

改進現今主流的認證系統模式，降低其遭受惡意攻擊的可能性，提高使用者使用的便利性，減少服務提供者的建置成本，而且又可以達到良好的認證效果，是 SSL(Simple Steganographic Login)圖形驗證法的主要訴求。

1.1 研究背景與動機

各式各樣的網路服務已無所不在的充斥於我們的生活之中，透過隨手可得的資訊工具如：手機、PDA 或手提電腦，再加上日益便利的無線網路，幾乎是時時刻刻都可以使用所需的網路服務。然而，在現今網路世界中，伺服器所提供的服務通常都是藉由使用者的帳號、密碼進行使用者身分認證，藉此辨別使用者的身分是否正確後，方提供各項系統中的服務。

對於網路服務日益發達的現代生活，使用者認證已成為不可或缺的要角。因此如何確認使用者與服務端雙方的身分以及怎麼樣提高雙方的通訊安全性，降低重要資訊被盜取的機會，儼然為資訊時代一個重要且必要的議題。但網路通信會被竊聽，私有的智慧財產可能會被駭客盜取，所以如何達成有效確認使用者的身分和確保通信的可靠性，便愈顯重要。

可是經過了幾十年的資訊發展歷程，使用者帳號、密碼被竊取的問題仍層出不窮，而使用者在密碼的選擇上仍舊偏好簡單易記的密碼，但是這樣類型的密碼，卻很容易遭到惡意攻擊而破解，所以如何讓使用者選擇好的密碼，成了提高安全

性不可或缺的要角。

1.2 研究目的

由於網際網路縮短了地球村的距離，許多商業交易都可以在網路上進行，而對於資訊安全所知不多的消費者，幾乎就成了駭客的待宰羔羊。駭客透過了許多的軟體漏洞或是其他不法途徑，竊取使用者網路系統服務的帳號、密碼，甚至於重要資訊。因此如何提高使用者使用諸多服務的安全性，便顯得日益重要。而透過資訊隱藏技術來做為保護登入資訊等機密資料，亦是提高駭客竊取重要資訊難度的有效方法。

已有許多研究者提出，以智慧卡進行使用者認證，並達到很高的安全性能。但唯一美中不足的是，其相關設備的取得、價格和便利性，以及智慧卡在初次註冊傳遞上的時間成本；而新的圖形化密碼認證方法，提高的駭客破解使用者帳密的難度，甚至是指紋辨識等生物科技式的認證方法，也都有非常好的安全性效果，但為何仍舊無法讓系統服務者或使用者大量採用呢？是價格因素或是使用者便利性的問題。

本研究是探討如何透過資訊隱藏在圖形檔案的方式，來加強使用者認證的安全性；因對於服務端而言，複雜度高且長度足夠長的密碼，才能符合服務端對於安全性的需求；但對於使用者而言，容易記憶與使用的便利性才是重要的。因此本研究希望同時達到系統服務端和使用者端的需求，可以讓使用者迅速且方便的記憶下自己的密碼，而系統服務端又可以得到良好的安全性效果的目的。

1.3 論文架構

本論文共分成六個章節，第一章為研究動機與目的，第二章為相關背景知識介紹，包括使用者認證介紹、資訊隱藏學(Steganography)、圖形化密碼(Graphic Password)、智慧(晶片)卡簡介等。第三章則對與本研究相關的期刊和論文進行分析探討，共分成圖形化密碼認證、智慧卡認證和密碼安全性等三節。接著在第四章提出本研究的設計架構和概念，並作詳細的說明。於第五章進行系統環境與成果介紹，也與不同的認證方法進行分析和優缺點比較。第六章則為本研究做出結論及討論未來的研究方向。

第二章 背景知識

本章將介紹論文相關的背景知識及技術，首先在2.1節介紹使用者認證的概念與運作模式；在2.2節對資訊隱藏學的概念和分類作一簡單說明；於2.3節介紹圖形化密碼認證方法的技術知識；2.4節則對加解密函數與雜湊函數做介紹；最後在2.5節將說明智慧卡的概念與應用。

2.1 使用者認證

一直以來，如何在網際網路上確認遠端使用者的身分，是網路安全中十分重要的研究議題，而當遠端使用者基於獲取網路上伺服器的某項服務而進行登入時，遠端使用者首先要通過伺服器的驗證程序，方能取得服務的使用權，在網際網路蓬勃發展的現今，使用者認證的重要性也愈顯重要。而使用者認證的過程和運作程序，將由下列二個小節來說明。

2.1.1 使用者認證及運作方式

使用者認證的運作通常是由幾個重要的步驟構成(如圖1所示)：1.使用者端向伺服器端提出登入請求。2.伺服器端回應使用者，並要求使用者提供認證資訊。3.伺服器檢查登入認證資訊是否正確。4.認證資訊正確，則通過認證；反之，則不予登入。

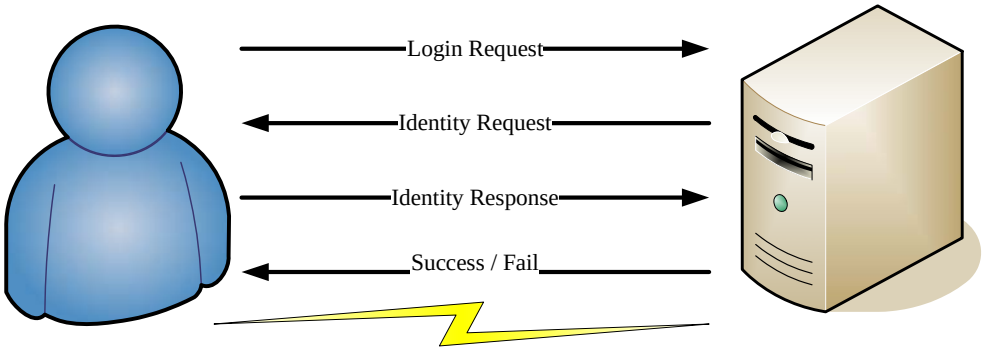

圖 1 認證示意圖

使用者認證的最主要目的即是確認該服務使用者的身分，因此如何得到強而

有效的認證(Strong authentication)便成為認證過程的一大重點—所謂強而有效的認證，乃是指在認證的過程中，安全性很高並且透過多方又具公信力的單位，來確認該使用者的身分證切確無誤。於是各式各樣的認證方式，都朝著有效確認使用者身分的目標邁進，並期許能降低認證過程中所遭受的安全性威脅。

在多年的網路服務發展之下，許多的身分認證方法隨之誕生，目前網路上驗證使用者身分的方法，可以區分為以下的三種模式[1]：

1. 單向驗證(One-Way Authentication)

   此方式乃是目前網路上最常見的驗證方法，由於此驗證方式簡單且容易實作，因此網路上大多數的服務系統都是使用此方式來驗證遠端使用者的身分。在單向驗證模式中，遠端使用者必須事先在伺服器上註冊自己的帳號與密碼，一旦當遠端使用者想要登入伺服器時，遠端使用者就必須提供所註冊的帳號與密碼給伺服器，以供伺服器進行驗證的確認動作，由伺服器判斷遠端使用者是否核可登入系統或使用該伺服器所提供之服務。但是，此驗證方式有一重大缺點，即是遠端使用者無法辨認伺服器是否真實且合法，故只要有網路攻擊者偽裝為一台合法的伺服器時，遠端使用者不但無法察覺，更會將使用者的重要資訊完全提供，所以本方式存有著伺服器偽裝攻擊(Server Spoofing Attack)的安全性問題。

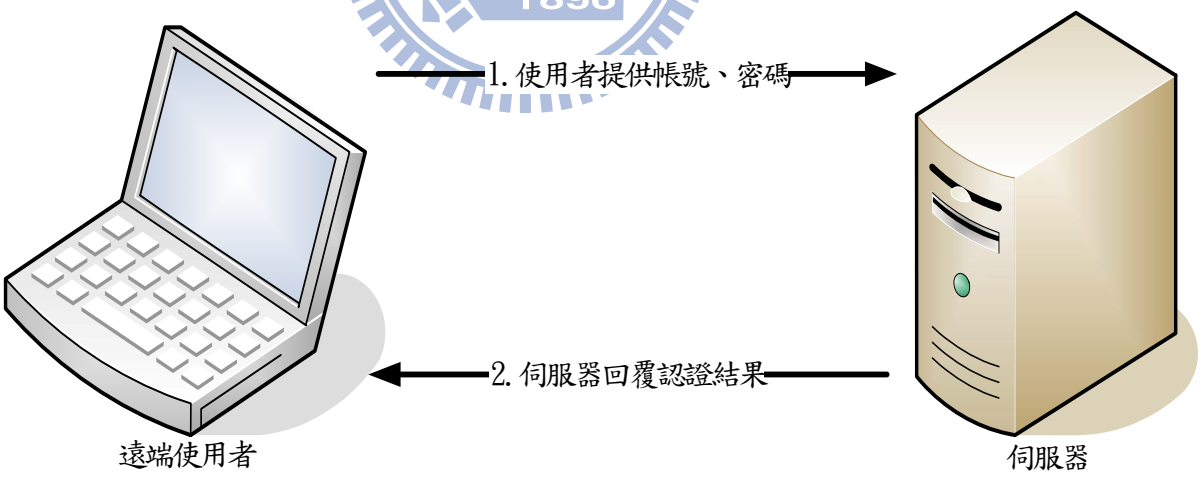

圖 2 單向認證

2. 雙向驗證(Mutual Authentication)

   由於前述的單向驗證方式，存在伺服器偽裝攻擊的重大安全問題，因而有了雙向驗證的概念提出，以解決可能產生的伺服器偽裝攻擊之缺點。雙向驗證的進行方式，便是要求遠端使用者以及伺服器，雙方都提供自己的認證資訊

給予對方，以確認雙方彼此的身分都是正確無誤。如此一來，便可以有效解決因伺服器偽裝而產生的問題—遠端使用者不知道伺服器是假的，而提供自己的帳號與密碼給偽裝伺服器。但是事實上，雙向驗證的安全性仍舊存在著一些安全性問題，此驗證方式存在一個不容小覷的安全性問題，即使用者如何保存遠端伺服器的認證資訊，以利往後的身分驗證呢？因為萬一遠端使用者所保存的伺服器的認證資訊不幸遺失，或是被合法使用者所盜用時，反而提供了攻擊者，可以假冒合法的伺服器去欺騙其他的使用者的便捷方法。故為了保護伺服器的認證資訊，常利用智慧卡來存放伺服器的認證資訊，以避免一個攻擊者在取得合法使用者的智慧卡時，盜取這些認證資訊來假冒成合法伺服器，而且更安全的作法是每個遠端使用者所擁有的伺服器認證資訊皆不相同。

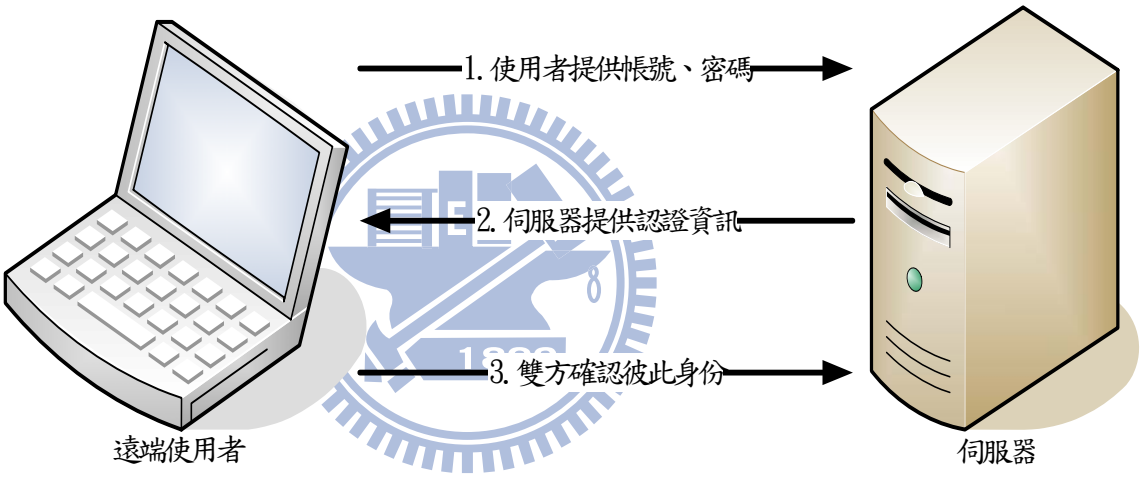

圖 3 雙向認證

3. 使用公正的第三方驗證(Trusted Third-party Authentication)

使用公正的第三方驗證所要達成的目的，和前述的雙向驗證相同，希望可以讓遠端使用者和伺服器雙方驗證彼此的身分是否合法，不過使用了另一種方法來達成此一目的。此法是在驗證的程序中，加入了一個第三方公正單位，此一公正單位特點是，遠端使用者和伺服器端都信任他，因此藉由這一個雙方都信任的第三者，來協助驗證的程序並確認彼此的身分。當遠端使用者希望獲得伺服器的認證時，遠端使用者和伺服器端都必須先通過第三方公正單位的認證，當身分獲得第三方公正單位認可時，第三方公正單位會核發一認證資訊給要求認證者，於是遠端使用者和伺服器端，利用自身取得的認證資訊來讓通訊的雙方確認合法身分。這個方法雖然安全，但仍有一明顯的缺陷，即第三公正單位的安全性將顯得非常重要，一旦此公正單位遭受惡意攻擊者的入侵，則可能導致所有的伺服器與使用者，都暴露在不安全的環境之下，

而且所有使用者與伺服器均渾然無所覺。

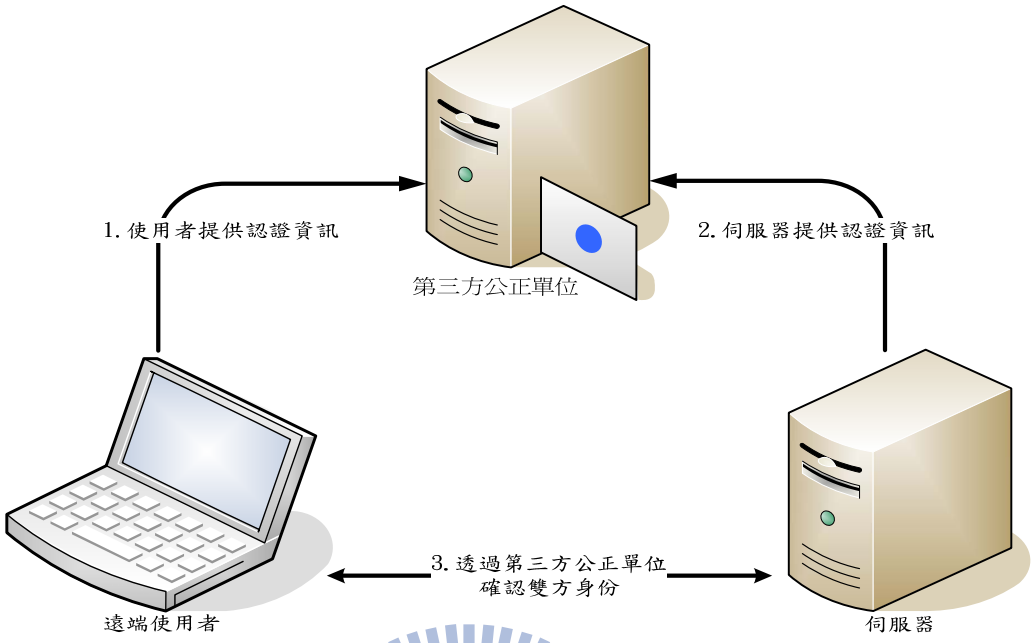

圖 4 公正的第三方驗證

2.1.2 使用者認證技術分類

現行眾多的使用者認證方法，根據Rachna Dhamija及Adrian Perrig[2]兩位學者研究，將使用者認證方法依技術分類成以下三類：Knowledge based authentication, Token based authentication 和 Biometric based authentication 。

1. Knowledge based authentication(以使用者個人資訊為基礎的認證)
   此種方式是透過每個使用者擁有的不同個人資訊的前提下，做為認證的一種方法，例如國人所擁有的身分證字號或是使用者自行建立問題與答案等，雖然這個方式很容易實作，但它依賴的安全性基礎是建立在連接遠端使用者和伺服器之間的網路通道必須是安全且可信任的。但是，現實的網路環境並非如此，現實的網路環境充斥著各種威脅，而遠端使用者和網路伺服器之間的通訊，是極容易被有心人士所竊聽的。

2. Token based authentication(以憑證為基礎的認證)
   此種方式是利用如智慧卡等憑證當做媒介來進行認證的方法，許多實際應用的此類認證方法通常會搭配Knowledge based的技術來提高其安全性，例如

ATM 提款卡再使用上都會搭配一組 PIN 碼使用，來降低冒充身分使用或是卡片被竊取的安全性問題，也提高使用的安全性。

3. Biometric based authentication(生物識別科技的認證)

生物識別方式的認證，例如指紋辨識、瞳孔辨識或是臉部特徵辨識系統都屬於此種認證方式，雖然這種方式是上述三項技術中安全性最高的，但是由於系統造價昂貴及辨識時間等問題，也造成了此種方式並沒有被廣泛使用。另外，這種類型的認證系統也有非法用戶通過認證(IPR)和合法用戶無法通過認證(FAR)[3]的缺點。

而在當下的認證系統中，最被廣泛使用的仍屬 Knowledge based 的認證方法，雖然其安全性並非最高，但由於實際應用上相對簡便，又無需昂貴的額外設備，因此多數的系統還是採用 Knowledge based 為技術基礎的認證方式。

2.2 資訊隱藏學

資訊隱藏學(Steganography)是一門與密碼學(Cryptography) 不同的學問，雖然這兩者有同樣的目的，但所使用的方法卻有很大的差異。

密碼學通常以改變訊息的內容為方式，讓除了收件者以外的其他人都無法解讀。而由於收件者握有解開加密訊息所需的「金鑰」，金鑰可以將看似無法解讀的訊息解譯成為可以了解的內容，藉此收件者才得以瞭解寄件者所要傳達的訊息。由於加密後的訊息並非肉眼看不見的，況且加密後的訊息，在送進或送出的過程中都可能受到監視或偵測。一旦第三者找出加密的方法或是取得解密金鑰，機密訊息便會被破解並讀出。

資訊隱藏和密碼學不一樣，它隱藏訊息的方式，會讓有心的第三者根本察覺不到有訊息的交換或是自然而然的忽略它。我們可以將資訊隱藏想成一種非常特別的加密方式，資訊隱藏後的資訊是無法簡單偵測的，使得人眼只看得見一般的資料，而看不到隱藏的加密訊息，便會很自然的將其視為普通的資訊，絲毫都不會想到其中隱藏有秘密的資訊。

故由上述簡介可以得知，資訊隱藏和密碼學有一個重要且明顯差異，密碼學所強調的是在難以破解的加密方式，因此透過密碼學所產生的結果，會看起來不甚自然，而引起他人懷疑。但資訊隱藏學，則是去除了此項引人疑竇的因素，來達成加密的效果。於是在現今的應用上常常利用兩種隱藏訊息的方法相輔相成，利用資訊隱藏來輔助加密。透過密碼學的加密方法，再結合資訊隱藏技術，將加

密後的結果「隱形」，便能非常有效的保護資訊而不讓資料間諜得手。

如圖 5 是資訊隱藏的一個應用範例，利用圖片檔 a 內隱藏圖片檔 b。其中圖 5 (a)是把秘密影像(Secret image)藏入掩護影像(Cover image)後的影像檔，被稱為偽裝影像(Stego image)。圖 5 (b)是從偽裝影像中利用技術性資訊隱藏的技術所恢復的秘密影像。這種技術性資訊隱藏的方法，不論是藏入資訊後的偽裝影像，或是從偽裝影像恢復出的秘密影像，都擁有很高的影像品質，而且肉眼根本無法判別偽裝影像中藏有秘密影像，如同圖 5 所示。

200 X 200 pixels, file size: 88 KB　200 X 200 pixels, file size: 19 KB

(a)Image of a tree. (stego image)

(b)Image of a cat extracted from the image of a tree. (cover image)

圖 5 資訊隱藏圖例
資料來源：Wikipedia [4]

2.2.1 資訊隱藏的特性

而這樣特別的資訊隱藏方法有許多特性，通常分成以下六大類：

1. 不可察覺性(Imperceptibility)：
   係指常人無法直接藉由感官能力，察覺該資料是否藏有秘密訊息，此特性常與人類的視覺或聽覺特性有關。

2. 強韌性(Robustness)：
   係指經過資訊隱藏技術後的資料，在一般使用情況下(如檔案格式轉換、失真壓縮、數位類比轉換.....等)或是蓄意破壞後(如檔案編修、變形、剪裁.....等)，還是可以有效的恢復原本隱藏的秘密訊息。

3. 容量(Capacity)：

係指在相同的隱藏資訊條件下，可以藏入愈多的秘密資訊則愈好。

4. 不可偵測性(Undetectability)：
係指在統計分析下，無法分析出資訊隱藏技術的特性，降低暴露秘密通訊的可能性。

5. 安全性(Security)：
係指資訊隱藏方法不會輕易的被發現，縱使經偵測發現有資訊隱藏方法的使用，有心人也無法輕易的破譯內嵌的秘密資訊。

6. 有效及簡易性(Efficiency and Simpleness)：
係指可以簡便或迅速的將希望隱藏的資訊，隱藏成所需的樣貌。

上述的幾項特性雖然都很重要，但卻存在著某些互斥的取捨（Trade-off）關係，例如當我們要求強韌性效果要高時，則所隱藏的訊息容量便只能降低；反之要求隱藏的訊息容量要高時，強韌性的效果則會下降。因此在面對不同資訊隱藏需求時，會制定合適的隱藏方法，也造成目前沒有一個資訊隱藏方法，可以同時滿足所有的特性[5]。

2.2.2 資訊隱藏的分類與應用

根據希臘歷史學家希羅多德的著作《歷史》中所記載，最早的資訊隱藏方法[4]，可以追溯到西元前440年，一位希臘人狄馬拉圖斯（Demaratus）在波斯的蘇薩城（Susa），無意間得知了波斯人將進攻希臘的消息，便利用木板刻下戰爭訊息，然後用一層蠟將木板上的字遮蓋住，立即派人將木板送往自己的國家，藉此揭發了人波斯的不軌計謀。最後，波斯海軍戰敗並覆沒於雅典附近的沙拉米斯灣（Salamis Bay），這就是歷史上有紀錄的最早使用資訊隱藏的實例[4]。

由於科技的發展，資訊隱藏技術的進步也十分迅速，已經有許多數位檔案的類型，可以嵌入祕密資料在內，這些數位檔案包含了影像檔、電影檔、聲音檔以及文字檔等，也包括把機密訊息隱藏在網路上的影像或電腦內的檔案之中。一般來說我們將資訊隱藏的技術分成以下幾大類的應用，如圖6所示Fabien, Anderson, & Kuhn(1999)三位學者將資訊隱藏學與其應用所做分類[5]。

資訊隱藏的技術由於迅速的應用和發展，已有許多分類(如圖6所示)，其中隱蔽通道 (Covert Channel)是一讓管理者無法察覺的秘密通道，最常被應用在木馬

程式的操控端與受控端的溝通上，經常是被不法的第三者用來竊取企業或是個人重要資料的管道。匿名技術（Anonymity)則可以保護個人資料在網路或無線通訊傳遞時不會被截取，進而達成保護個人私密資料不被竊取，甚至使其他人無法追蹤到發送訊息的來源的技術。版權烙印（Copyright Marking）也稱作數位浮水印(Water Marking)，則是常應用在版權的保護方面。在數位浮水印應用上，其所著重的是利用藏入的資訊(數位浮水印)來做所有權的證明，這樣才能確保當數位檔案遭受複製、修改或剽竊時，依舊可以從中取出原本的數位浮水印來證明這個數位檔案的權利歸屬。也由於版權證明的需求不同，而有強韌性烙印(Robust Copyright Marking)與細緻性烙印(Fragile Copyright Marking)的區分。強韌性烙印(浮水印)是指烙印圖片經過影像處理後仍能有效偵測出原烙印來，也有應用在點對點分佈系統的指紋技術(Fingerprinting)，是一種專門用來驗證隱藏字串資料的浮水印。以及浮水印技術(Water Marking)是應用在影像方面，其分為可視性浮水印技術(在偽裝影像上可看見浮水印圖像)和非察覺性浮水印技術(在偽裝影像上看不到浮水印圖像)。細緻性烙印(Fragile Copyright Marking)則被專門用來作為防止竄改(Tamper)與進行驗證（Verification)之用，其所嵌入的浮水印，只要經過修改的動作則會很明顯的產生差異。最後，資訊隱藏學(Steganography)則是一種將重要的訊息隱藏到掩護媒體(Cover Media)中，產生一偽裝媒體(Stego Media)，而偽裝媒體在傳輸的過程中，不會引起其他人懷疑的一門學問。

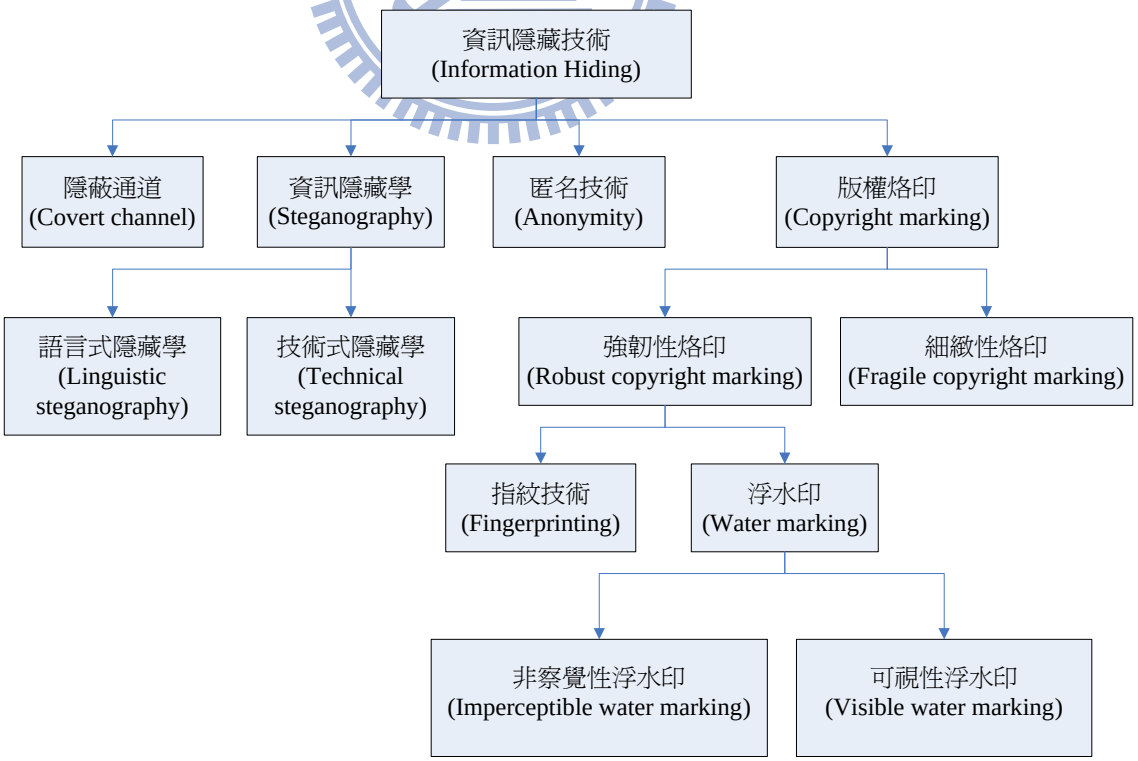

圖 6 資訊隱藏的分類

資料來源：Information Hiding—A Survey [5]

資訊隱藏學(Steganography)在應用上又可分為語言式隱藏學(Linguistic Steganography)和技術式隱藏學(Technical Steganography)兩類型。語言式隱藏學是把欲傳遞的重要資訊很巧妙的排放在一般的文字或文章之中(如圖7，德軍在二次世界大戰中傳遞的訊息)，只要將每個單字的第二個字母取出，便可以得到圖7下方的隱藏訊息，由於他人不知其排放的規則，所以不能輕易解讀出該文字或文章中隱含的機密訊息，因此隱藏於其中的機密訊息得以受到安全地傳送到接收者，但語言式隱藏的方法還是可能被有心人士得知規則而破解，而其最大缺點則是所隱藏的訊息量無法太多，故在應用上有所侷限。

Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils.

Pershing sails from NY June 1

圖 7 語言式隱藏學實例

本研究所提出的SSL圖形化驗證方法，將會應用資訊隱藏學中技術性隱藏的方法，此部份則留待第三章再以相關論文研究作深入討論。

2.3 圖形化密碼

相較於透過使用者提交用戶帳號和密碼的認證方式，圖形化密碼(Graphic Password)是一種新類型的認證技術，藉著對圖片媒介進行點擊等動作，以改進許多存在的安全性問題，提高使用者的便利性且加強認證的安全性[6]。目前已有被應用在PDA、ATM等系統上，雖然未有真正的商業用途應用，但由於安全性提高、使用者和善度增加等因素，所以未來有很大的發展空間。

由於Knowledge-based的使用者驗證方式，存在許多人性因素的缺陷，因此多位研究者提出了，對於使用者容易記憶的方式—圖形化密碼。圖形化密碼的使用者驗證已有許多方法被提出，根據其認證機制的不同，常被分為以下兩大類：1. 回想基礎技術(Recall-based technique)。2. 辨識基礎技術(Recognition-based technique)。或者是依據其使用圖片數量，也可分為grid-based schemes, single-image based schemes和multiple-image based scheme by the features三種。以下將對回想基礎技術和辨識基礎技術做簡單的介紹。

2.3.1 回想基礎的技術

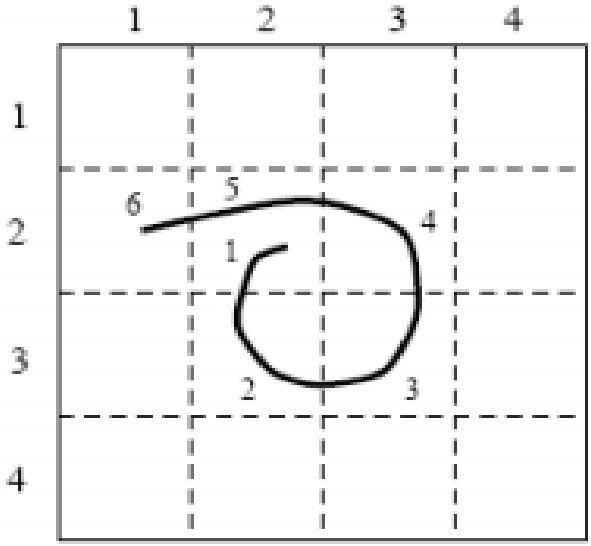

圖 8 D-A-S 系統示意圖

資料來源：A sample of DAS password[8]

回想基礎技術(Recall-based technique)的方式為要求遠端使用者在登入時，依靠使用者的記憶，重做他或她在註冊階段所繪製或選擇的圖形的步驟，以確認遠端使用者的身分。圖8為此項技術基礎的D-A-S方法示意圖，此法透過在一個四乘四的方格陣中，繪製一個圖案來進行認證，而這個圖案是由使用者在註冊階段時所自行建立的。

或是另一種圖形化密碼類型，要求使用者在一張圖形畫面上，按順序點選數個地方作為密碼，此類型的方法以PassPoints[7]為例(如圖9)。

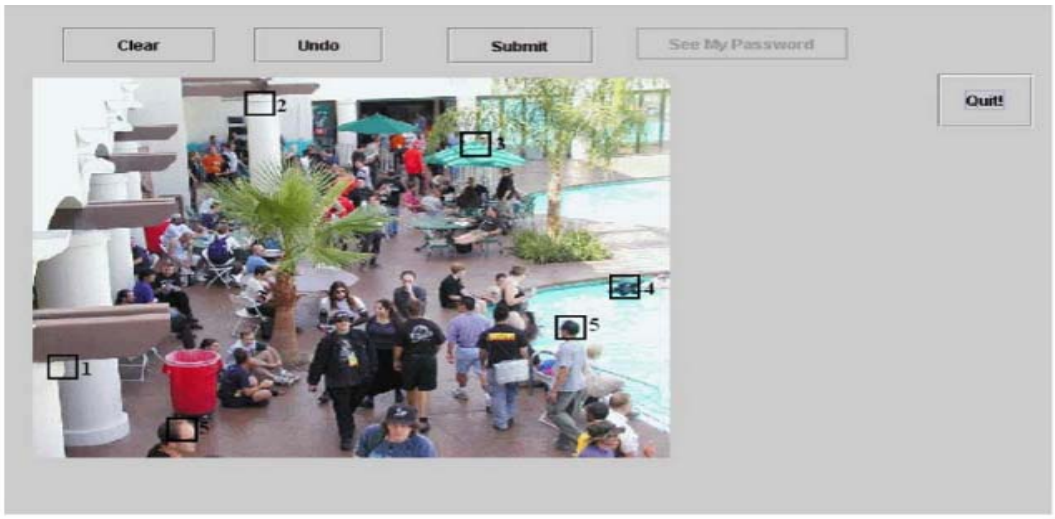

圖 9 PassPoints 系統使用畫面

資料來源：Example of graphical password with click order displayed.[7]

2.3.2 辨識基礎技術

辨識基礎技術(Recognition-based technique)，乃是建構在使用者辨識於註冊階段，所選取的一組圖片或圖像，透過正確的點選圖形作為認證資訊，向系統進行身分認證。如圖10，Déjà Vu系統即是一個例子，在系統提出的許多圖片中，點選使用者先前所選取的圖片，經過幾回合的圖形點選來進行認證。由於辨識基礎技術的認證模式，是透過滑鼠的點選取代鍵盤的輸入密碼，因此可以有效的抵擋字典攻擊(Dictionary Attack) [8]。

雖然有些回想基礎技術的認證方法可能會遭受到字典攻擊，不過其破解的難度也相對於數字字母式密碼方法提高。因此整體來說，圖形化密碼的認證方式相較於數字字母式密碼的認證方式而言，是有比較好的安全性效果[1][8]。[1]

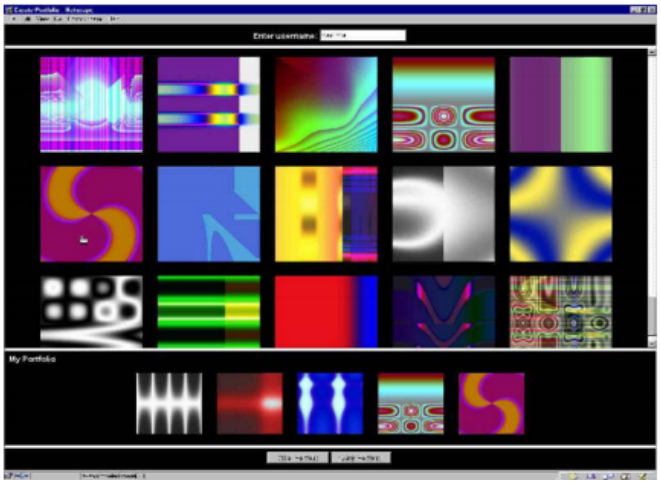

圖 10 Déjà Vu 系統圖形認證畫面

資料來源："Déjà Vu: A User Study Using Images for Authentication"[2]

由於人類生來對於圖像的辨識和記憶能力就比較好，對於正常人而言，分辨出不同的圖案、人臉或是地圖，都只需要很短暫的時間就可辨到；但對於電腦要判斷或分辨這些圖形而言，卻需要耗費大量的計算時間。因此，使用圖形化密碼的好處便顯而易見；使用者可以快速的記憶圖形化密碼，而企圖攻擊者卻要花上大量的電腦運算時間才可能破解。此外，圖形化密碼對於字典攻擊與自動化測試攻擊也有很好的防禦效果，雖然使用者認證的時間會較以往認證方式來的稍長一些，或許也可能需要額外的硬體設施支援；但隨者硬體元件的技術發展，和使用上的優勢，相信假以時日，圖形化密碼將會在使用者認證上佔有重要的一席之地。

2.4 加解密函數與雜湊函數

由於要保護資料，為了增加其安全性和隱密性，通常會藉由把所有人看得懂的資訊內容轉換成一堆看似無意義的代碼，此一過程經由數學的運算，便可以快速的達成，而這些轉換後的混亂代碼，在不知道該數學反向運算的情形下，要利用猜測的方式，達到解讀原本訊息的效果，是非常艱難的。這種由可懂的訊息經由數學運算變換成混亂的代碼過程，便可以視為加密；反之，將混亂的代碼藉由原數學運算的反運算，還原為易讀訊息內容的過程，則可稱為解密。在實作上，經常會利用數學函數運算來達成這樣的效果。

加解密的方法可以分為兩大類的運算系統：第一類為對稱式加密(Symmetric Encryption) ，第二類為非對稱式加密(Asymmetric Encryption)。然而，更有一種將不固定長度訊息轉換成一串固定長度的亂數訊息的運算，稱為雜湊函數(Hash Function)，以下將為這些函數一一介紹。

2.4.1 加解密函數介紹

首先對加解密運算系統中的對稱式加密做介紹，對稱式加密(Symmetric Encryption)又稱秘密金鑰加密(Private-key Encryption)，是指利用一把金鑰，同時在加密與解密的過程中使用，這把金鑰必須妥善保管；而依其運算的過程不同，又可以再分為兩類：一是串流式加密(Stream Ciphers)，另一種是區塊式加密(Block Ciphers)[9]。進行串流式加密時，每次只加密資料中的一個位元或是位元組，直到所有資料加密完成才宣告結束。而利用區塊式加密時，會先將需要加密的資料，以固定大小區分為很多小區塊，然後將各個區塊進行加密或解密動作，例如把所有的資料，每64個位元分割成一個區塊，萬一最後一個區塊不足64位元時，則進行字元填充讓該區塊達到64位元，再進行區塊加密動作。

對稱性加密的特色在於加解密的速度十分迅速，所以常用來當作一般的加密工具，但是它優點也成了它的缺陷，因為它快速的運算速度，所以只要利用窮舉法(Brute Force)來破解此加密法也是十分容易。可是，它還有一個十分嚴重的缺點，就是使用者跟伺服器都沒有辦法利用它，來判定的雙方身分是否正確，因此它還是不能有效的提供我們高度安全的使用環境。於是，在實際應用上我們常會將它搭配非對稱式加密法(Asymmetric Encryption)一起使用，藉由兩種方法的優點來確保資料的安全性[9]。

目前比較具代表性的對稱式加密法乃是最被廣為使用的美國國家標準與技術

研究院（NIST）於2002年正式推出的下一代密碼演算法AES(Advance encryption standard)及之前著名的DES(Data Encryption Standard)密碼演算法都屬於此一類型的加密方法。

非對稱式加密法(Asymmetric Encryption)又稱為公開金鑰加密(Public-key Encryption)[10]則是利用一組的金鑰對(Key pair)來進行加解密運算，每一組金鑰對都包含兩把相互對應的金鑰，一把是可以公開的金鑰(簡稱公鑰)，一把是必須保持秘密的金鑰(簡稱私鑰)，而且很難藉公鑰利用運算推導出私鑰。

非對稱式加密法在使用上來說，通常是將公開金鑰開放給其他的人使用，祕密金鑰則由使用者自己使用。公開金鑰可以解開利用私密金鑰加密的文件，亦即其他人可以解密使用者利用私密金鑰加密的文件，而其他人如果想要傳送加密的文件給原使用者，則只要利用公開金鑰加密該文件，再送予使用者，使用者便可利用私密金鑰進行解密動作。而現在使用的最廣泛、最常見非對稱性加密法便是RSA演算法。

非對稱式加密法的最大優點在於金鑰安全性的提高，可以將公鑰公開給予所有人知曉，無須擔心如對稱式加密法的金鑰傳送安全問題，只需注意私鑰的安全性即可。不過非對稱式加密法的運算法較為複雜，因此有著運算速度慢的缺憾。但這種方法也有另一個好處，由於金鑰對(Key pair)的特性，只要是利用公鑰所解密的文件，即可十分肯定該文件的發布人是同一組金鑰對之私鑰持有人無誤，因此在現實中，也常利用此一類的演算法(如RSA)來提供數位簽章的服務。

表 1 對稱與非對稱加密法比較

	對稱式加密法	非對稱式加密法
加解密金鑰	相同	不同
金鑰可否公開	不可	公鑰公開，私鑰不公開
金鑰保管	與K人交換訊息，則需保管與傳遞K把金鑰	不論與任何人交換訊息，只需保管私密金鑰
加解密速度	快	慢

2.4.2 雜湊函數介紹

雜湊函式（Hash Function)是一種毋需金鑰的加密技術，其概念類似人的指紋，利用雜湊函式來產生該資訊獨一無二的「數位指紋」(Digital fingerprint)，也就是所謂的雜湊值（Hash Value）。典型的雜湊函式都擁有無限定義域(任意長度的輸入字串)和有限的值域(固定長度的輸出結果)以及下列幾項特性，如果輸入一資料運算而得到雜湊值後，將原資料的任何一小部分內容改變，再求一次雜湊值，將會得到一個完全不同的雜湊值，此為雜湊函式的擴張性（Diffusion)；因此如果經雜湊運算得到兩個不同的雜湊值，我們便可以肯定這兩個雜湊值的原始輸入一定不相同。而另一特性—抗碰撞性（Collision Resistance)，則是雜湊函數還必須具備能夠讓每份不同資訊所產生的雜湊值都是唯一且相異，如果想要找出兩份資訊輸出得到相同的雜湊值，在計算上是不可行的。所有雜湊函式除了具有上述特性外，還具有不可逆的單向(One-Way)特性，故雜湊函式也稱為單向雜湊函式(One-Way Hash Function)，此處單向的意思是指，當輸入資料經過雜湊函式運算得到雜湊值，是沒有辦法進行逆運算，還原成原本的資訊；因此，沒有辦法利用運算得到的雜湊值來反推產生原始的輸入訊息。在使用上常見的雜湊函數有MD5 及 SHA（Secure Hash Algorithm ）。

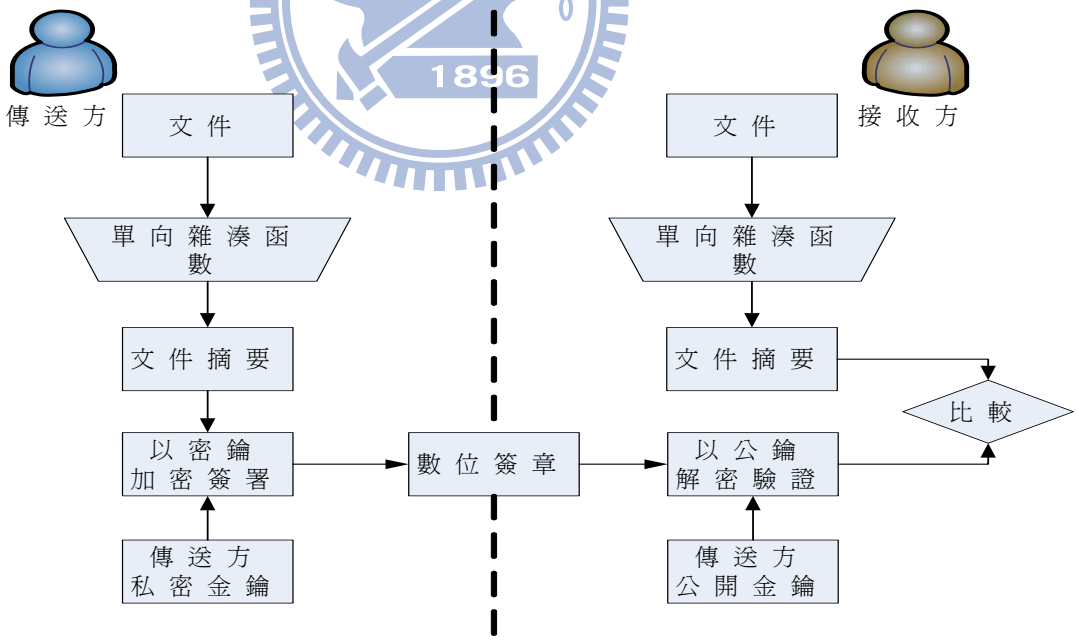

圖 11 數位簽章示意圖

由於非對稱加密系統先天上的限制，以及驗證性的考量下，對明文直接使用私密金鑰加密的效率實在很低，因此便導入了利用單向雜湊函數計算明文的雜湊值作為數位簽章(Digital signature)的機制(如圖11)，以此機制視同本人親筆簽

名的功用。其做法如下，在數位簽章的過程中，簽署者必須先利用單向雜湊函數將電子文件轉換成固定長度的雜湊值，稱之為“摘要”（Digest）或是 “指紋”(Fingerprint)。隨後再利用非對稱加密法的私鑰對該摘要進行加密動作，以產生該電子文件的數位簽章；同樣地，當驗證者想要確認該電子文件的簽署人身分時，驗證者需要先使用同一單向雜湊函數，將該電子文件用單向雜湊函數轉換成固定長度的雜湊值，再利用簽署者的公開金鑰取出數位簽章內的雜湊值，進行核對的驗證動作，則可確保簽署者的身分，此一數位簽章機制即是雜湊函數重要的應用之一。

2.5 智慧卡簡介

智慧卡又稱為晶片卡或IC卡，其規格是依照ISO7816系列的標準，外觀為85.6mm(長) x 53.98mm(寬)，厚度為0.8mm的長形卡片，因應不同的需求與功能要求而其卡片構造略有差異，如電話卡與金融卡則是常見的智慧卡應用[11]。這項設計是由德國Jurgen Dethloff和Helmut Grotrupp兩位學者在西元1968年所提出，此兩位學者提出將所謂的積體電路結合在卡片中的想法，並獲得了專利權。當時最早的應用乃是法國電信公司(Postal and Telecommunications Services)的電話卡產品，在此之後晶片卡的應用便日益常見的出現在大眾的生活週遭，如晶片金融卡、健保卡、手機SIM卡等亦是。

晶片卡如果依照內部結構來區分的話，可以分為記憶卡(Memory Card)及智慧卡(Smart Card)，記憶卡的成本比起智慧卡要低廉，它的記憶體容量約為256Bytes～64KB大小，因其未內含CPU，故只具有儲存資料的功能而不能作邏輯或運算等處理。不過基於安全性理由，大多的記憶卡還是具有安全邏輯機制，使用者必須輸入密碼才能夠存取卡片上的資訊，若是數次輸入錯誤後，安全機制便會將卡片資料鎖死。記憶卡的應用最常見是在儲存資料或門禁管制，另外如電話卡、儲值卡等安全性不高的應用也很常見。而智慧卡因為包含有CPU、記憶體及其他運算處理單位在內，所以卡片可以進行一些運算的處理。智慧卡上的CPU一般為8位元單晶片處理器，也已有16位元或是32位元的處理器的新一代智慧卡晶片，由於具有運算功能，因此在卡片做加解密運算下(如DES、3DES、AES、RSA)，可以提供更高的保護效果，便常應用於金融或電子商務的資料存放。

晶片卡若是依照讀取的方式來區分，主要可以分為接觸式卡(Contact Card)、非接觸式卡(Contactless Card)以及接觸式與非接觸式的混合卡(Combi-Card)三種。接觸式智慧卡是指卡片上的晶片必須接觸讀卡機的讀寫頭才能進行資料的存

取，所以在此三種類型中，是具有較高的安全性及正確性的一類，在晶片上通常有8個金屬接觸點，以此介面與讀卡機進行資料的存取與傳送。但是，接觸式智慧卡也不是沒有缺點，由於每次的資料傳輸都倚靠著晶片與讀卡機的讀寫頭接觸，這項動作過於頻繁所引發的磨損，也導致接觸式的卡片使用壽命相較於非接觸式的卡片使用壽命要來的短[11]。

因此在使用者認證系統中，也有不少使用智慧卡為媒介，進行使用者身分認證的應用存在。

第三章 相關研究

本章就近年來諸多認證方式技術研究中，摘錄與本研究相關的文章進行探討。於 3.1 節討論圖形化密碼的認證研究；3.2 節對智慧卡認證方式的研究進行討論；3.3 節為則對密碼安全性的影響因素作探討。

3.1 圖形化密碼認證之研究

自從 1993 年G.E.Blonder首先提出圖形化密碼方法之後，許多的學者便投入此類型認證方法的研究，相較於以往存在著許多缺點[12]的帳號密碼登入方式[12]，圖形化密碼認證方法降低了暴力攻擊與自動化攻擊的破解可能性。由於使用者大多趨向選擇容易記憶的密碼，而此種容易記憶的密碼也容易被猜測破解；若是要使用者選用不容易被破解的密碼，那麼這樣的密碼對於使用者而言卻是很難記憶起來的。為了改善這個問題，於是圖形化密碼便因應而生，解決了字母數字類型密碼對於使用者的不友善性[28]。圖形化密碼經過多年的發展後，學者Dhamija, R.和 Perrig, A.在 2000 年時提出了Déjà Vu認證 方法，將圖形化認證帶至另一個層次[2]，以下則簡略介紹此認證方法。

3.1.1 Déjà Vu認證法

由於人類對於圖像有優異的記憶能力[3][15][28]，Déjà Vu系統是以此為基礎而建構的一種認證系統。Déjà Vu系統的使用者，必須由系統圖庫中選取p個影像(如圖 12 所示)，建立一套自己的認證圖組(image portfolio)。當進行使用者認證時，系統會顯現出n張圖像的認證判斷圖組(challenge set)，該認證判斷圖組(challenge set)中包含了m張使用者所選認證圖組中的圖片。而使用者必須正確點選出屬於他自己認證圖組的圖片數次以通過系統認證，Déjà Vu系統包含以下三個階段:認證圖組設定階段,訓練階段和認證階段。

在使用者選定自己的認證圖組(image portfolio)後，便會進入訓練階段，透過訓練階段的練習，讓使用者更加熟悉系統認證的模式以及加強記憶自己所選的認證圖組(image portfolio)圖片。訓練階段被要求在安全的環境下進行，以防止第三者得知使用者的圖組圖形，導致認證的安全性降低。

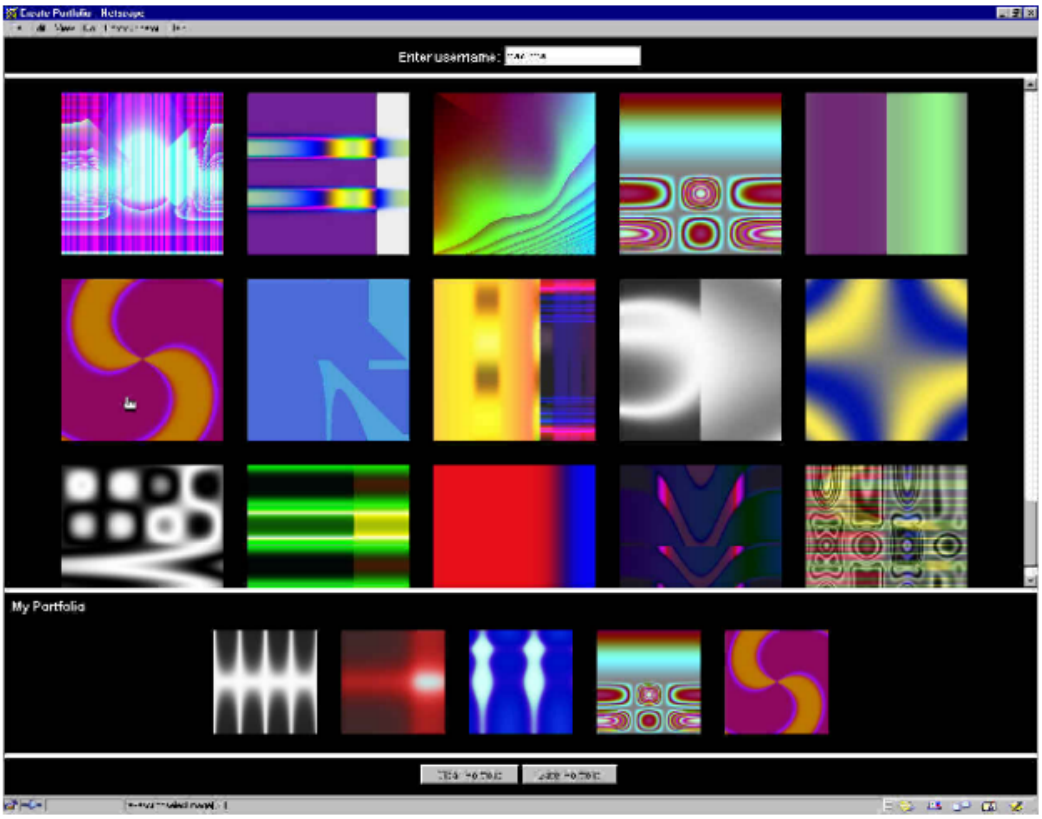

圖 12 Déjà Vu 認證圖組設定畫面

資料來源："Déjà Vu: A User Study Using Images for Authentication"[2]

但是此系統的缺點在於使用者的圖形組檔案的產生，是以明碼儲存在伺服器中，因此伺服器必須確保絕對的安全與可信任性，也造成了安全上的疑慮。另外，雖然研究結果顯示[29]，圖形化密碼的準確性較字母數字密碼高出許多，可是在平均的登入時間卻也要花較長的時間，再加上訓練時間的因素考量，將會降低使用者使用此系統的意願。

因此在 2005 年學者Wiedenbeck等五人提出了PassPoints系統，改進了Blonder's方法[12]一些缺點而成的，期望讓圖形化密碼認證方式，能有更好的使用效果。

3.1.2 PassPoints認證法

PassPoints系統[7]是Wiedenbeck等學者在2005年所提出，該研究進行了六周的使用者實驗，發現使用者在使用此系統，相較於數字字母密碼認證方式，有較少的錯誤產生，而且密碼的記憶性也比較好。對於使用者更方便的是PassPoints認證系統允許使用者自己選取認證時用的圖形圖片檔案。

PassPoints系統除了讓使用者可選擇想要使用的圖片之外，還將必須儲存的密碼以雜湊函式加密的方式儲存，讓駭客難以從結果反運算得到原本的密碼，在使用者進行認證的時候，也透過雜湊函式的的結果來比對使用者所輸入的圖形密碼正確性。由於圖形化密碼是透過在圖片上點選數個位置來作為密碼，但是使用者也非常難在圖片點選的時候，非常準確的點在特定的幾個點上，因此通常是允許使用者點選在一個區域以內，便視為正確的點選。正由於這樣的不準確性，根據雜湊函釋的擴張性(Diffusion)，要得到相同的雜湊函式輸出結果，是有些困難的。但透過將圖片區分成適當大小的方格區塊後，並且做一些適度的系統調整，便有效達成雜湊函式應用的結果，這在安全性上是有很大的助益。

而究竟圖形化密碼的有效密碼空間[1]可以有多少呢？在數字字母型的密碼上，一般只使用64個字元[2]讓使用者選用，這樣的密碼變化性如果應用在八個字元長度的密碼上，會有$64^8 = 2.8 \times 10^{14}$的密碼可能性出現；而在圖形化密碼，如果使用了451 x 331像素大小的圖形檔案，並將其以20 x 20像素小區塊作區分，大約可以將這個圖檔分成451 x 331 / 20 x 20 = 373個小區塊。若以五個選擇點作為密碼的話，可以得到$373^5 = 7.2 \times 10^{12}$大小的密碼空間可能性。要是使用了1024 x 752像素大小的圖形檔案，並相同的小區塊區分及五次點選密碼，可以得到2.6×10^{16}種密碼可能性，如表2便呈現PassPoints系統五次點選圖形密碼和八字元長文字密碼的密碼空間比較，由此表所比較的結果，可以知道圖形化密碼系統，相較於文字密碼而言是有較大的密碼空間，而且使用者僅僅使用滑鼠做點選的動作，就可以達到這樣的效果[2][7]。

表 2 密碼空間比較表

Comparison of password space for alphanumeric passwords and PassPoints with different parameters

	Image size	Grid square size (pixels)	Alphabet size/ No. squares	Length/No. click points	Password space size
Alphanumeric	N/A	N/A	64	8	2.8×10^{14}
Alphanumeric	N/A	N/A	72	8	7.2×10^{14}
Alphanumeric	N/A	N/A	96	8	7.2×10^{15}
Graphical	451 × 331	20 × 20	373	5	7.2×10^{12}
Graphical	1024 × 752	20 × 20	1925	5	2.6×10^{16}
Graphical	1024 × 752	14 × 14	3928	5	9.3×10^{17}
Graphical (1/2 screen used)	1024 × 752	14 × 14	1964	5	2.9×10^{16}

資料來源：PassPoints: Design and Longitudinal Evaluation of A Graphical Password System[7]

[1] 密碼空間是指在給定密碼可用字集集合與規則時，所有可能產生密碼的集合。

[2] 通常是 0~9 等 10 個數字，大小寫英文字母共 52 個、_(底線)及 .(點)等，共 64 個。

當然，密碼空間越大對於安全性的提升效果越好，這是顯而易見的。但是使用者的使用性與記憶能力，也是不容忽視的重要因素。於是在2008年學者郭信東[16]提出了使用資訊隱藏學的技術，來改進PassPoints的圖形密碼系統的圖片檔案儲存資料量的問題，和兩個輔助使用者記憶圖形化密碼點選的技巧。

不過，這樣的圖形化密碼方法雖然有效提升的密碼空間大小，也對使用者在使用上的便利性有很好的提升，可是對於視力較差、色盲或動作控制不良的使用者而言，圖形化密碼卻是不夠友善的；另外，肩窺(Shoulder Surfing)的安全性問題還是存在，因此本研究也試圖改善這些缺陷，期許達到更好的使用者友善性和安全性效果。

3.2 智慧卡身分認證之研究

通常智慧卡的使用是將卡片授予認證人使用，由於只要持有卡片，就如同取得了該項服務的使用權，因此為了避免卡片遭到非法的使用者竊取利用，便透過了個人驗證碼(Personal Identification Number，PIN)系統協定的方式，來簡易的確認該使用者的第一階段的合法性，若智慧卡在連續數次的個人驗證碼錯誤後，智慧卡便利用自動鎖住卡片的方式，來停止該卡片的使用性。也由於智慧卡本身晶片具有運算功能，因此可以進行演算法的運算，來提高使用的安全性，例如在晶片進行資料存入時，會先進性加密運算，再將資料儲存。由於智慧卡的安全性效能的優勢，因此有不少的認證方式，是建構在智慧卡的基礎之上，以下介紹與本研究相關的方法。

在西元2005年學者Lee等三人[17]提出了一個以智慧卡搭配隨機亂數(nonce)的無認證表驗證機制，這個驗證機制在運算上僅使用了XOR運算和單向雜湊函數，便達成互相驗證的效果，因此對於現今在分散式的網路系統環境中，進行使用者身分認證是很好的一個方法，可是在安全驗證機制上，有離線字典攻擊的安全性缺失存在，於是學者蔡佳倫(2007)[18]所提出的驗證機制[18]，改進了過去以單向雜湊函數為基礎認證研究，提出了一個多伺服器無驗證表的驗證機制，利用驗證中心(Registreation Center)提供伺服器與使用者進行互相驗證，讓使用者只需進行一次的註冊動作，就可使用伺服器提供的服務，而且不管是伺服器或註冊中心端都不需要存放驗證表，以下將介紹此驗證機制的細節。

學者蔡佳倫所提出的驗證機制，要求使用者在伺服器上進行註冊期間的所有動作，且利用安全通道傳送使用者所使用的帳號和密碼，並由伺服器端的一個私密變數x，來避免使用者必須記憶記十分長串的帳號ID，也能達到無認證表驗證系統的方式，以下是該法所能達到的安全性效果詳細介紹。

學者蔡佳倫(2007)所提出的驗證機制[18]，可以有效抵擋驗證表被竊攻擊(Stolen-Vertifier Attack)、重送攻擊(Replay Attack)、伺服器偽裝攻擊(Server Spoofing Attack)、使用者假冒攻擊(Impersonation Attack)等多項惡意攻擊方式，更可以提供安全的階段鑰匙(Session Key)，利用伺服器與使用者端所產生的隨機亂數，在每次雙方的通訊階段中使用，而當使用者重新進入系統時，整個驗證機制再驗證成功之後會另外產生一把新的階段鑰匙(Session Key)來加密通訊過程中的所有資訊，因此攻擊者無法利用過去所竊取或破解而得的階段鑰匙來解密通訊過程中的資訊，再加上每一次產生的階段鑰匙無任何關聯性，於是攻擊者也難以利用猜測的方法猜出階段鑰匙的值。

因此本研究利用學者蔡佳倫[18][18]所提出的驗證機制為基礎加以修改，應用在SSL認證方法中，提升本研究認證方法的安全性效能。

3.3 密碼安全性之研究

自從西元1990年開始，密碼被駭客破解的事件便層出不窮，至今已過了約三十年，但現代的使用者卻還是和過去的使用者相同，在選擇密碼這件事情上沒什麼太大的差異。在西元2009年十二月[19]，一名駭客竊取了知名社交網站上3200萬筆的使用者帳號、密碼資料，並將其公佈在網路上。讓網路安全專家有了一次新的研究機會，結果卻出人意料的，當今使用者所選擇密碼的特性與西元1990年起的許多研究結果大同小異！近年來許多網站因安全性考量，紛紛要求使用者將五碼密碼更新成六碼，以求得到更高的安全性，但這個善意的密碼更新政策，得到的結果卻是，最多使用者選擇的密碼只是從12345改成了123456[19]，使用者的確是將密碼改成了六碼，不過安全性的效果卻是非常的低。在這3200萬筆資料的研究結果中，發現容易被破解的密碼便佔了兩成，還有近五成的密碼只要透過暴力攻擊(Brute Force Attack)便可以簡單的破解，因為這些使用者選擇了名字、常見字、字典中的單字、連續數字或鍵盤相鄰按鍵等簡單模式的密碼。雖然多數的網站都會建議使用者，選擇足夠長度且英數混合的密碼，不過人們很顯然地還是偏好簡單一些的方式。

在這將近三十年的時間裡，使用者選擇密碼的方式並沒有太大的變化，但是電腦的計算能力卻已突飛猛進。如果把這個研究的結果轉換成實際的數據，駭客只要利用挑選過的字詞為基礎，搭配上自動破解程式進行暴力攻擊法，便可以在執行110次的錯誤以內，破解一個帳號；或是在17分鐘內破解並取得近千個帳號的權限。這是多麼驚人的數據，因此許多學者開始驚覺，究竟是哪裡出了問題，想要找出原因並對症下藥，以解決這個可怕的問題。

表 3 使用者的密碼分佈前二十名

1	123456	11	Nicole
2	12345	12	Daniel
3	123456789	13	Babygirl
4	password	14	Monkey
5	iloveyou	15	Jessica
6	princess	16	Lovely
7	rockyou	17	Michael
8	1234567	18	Ashley
9	12345678	19	654321
10	abc123	20	Qwerty

資料來源：Analysis of 32 million breached passwords [19]

一個好的密碼或稱為做強密碼[3]，是指此密碼被有心人士破解的難度而言，由於密碼是建立在使用系統的規則上，而且還要視電腦運算速度的差異而不同，因此密碼的強或弱，乃是相對並非絕對，況且強度再高的密碼也可能被破解、被竊取或甚至是洩漏。也有研究[20]指出要一個人強記住無意義的字串，不僅耗時且難以達成。但如果是對於使用者有意義的字串的話，就比較容易達成記憶的效果。一個密碼看起來簡單而且對使用者有意義的話，使用者會很快且準確的記住它，但是這樣的密碼卻難以抵擋駭客的破解攻擊。可是，當系統要求使用者使用複雜且亂度高的強密碼時，使用者卻將很難記住這樣類型的密碼，因此使用者將會把它寫下來，以幫助記憶。這樣將密碼寫下的一個動作，雖然讓使用者方便記憶，但卻也造成密碼遺失的可能性並產生認證安全性的疑慮。

儘管系統安全人員及專家們對使用者提出了許多選擇密碼的建議，像是使用者的密碼最好至少九碼以上，也應該包含一些特殊字元，而且最好不要只有在開頭或結尾使用特殊字元，最重要的是不要使用與個人相關的資訊來當作密碼，例如生日或電話號碼等。但還是有大約20%的使用者用了5000個最為常見的密碼[19]，將這些建議置之不理，而有更多的使用者，會將所有的網路服務登入密碼都設定成一模一樣[21][22][23]。由於人類天生的限制，要普通人記住又長又沒有意義的字串，總要花上許多的時間而且又容易出錯[20][24]，可是這樣複雜的長字串，對於密碼認證來說卻是有著很好安全性效果的強密碼。另外，由於過去都普遍認為，系統安全性是設計與技術層面的問題，但隨著系統服務提供者竭盡

[3]強密碼(strong password)：係指該密碼字串與使用者資訊沒有直接關聯性，且對於攻擊者而言難以破解的密碼。

一切的努力來提高安全性時，安全性的威脅和問題卻仍舊持續增加(DeAlvare, 1990; Gor don, 1995; Hitchings, 1995)[25]。在許多研究投入之後，發現僅僅使用者因素對於系統安全性的影響，超出我們的想像，Hitchings (1995)[25] 認為，使用者或系統使用性這兩項因素對於系統安全機制的影響，其實是不容小覷的。Davis & Price (1987)[25]更指出，人性因素應該在系統開始設計時，就納入考量之中，因為安全性機制都是由人去設計、實現和操作，甚至破壞也是。Adams 等人的研究結果[25]也指出，如果使用者造成某些安全性的問題，通常的主因是由於安全性系統設計上的問題所導致。這種缺陷是由於系統在設計的過程中，並沒有考慮到人(使用者)這個重要關鍵因素。

系統安全性的問題，在過去通常都被認為是技術與設計上的問題，而少有系統安全性和研究。可是在層出不窮的安全性問題狀況下，學者們開始從其他觀點進行研究，許多研究著眼在系統的建置流程上，發現未將使用者因素考量進去的話，將會對系統造成莫大的安全性威脅。因此，縱使再完美的技術機制，如果忽略了安全性機制對於使用者實用性的考量，也可能在實用上鑄成註定失敗的命運。

由於人的因素確實對於系統的安全性，造成了不少危害的可能。若服務提供者在認證系統上，未能有效的讓使用者願意改變，那只能試著迎合使用者的胃口，讓使用者自然而然的使用高安全性的密碼，來降低各種被攻擊的可能性。

然而，密碼的安全性極度重要，是眾所皆知。但是在此一要點上，也出現了和先前資訊隱藏學同樣的不完美，便是在安全性和使用便利性上，由於人的因素考量，必須在兩者中取得一個平衡點，方能達到「人」追求方便的訴求。

隨著科技的日益進步，密碼強度的需求也日漸提高，以因應電腦運算速度的提升和價格日益低廉的兩大威脅[19][26]，但在使用者因素的考量之下。目前看來最好的方法，便是結合各種不同有效且安全的認證方法，來達到更好的安全性效果，以抵擋有心人士的破解與攻擊，也符合使用者的需求。於是本研究提出的方法，將結合圖形化密碼和資訊隱藏學的優點，並注重人性因素的考量[24]，使用者無需記憶無意義或長字串的強密碼，僅以容易記憶的一張圖片檔案來達成有效的安全性登入機制。

第四章 SSL圖形驗證法

此章將會介紹本研究提出的方法，SSL(Simple Steganographic Login)圖形驗證法是利用資訊隱藏學的技術搭配一圖片檔案，以及使用單向雜湊函數進行運算，來完成使用者認證。使用者僅需在記住自己的帳號及密碼之外，記憶所使用的圖片檔案，即可達到高強度的安全性認證。以下 4.1 節至 4.5 節將會由註冊階段和登入階段等使用主軸，來進行系統架構說明。開始介紹本研究提出的驗證機制前，先對於所使用的符號進行定義:

表 4 符號說明表

符號	說明
U,u;S,s	代表使用者與伺服器
U_{ID} , U_{PW}	使用者的帳號、密碼
P , P_S	BMP 圖形檔案與嵌入金鑰的圖形檔案
U_A , S_A	以使用者或伺服器的資訊經單向雜湊函數產生的數位指紋
T_{key}	臨時識別碼
N	隨機所產生的亂數
X->Y:M	X 傳送 M 給 Y
R	亂數位移量
H( )	單向雜湊函數
⊕	XOR 運算
||	連結

4.1 SSL方法介紹

下圖(圖 14)是 SSL(Simple Steganographic Login)認證法的流程示意圖，使用者在登入系統時，仍是使用帳號、密碼進行第一階段的認證動作，當使用者通過帳號、密碼的確認，隨後系統會要求使用者提供圖形金鑰或是臨時識別碼，做

進一步的身分確認動作，待圖形金鑰或臨時識別碼核可無誤後，使用者才會成功的進入系統服務；反之，只要有一個階段無法通過認證，則不予登入。

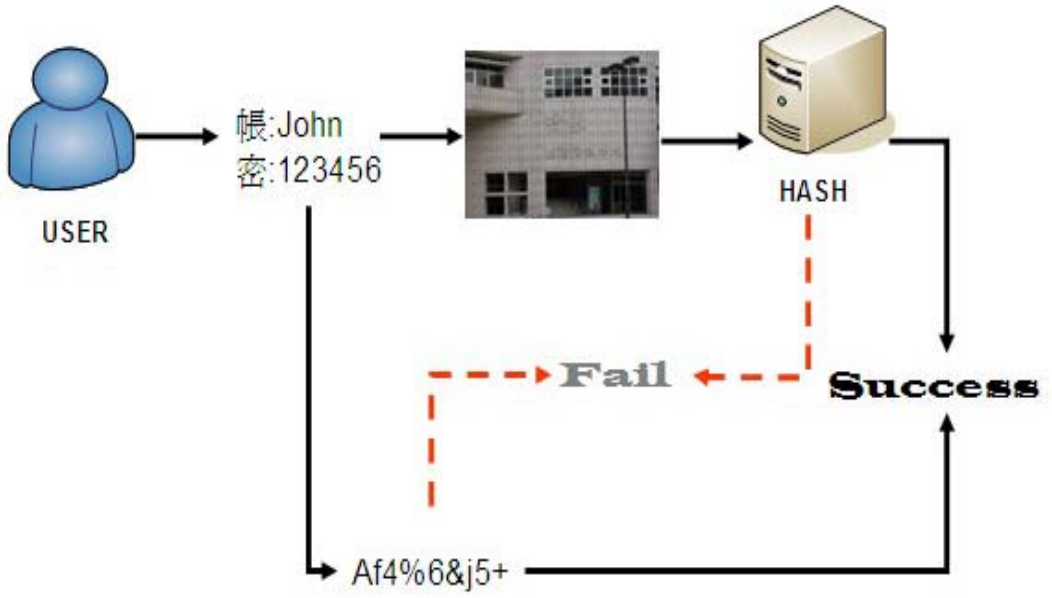

圖 13 認證流程示意圖

本研究所提出的 SSL (Simple Steganographic Login)認證是以現今最多系統採用的帳號、密碼認證方式為基礎，搭配一個 BMP 圖片檔案，使用資訊隱藏技術以及單向雜湊函數的結合，將高安全性的認證金鑰，隱藏在 BMP 圖形檔案中，來進行使用者認證的一個方法，期許達到使用者的便利性與系統的認證安全可靠性的一種認證方式。以下將會把 SSL(Simple Steganographic Login)認證方法分成註冊階段、登入階段、驗證階段、登出階段、密碼變更等階段做詳盡說明。

4.2 註冊階段

伺服器和使用者端透過安全通道連接後，才會進行註冊階段的各項動作，當使用者備妥註冊資訊送交伺服器後，伺服器便會將使用者導入嵌入圖形金鑰的階段。此時，使用者必須上傳一個 BMP 格式的圖片檔案，而檔案的長 x 寬像素大小，必須大於 600 x 400 的限制，以利於系統嵌入認證金鑰的動作。系統端在接收到使用者所上傳的圖形檔案後，會將所計算出的金鑰 P_{key}，嵌入該圖形檔案內，並回傳給使用者包含金鑰的 BMP 圖形檔案，要求使用者妥善保存該圖形檔案，整個註冊期的流程將由以下兩小節詳細敘述。

4.2.1 使用者帳號資料建立

透過安全通道傳輸使用者建立帳號的所有資訊，作為防止資料遭竊取的第一道防線，這個階段要求使用者提供的資料，可依照服務系統所需而有所不同，但當使用者完成資料建立後，系統會利用此階段使用者所輸入的個人資料，經由雜湊函數H( )產生嵌入金鑰的部份依據U_A。在這個階段中，使用者像一般的註冊模式一樣，需要牢記自己所挑選的帳號(U_{ID})與密碼(U_{PW})等註冊資訊，以進行日後的登入動作。

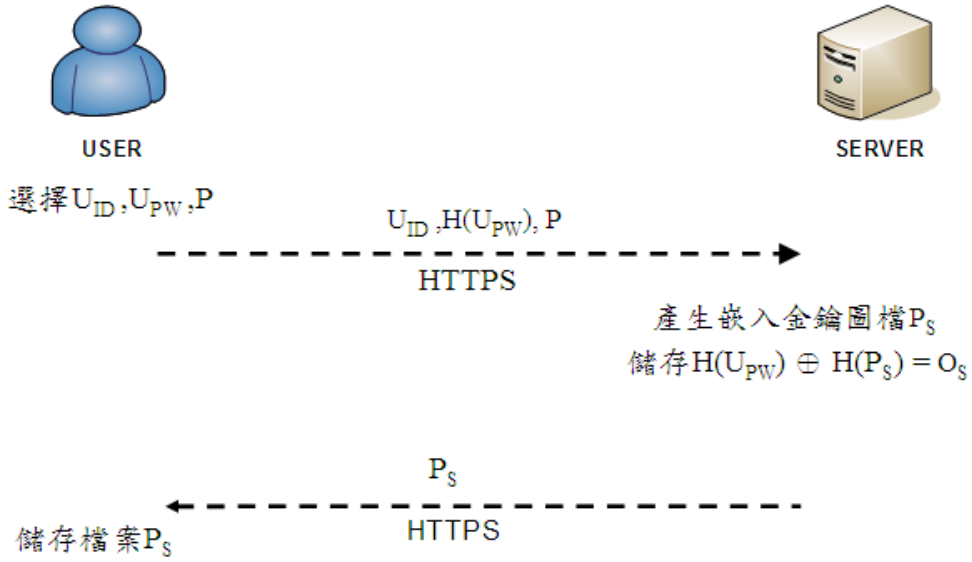

圖 14 SSL 註冊階段示意圖

Step 1： U->S: U_{ID}、$H(U_{PW})$、P

使用者傳送自己的帳號(U_{ID})、密碼(U_{PW})及圖形檔案 P 以安全通道的方式傳送到伺服器。

Step 2： $H(U_{PW}) \oplus H(P_S) = O_S$

伺服器端進行圖形金鑰嵌入動作並儲存 O_S。

Step 3： S->U: P_S

伺服器端回傳圖形金鑰檔案，要求使用者妥善保存。

4.2.2 圖形檔案嵌入金鑰

當伺服器端由使用者資訊藉雜湊函數產生U_A並紀錄產生依據之後，伺服器端也會依照伺服器資訊以雜湊函數產生S_A，作為產生嵌入金鑰的第二部份資訊，隨後伺服器便產生嵌入圖檔的金鑰$H(U_A||S_A)=P_{key}$，將P_{key}做位移R bits後嵌入使用者所上傳的圖形檔案P中，得到圖形金鑰P_S並記錄$O_S=H(U_{PW})\oplus H(P_S)$與R，作為日後認證階段使用，以上所述同圖15所示。

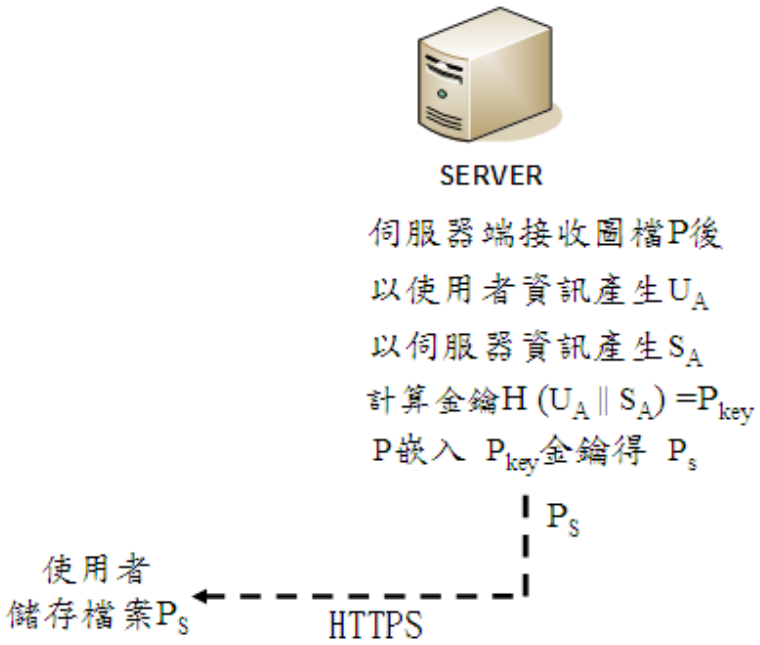

圖 15 嵌入金鑰示意圖

Step 1：

伺服器端產生 $H(\text{使用者資訊})=U_A$、$H(\text{伺服器資訊})=S_A$、$H(U_A||S_A)=P_{key}$。

Step 2：

伺服器端進行圖形金鑰嵌入動作產生 P_S 並回傳。

本研究中採用的圖形檔案格式為24-bit BMP格式，因24-bit BMP (BitCount=24)圖形檔案不具色表，所以讀取較容易，且檔案內容分為三部分，檔頭部分、影像資訊、資料部分。而圖形的大小要求使用者提供至少需600像素 x 400像素，約700KB大小以上的BMP圖形檔案，以供SSL系統嵌入金鑰。

4.3 登入階段

當使用者完成一切的註冊動作，並將嵌入金鑰後的圖形檔案下載且儲存後，便可以開始執行登入的動作。在SSL(Simple Steganographic Login)的登入階段中，分成使用者正常利用嵌入金鑰的圖形檔案執行登入動作，和使用者未攜帶圖形檔案時的登入行為兩種模式，以下則以「正常登入階段」與「不正常登入階段」分別稱之。

以下4.3.1節會對「正常登入階段」做詳細的流程說明。4.3.2節則會介紹「不正常登入階段」的各個步驟。

4.3.1 正常登入階段

本研究所稱的正常登入階段，係指使用者利用嵌入金鑰的圖形檔案進行登入的行為，當使用者向伺服器端提出登入請求後，伺服器端隨即會與使用者端建立起安全通道，然後使用者便可以輸入自己的帳號與密碼，以進行第一階段的身分認證動作與伺服器端身分確認，當系統端確認該使用者的帳號與密碼正確後，便要求使用者提供圖形金鑰檔案，讓系統端可以進行圖形金鑰確認的動作，來核可使用者的身分，最後系統端將會回覆使用者，是否登入成功或失敗，之後此階段便宣告結束。

以下是本階段的詳細流程說明：

當系統端與使用者端建立安全通道後，進行以下動作。

Step 1： U->S: U_{ID}、H(U_{PW})

使用者傳送自己的帳號(U_{ID})、密碼(U_{PW})到伺服器，進行第一階段使用者身分認證。

Step 2： S->U: P_S？

伺服器端確認使用者身分，並要求使用者提供圖形金鑰 P_S。

Step 3： U->S: P_{key}

伺服器端以本地端資訊計算 P_{key}’ 與使用者端提供之 P_{key} 進行比對。

Step 4： S->U:

伺服器回覆認證結果。

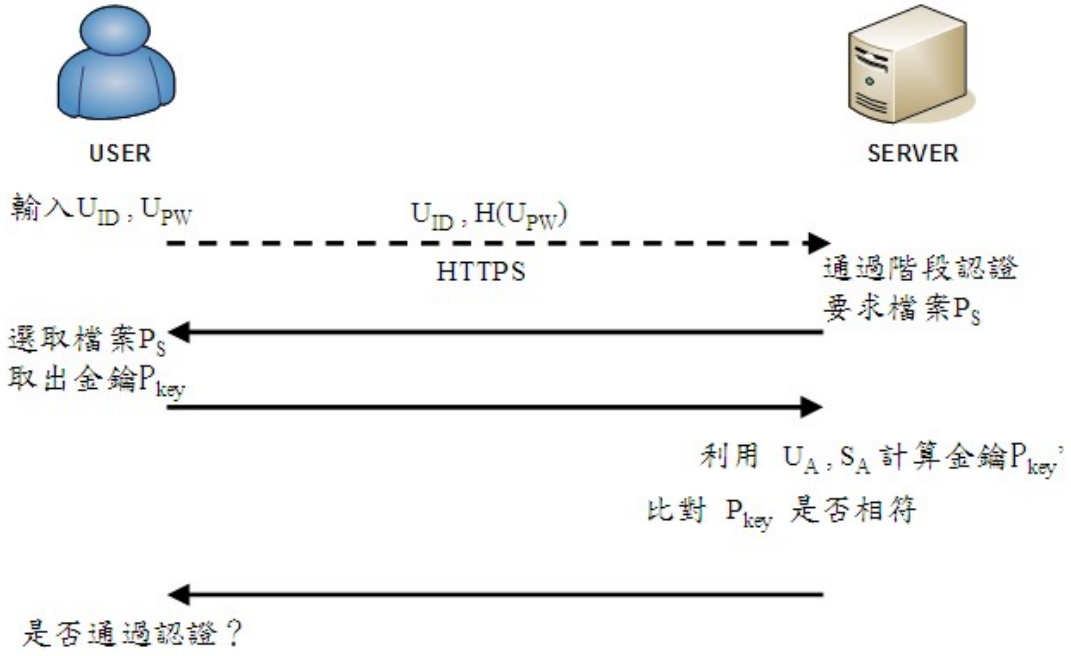

圖 16 SSL 正常登入期示意圖

4.3.2 不正常登入階段

本研究所指的不正常登入階段，係指使用者因為沒有攜帶圖形金鑰檔案，卻希望進行登入的動作過程。當使用者向伺服器端傳送不使用圖形金鑰的登入請求後，伺服器端仍會與使用者端建立安全通道，要求使用者端輸入自己的帳號與密碼，當系統端確認該使用者的帳號與密碼正確後，便利用使用者於註冊階段所指定的電子郵件信箱位址，傳送包含臨時識別碼的信件給使用者，並要求在系統指定的時間內使用，使用者則必須進入自己的電子郵件信箱，取得該臨時識別碼，以進行後續的身分認證確認動作，若超過系統核定的有效時間，則該臨時識別碼失效。系統端則依據識別碼的使用時效和使用者所輸入的識別碼，來判斷使用者的身分正確性，以達到不使用圖形金鑰檔案進行登入的動作。同時伺服器可以將此次無圖形金鑰登入之動作進行紀錄，並通知系統管理員及該使用者，以確保無非法登入動作產生；或是要求該使用者於次一回的登入動作時，需更新自己的圖形金鑰檔案。

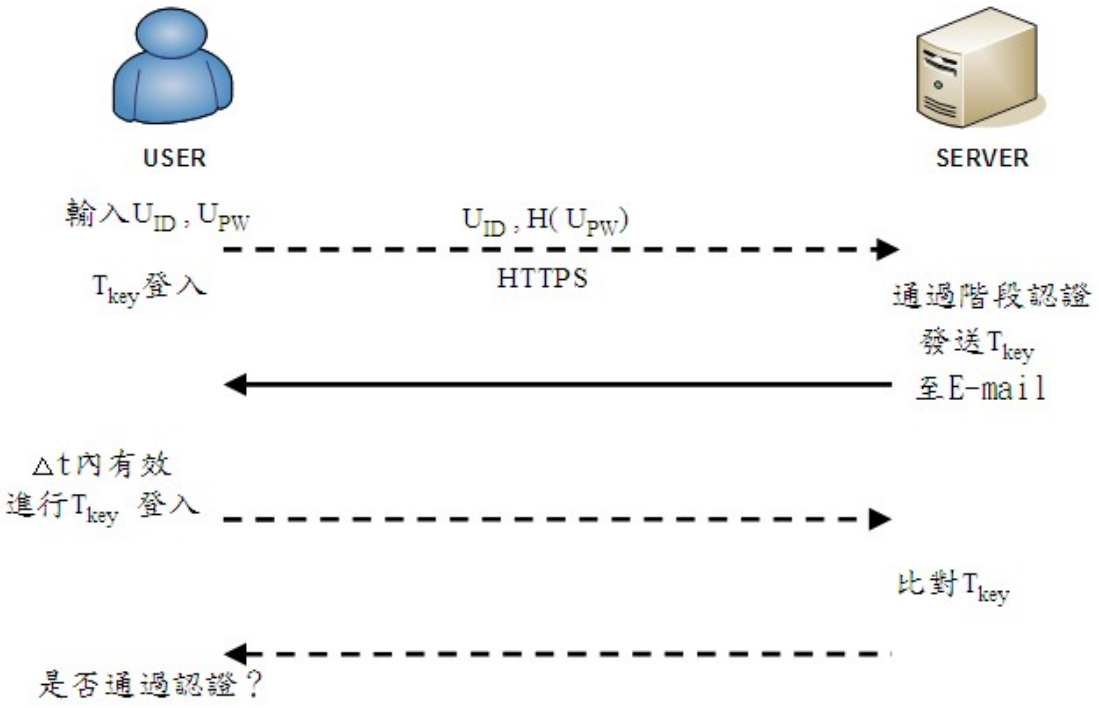

圖 17 SSL 不正常登入期示意圖

Step 1： U->S: U_{ID}、$H(U_{PW})$

使用者將自己的帳號(U_{ID})、密碼(U_{PW})以安全通道的方式傳送到伺服器，並要求以臨時識別碼方式登入。

Step 2：

伺服器端比對帳號、密碼是否正確。

Step 3： S->U(E-Mail): T_{key}

通過帳號、密碼階段認證後，伺服器端傳送臨時識別碼 T_{key}，至使用者註冊時使用之 E-Mail 信箱。

Step 4： U->S:輸入 T_{key}

使用者取得 E-Mail 信箱之臨時識別碼認證信，於伺服器端的限定時間內回傳 T_{key} 給伺服器端進行身分認證。

4.4 驗證階段

本研究的認證機制是利用 XOR 運算和雜湊函數應用所達成，當使用者傳送帳號(ID_u)給伺服器之後，伺服器會確認該使用者帳號是否有效，然後伺服器端與使

用者端會各自產生隨機亂數 N_s 與隨機亂數 N_u，然後透過 XOR 運算和雜湊函數的保護下互相傳遞、交換，以避免被攻擊者利用竊聽的方式去得知這兩個隨機亂數，在交換完這兩個隨機亂數 N_s 與 N_u 之後，兩端將會分別進行驗證對方在分解這兩個隨機亂數上是否正確，作為驗證使用者端與伺服器端的身分是否合法的依據。在驗證確認使用者以及伺服器的身分之後，通訊的雙方會產生一把階段鑰匙(session key)，用以加解密驗證時期以後所交換的資訊，保護資訊在網路中傳遞的安全性，以下則是整個驗證時期的詳細步驟。

如圖 18 所示：

Step 1： S: $O_s \oplus N_s = K1$

伺服器端利用所儲存的 O_s，搭配自己產生的隨機亂數 N_s，以 XOR 運算產生 K1。

Step 2： S->U: K1

伺服器端將 K1 傳送給使用者端。

Step 3： U: $O_u = H(U_{PW}) \oplus H(P_s)$

使用者端以自身所擁有資訊計算出 O_u。

Step 4： U: $O_u \oplus K1 = N_s$

使用者端用接收到的 K1 進行 XOR 運算得出伺服器端亂數 N_s。

Step 5： U: $O_u \oplus N_u = K2$

使用者產生一個隨機亂數 N_u，並以 N_u 與 O_u 計算出 K2。

Step 6： U->S: K2

使用者產傳送 K2 給伺服器。

Step 7： S: $K2 \oplus O_u = N_u$

伺服器接收 K2 並以此計算出隨機亂數 N_u。

Step 8： S: $H(O_s \,||\, N_s) = T1$

伺服器以雜湊函數計算出 T1。

Step 9： S: $K3 = T1 \oplus N_u$

伺服器計算 K3。

Step 10： S->U: K3

伺服器將 K3 傳遞給使用者。

Step 11： U:

當使用者收到 K3，先使用 N_s 與 O_u 計算出 T2。

Step 12： U: $T2 \oplus N_u = K3'$

使用者在利用計算出的 T2，將其與 N_u 進行 XOR 運算，得到 K3’ 。

Step 13： U: K3 =? K3’

使用者比對 K3 與 K3’ ，如果兩者相符，則伺服器端身分正確，則會繼續

進行圖 19 的驗證步驟；如果錯誤，便可以知道該伺服器是假冒的，而中斷之後的驗證步驟。

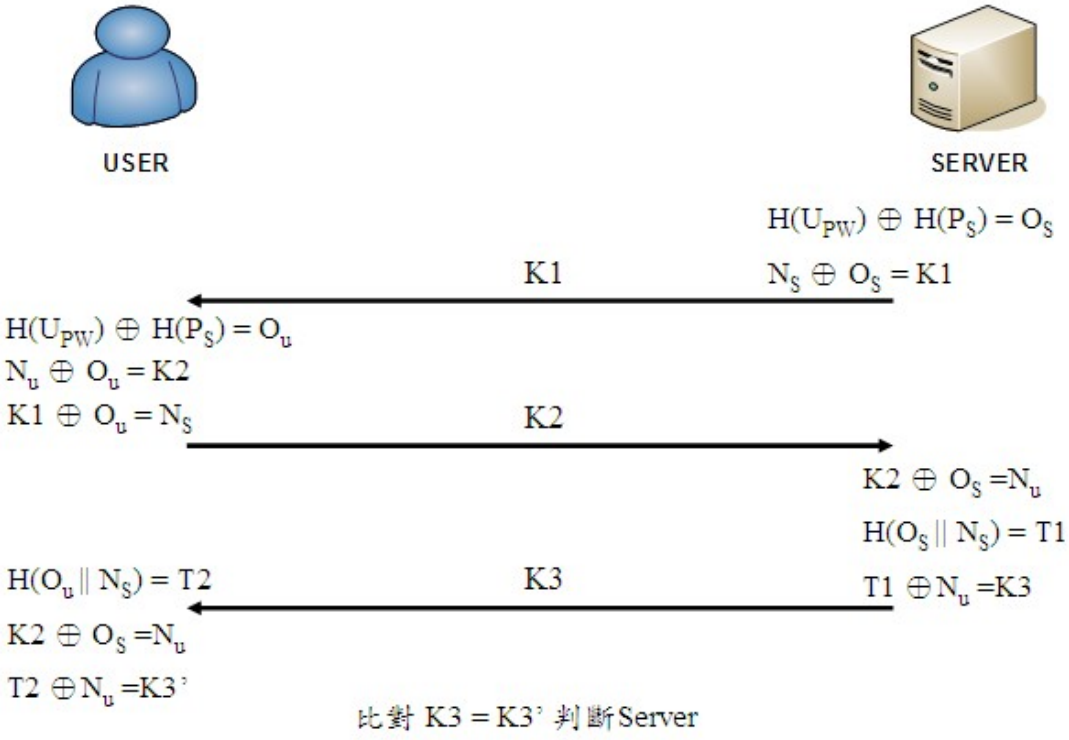

圖 18 驗證伺服器身分流程圖

Step 16： U: H（N_S || P_{key} || N_u）=K4

伺服器身分正確無誤後，使用者將會取出隱藏在 P_s 圖形檔案內的金鑰 P_{key}，搭配先前所傳遞和產生的亂數 N_S 和 N_u，以雜湊函數運算產生 K4。

Step 15： U->S: K4

使用者產傳送 K4 給伺服器。

Step 16： S: H（U_A || S_A）=P_{key}

伺服器會利用本機的資訊，計算出該使用者的金鑰 P_{key}。

Step 17： S: H（N_S || P_{key} || N_u）=K4’

伺服器身分正確無誤後，以上一步驟所得 P_{key}，搭配先前所傳遞和產生的亂數 N_u 和 N_S，以雜湊函數運算產生 K4’ 。

Step 18： K4 =? K4’

伺服器端比對 K4 是否等於 K4’ ， 如果兩者相等，則使用者端身分正確，予以通過認證進入系統服務；如果 K4 不等於 K4’ ，則判定該使用者無法通過身分認證，不予登入系統。

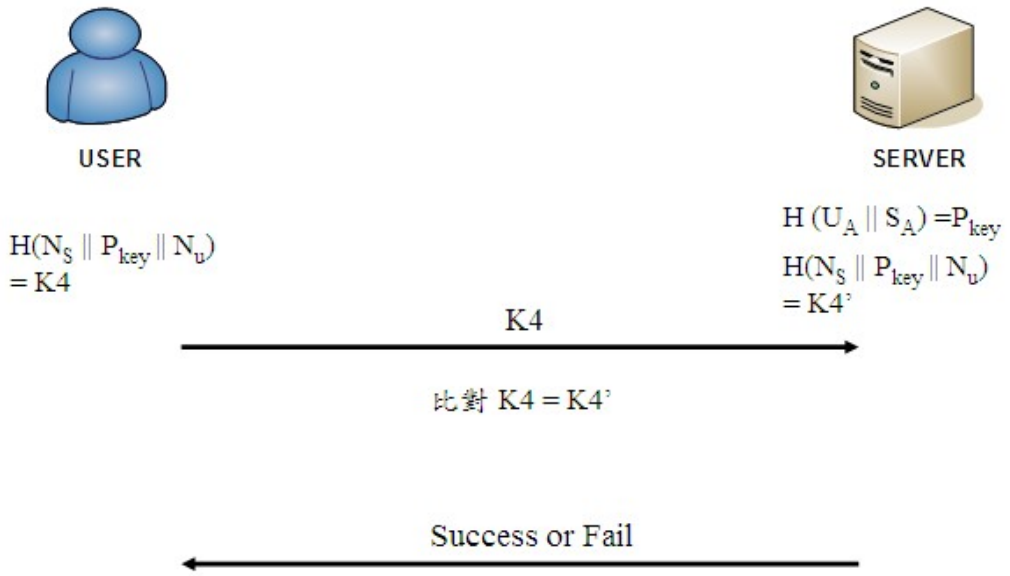

圖 19 驗證使用者身分流程圖

4.5 密碼變更階段

本研究的認證系統在密碼變更階段，與一般常見的認證系統沒有太大的差異，唯一不同的是多出一個圖形金鑰變更的設定。因此，此小節只針對圖形金鑰變更行為，做詳細的說明。

本階段可由系統所訂的更新金鑰規則或是由使用者自行提出而進行，本階段的行為將在使用者正確登入後方可進行，並且於進行時將以安全通道模式或是利用對稱金鑰加密等方法，以確保圖形金鑰的安全性，以下為更新金鑰的詳細流程說明。

Step 1： U->S: P'

使用者端傳送新的圖形檔案 P' 給伺服器端。

Step 2： S: H(U_A' || S_A')=P_{key}'

伺服器端產生新的 H(使用者資訊)=U_A' 、H(伺服器資訊)=S_A' ，以計算出新的認證金鑰 P_{key}' ，並儲存更新金鑰產生方法所需資訊。

Step 3： S: P_S'

伺服器端嵌入金鑰 P_{key}' 至圖檔 P' 得到新的圖形金鑰檔案 P_S' 。

Step 4： S->U: P_S'

伺服器端傳送 P_S' 給使用者。

Step 5： U: P_S'

使用者妥善保存新圖形金鑰檔案 P_S' 。

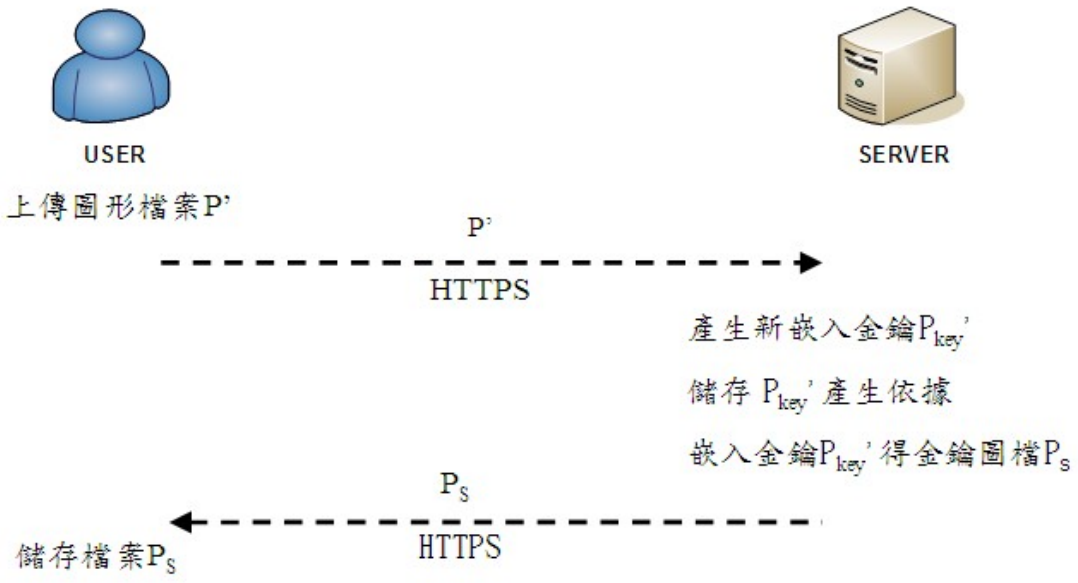

圖 20 更新圖形金鑰流程圖

4.6 登出階段

在此會提出登出階段進行討論，是基於考量安全性因素之緣故。一般來說當使用者向伺服器端提出登出請求時，通常所執行的動作是，伺服器端紀錄該使用者的使用資訊，然後中斷該使用者的服務使用權限，登出階段便告結束；若是為了提高圖形密碼的安全性，則可在使用者提出登出動作請求時，伺服器端以該使用者登出的時間或其他系統所選資訊，產生新的圖形認證金鑰 P_{key}，進行金鑰嵌入圖形檔案的更新動作，以提高次一回認證的安全性效果。這個階段，可以是系統服務者的要求，或是開放給使用者自行選擇的方式來進行。相對的，進行此一更新金鑰的動作，會造成使用者在使用上的困擾與伺服器端的負載，故對於本系統的提高使用者易用性主要訴求有所衝突，因此在實作上並未加入，但特別提出給不同需求之系統服務者作為提升安全性考量的參考。

4.7 偽裝金鑰產生階段

由於金鑰圖形檔案的大小不大，且為了方便使用者產生多個偽裝欺騙的圖形檔案，因此提供此一系統功能，讓使用者可以很迅速方便的取得包含假金鑰資訊

的多個圖形檔案，而且也有助於系統獲得更好的使用者端安全性保證。此階段的執行方法，其實非常簡易，系統端在接收到使用者提出偽裝金鑰產生的請求後，要求使用者上傳提供原始圖片檔案，系統在接收圖形檔案後，便進行假資訊隱藏進入圖片的動作，並產生亂數檔案名稱，本研究中預設的檔案數量為十個偽裝金鑰圖形檔案，產生完畢後壓縮成 ZIP 格式的壓縮檔案，供使用者下載，當使用者下載完成後，便可將此 ZIP 檔案解壓縮，即獲得十個嵌入假金鑰資訊的圖形檔案，再透過使用者自行記憶或變更檔案名稱的方式，即可分辨何者為真實的圖形金鑰檔案，哪一些檔案為偽裝圖形金鑰檔案，更加降低肩虧攻擊或檔案比對攻擊的可能性。

4.8 SESL方法

由於許多的因素，唯有不斷提高安全性效果，才能保有系統端和使用者端的資訊安全。於是此處將原有的 SSL 認證方法，加入的對稱式加密函數的應用，讓驗證的安全性更加提高，以下將對 SESL(Symmetric Encryption Steganographic Login)方法加以說明。

SESL(Symmetric Encryption Steganographic Login)方法，主要是將 SSL 方法的驗證階段，加入對稱式加密系統的保護，以抵禦攻擊者的惡意破解。而在其他階段部份，均與原有的 SSL 認證方法相同。而此處會採用對稱式加密系統方法，卻不使用非對稱式加密的原因，是由於對稱式加密方法在執行效率上比較好，故未使用非對稱式加密方法。

以下則對 SESL 利用對稱式 AES 加密方法的驗證階段進行詳細說明。

當伺服器端確認使用者的帳號 U_{ID} 與 $H(U_{PW})$ 正確後，將會進行下列驗證動作。

Step 1： S: $O_S \oplus N_S = K1$

伺服器端利用所儲存的 O_S，搭配自己產生的隨機亂數 N_S，以 XOR 運算產生 K1。

Step 2： S->U: K1

伺服器端將 K1 傳送給使用者端。

Step 3： U: $O_u = H(U_{PW}) \oplus H(P_S)$

使用者端以自身所擁有資訊計算出 O_u。

Step 4： U: $O_u \oplus K1 = N_S$

使用者端用接收到的 K1 進行 XOR 運算得出伺服器端亂數 N_S。

Step 5： U: $O_u \oplus N_u = K2$

使用者產生一個隨機亂數 N_u，並以 N_u 與 O_u 計算出 K2。

Step 6： U->S: K2

使用者產傳送 K2 給伺服器。

Step 7： S: K2 ⊕ O_u = N_u

伺服器接收 K2 並以此計算出隨機亂數 N_u。

Step 8： S: H(O_s || N_s) = T1

伺服器以雜湊函數計算出 T1。

Step 9： S: K3 = T1 ⊕ N_u

伺服器計算 K3。

Step 10： S->U: K3

伺服器將 K3 傳遞給使用者。

Step 11： U: H(O_u || N_s) = T2

當使用者收到 K3，先使用 N_s 與 O_u 計算出 T2。

Step 12： U: T2 ⊕N_u = K3’

使用者在利用計算出的 T2，將其與 N_u 進行 XOR 運算，得到 K3’ 。

Step 13： U: K3 =? K3’

使用者比對 K3 與 K3’ ，如果兩者相符，則伺服器端身分正確，則會繼續進行圖 4-8 的驗證步驟；如果錯誤，便可以知道該伺服器是假冒的，而中斷之後的驗證步驟。

Step 14： U:[H(N_s || P_{key} || N_u)]$_{Nu}$ =K

伺服器身分正確無誤後，使用者將會取出隱藏在 P_s 圖形檔案內的金鑰 P_{key}，搭配先前所傳遞和產生的亂數 N_s 和 N_u，以對稱式 AES 加密方法產生 K。

Step 15： U->S: K

使用者傳送 K 給伺服器。

Step 16： S: [H(N_s || P_{key} || N_u)]$_{Nu}$ =K’

伺服器身分正確無誤後，以本地端計算所得 P_{key}，搭配先前所傳遞和產生的亂數 N_u 和 N_s，以對稱式 AES 加密方法運算得 K’ 。

Step 17： K =? K’

伺服器端比對 K 是否等於 K’ ， 如果兩者相等，則使用者端身分正確，予以通過認證進入系統服務；如果 K 不等於 K’ ，則判定該使用者無法通過身分認證，不予登入系統。

第五章 實作成果

經由第四章對本研究提出的認證方法做詳細的介紹後，本章節依照設計的理念，說明利用 BMP 圖形檔案嵌入金鑰與使用 PHP 程式語言建置資訊隱藏技術的認證系統成果，之後再進一步評估系統的安全性。因此本章將先對系統的環境和成果畫面一一介紹，在第三節部分則對本系統的安全性做詳細的分析和比較。

5.1 實作環境介紹

本研究基於大多數帳號密碼認證系統的考量，希望只需少部分更動現有的認證系統環境下，加入圖形化密碼的認證優點於其中，並考量降低系統的建置成本來達成 SSL 認證系統，所以在選用網頁伺服器時，採用免費且穩定的 Apache Server 搭配功能強大的 PHP 網頁程式語言，來達到資訊隱藏技術與單向雜湊函數的功能，甚至允許各系統建置者自行加入更多功能。

而本系統所採用的硬體及軟體套件版本詳列如表 5，供實作系統環境之參考。

表 5 系統建置軟硬體規格版本

用途	名稱或規格版本
硬體環境 - CPU - RAM - HD	 Intel(R) Celeron(R) 4 CPU 2.40GHz 512MB 80G
作業系統	Ubuntu 9.10
伺服器軟體 - 網頁伺服器 - 資料庫	 Apache 2.2.12 MySQL 5.1.37
開發工具 - 伺服器端網頁程式語言 - 資料庫管理介面	 PHP 5.2.10 phpMyAdmin 3.1.3.2

5.2 成果介紹

本研究所提出的方法中，改進傳統帳號密碼型的登入模式，搭配一 BMP 圖形檔案金鑰，即可讓使用者擁有 256 bits 長的複雜型強密碼，有效提升認證的安全性與可靠性。使用者無需花費長時間記憶困難且複雜的強密碼文字，也改進現有多數圖形化密碼系統，需花費額外訓練時間的缺點，並簡化以往圖形化密碼如 PassPoints 等法所需多回合的密碼點選輸入步驟。相較於智慧卡認證系統而言，SSL 認證法無需額外的硬體設備，而且在圖形金鑰的傳遞上所花費的時間成本，比智慧卡的傳遞時間低上非常多，因此對於使用者的便利性也大幅提升。另外，對於系統服務端而言，僅利用 PHP 程式語言即可達到此效果，不用整套系統全部更新，只要做部分的修改即可，對於服務提供者而言也相當便捷。

圖 21 SSL 系統登入畫面

如圖 21 是本系統的登入畫面，使用者在此處可以選擇登入時是否使用圖形金鑰檔案進行認證動作，當使用者輸入自己的帳號與密碼之後，伺服器端便會進行初階段的核可動作，以進入如 4.3 節所述的不同登入階段。

當使用者輸入自己的帳號與密碼後即選用正常登入階段時，伺服器便會進行第一階段的認證，於使用者通過第一階段帳號與密碼的核可後，系統將會把使用者導入圖形金鑰選擇的畫面，如圖 22 所示，讓使用者進行圖形金鑰檔案的選擇動作，經使用者選取正確的圖形金鑰檔案位置，系統得以進行金鑰 P_{key} 取出與後續 P_{key} 比對的動作。

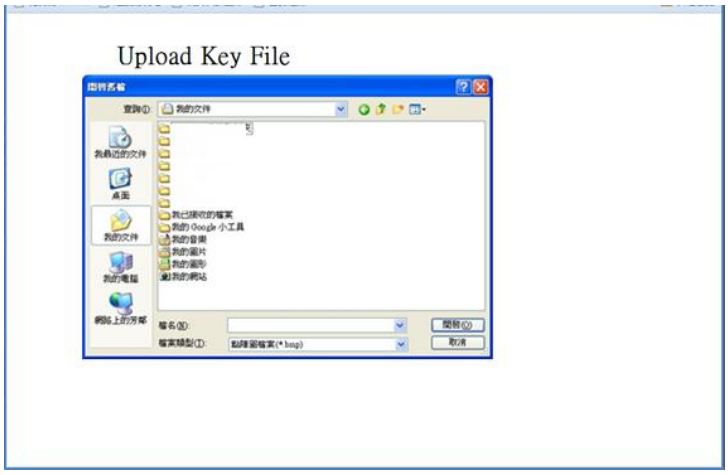

圖 22 選擇圖形金鑰檔案選取畫面

而圖 23 所呈現的，則是使用者所使用的 BMP 圖形檔案，圖 23 中右方的圖片是經 SSL 系統嵌入金鑰的圖形檔案，圖 23 中左方的圖片則是未嵌入任何資訊的原始圖形檔案，我們可以發現這兩張圖片在視覺上，是沒有辦法察覺有任何的差異，因此當攻擊者想要以視覺方式肩窺判斷金鑰圖形檔，將會是非常困難的。

圖 23 原始圖檔(左)與嵌入金鑰圖檔(右)比較

而萬一當使用者不慎遺失圖形金鑰或是遭惡意攻擊者取得嵌入金鑰的圖形檔案時，使用者可以利用 5.3.6 節所提供的混淆法，透過事先的圖形檔案處理動作，讓攻擊者難以探察出真正的圖形金鑰。而如圖 24 所示，可以發現嵌入金鑰的圖形檔案與沒有嵌入金鑰的圖形檔案，即使在檔案的大小上也是一模一樣，難以分辨兩者的不同，若是攻擊者想要藉著嵌入金鑰檔案的大小差異性，來作為判別的依據，也是完全不可能的。

原始檔案　　嵌入金鑰檔案

圖 24 圖形檔案大小比較

當使用者通過圖形金鑰驗證之後，所呈現的畫面，如圖 25 所示，使用者可以點選 Edit Account 來進行個人資訊修改，如密碼變更或圖形金鑰檔更新等動作。

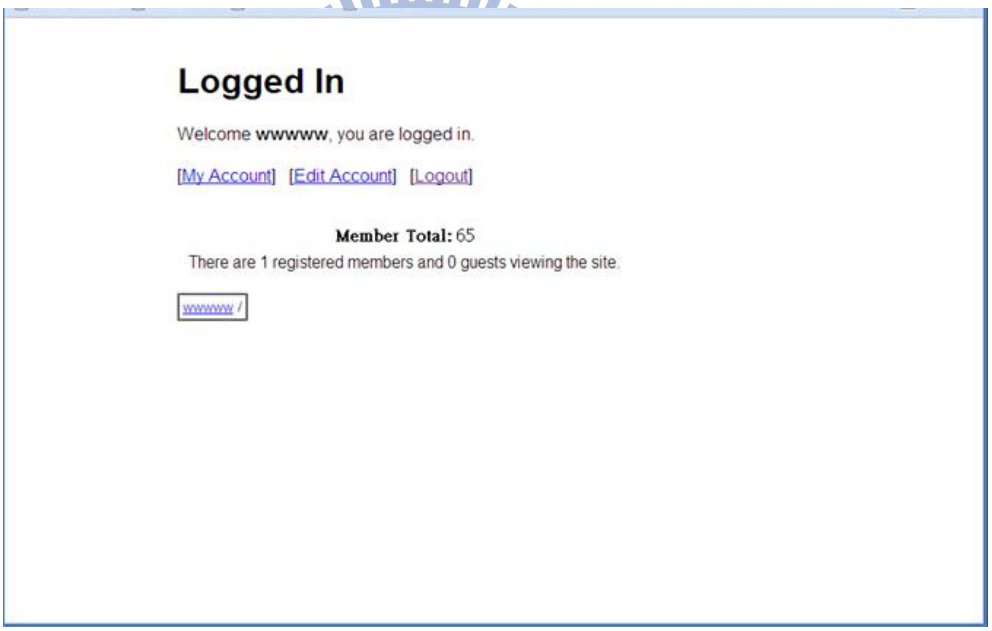

圖 25 使用者通過認證畫面

如果使用者想要修改個人資訊時，只要點選圖 25 畫面中的 Edit Account 選項，系統便會將使用者導入圖 26 的變更使用者資訊的頁面，在此頁面使用者可以進行電子郵件信箱變更、一般密碼變更或是圖形金鑰檔案更新等動作。若使用者點選更新圖形金鑰檔案之動作(Upload File)，系統則會進入 4.5 節所敘述的圖形金鑰更新動作，要求使用者提供新的圖形檔案，在系統嵌入金鑰完成後，一樣會要求使用者儲存已嵌入金鑰的圖形檔案。

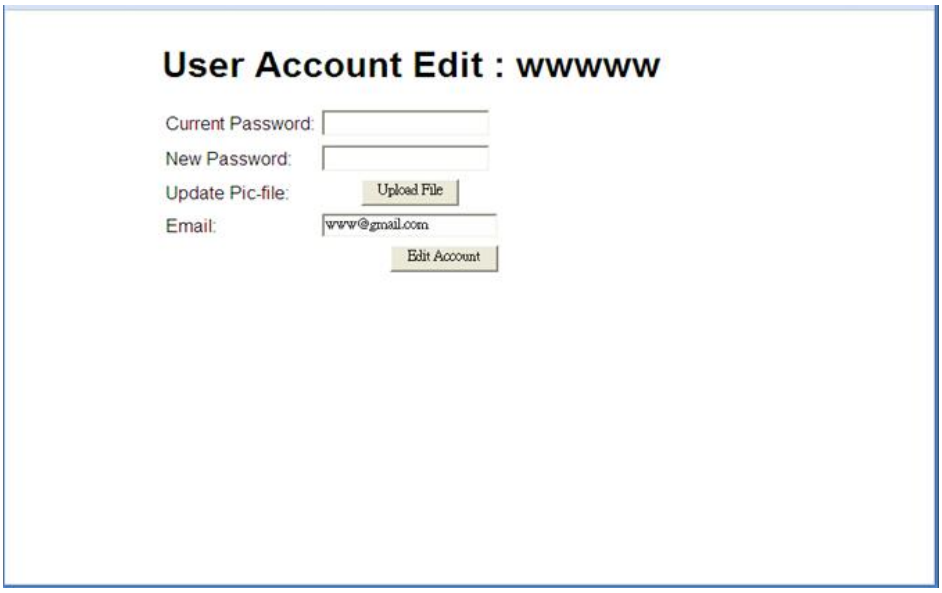

圖 26 使用者資訊變更畫面

SSL 驗證法由於著重在使用者的便利性的改進，將許多圖形化密碼認證方式的密碼訓練時間冗長的缺點，予以改進。但 SSL 驗證法仍需要使用者上傳與下載圖形金鑰檔案，因此在上傳與下載檔案的時間花費上，是否會降低使用者的使用意願，也是很重要的考量。

因此以下對於圖形檔案的上傳和下載時間，進行簡單的分析與說明，根據台灣網路資訊中心公布的 99 年「台灣寬頻網路使用調查」報告[27]，寬頻網路的使用普及率為 67.21%，而台灣地區家中連網方式為寬頻方式的比率高達 89.73%。其中，以ADSL連結上網的比例最高(68.72%)，其次是社區網路(13.70%)，Cable Modem 連結只有(8.98%)。所以此處以ADSL連線速率，對 700KB大小的BMP圖形檔案的上下傳時間，進行分析。

表 6 檔案 700KB 上、下傳時間參考表

ADSL	連線速率 bps	連線速率 KB/s	所需時間參考值 s
上傳	256 K	13 — 24	30 — 60
	1 M	50 — 100	8 — 15
	2 M	100 — 200	4 — 8
下載	256 K	13 — 24	30 — 60
	1 M	50 — 100	8 — 15
	2 M	100 — 200	4 — 8
	8 M	150 — 800	1 — 8

由表 6 可以看出 700KB 大小的圖形檔案，在上傳的時間上，最多可能需要一分鐘的時間，而最短只在五秒以內，而使用者需要執行上傳的動作僅發生在使用者註冊階段和圖形密碼更新階段，因此僅佔整個認證過程中的一小部分使用情

形。在主要的認證過程中，僅需傳遞小量的雜湊函數結果數值，來進行主要的認證動作。因此在時間花費成本上，相較於圖形化密碼的訓練時間或是智慧卡的傳遞時間上，均大大的降低使用者在時間成本的花費。

而在圖形金鑰嵌入的過程中，除了圖形檔案的上傳之外，另一個可能會讓使用者感到不便的，應該就是金鑰的嵌入和取出的時間，本研究所使用的圖形金鑰嵌入與取出的程式語言是 PHP 程式語言，而在檔案嵌入金鑰和取出金鑰的時間成本，以下表作一簡單分析和說明。

表 7 圖形金鑰嵌入與取出時間(sec)

檔案	嵌入金鑰時間	取出金鑰時間	檔案	嵌入金鑰時間	取出金鑰時間
PIC1	5.076	5.098	PIC2	4.235	4.272
PIC3	3.936	4.068	PIC4	2.287	2.145
PIC5	2.500	2.806	PIC6	5.681	5.581
PIC7	6.366	6.312	PIC8	3.731	3.777
PIC9	4.561	4.719	PIC10	2.952	2.969
嵌入金鑰平均時間		4.133	取出金鑰平均時間		4.175

表 7 中顯示十個不同的圖形檔案，在進行圖形金鑰嵌入與取出的時間比較，可以看出最短的金鑰嵌入時間約 2.3 秒，最長的時間約為 6.4 秒，平均的圖形金鑰嵌入時間約為四秒鐘；而在圖形金鑰的取出時間中，最短的約 2.2 秒，最長的也只需約 6.3 秒的時間，平均的金鑰取出時間約為四秒鐘。這樣的時間效益和圖形化密碼認證方法比較而言，所花的時間成本是比較少的。而相較於智慧卡認證方法，智慧卡的傳遞時間是最耗費使用者的時間成本，如以取得銀行帳戶的智慧卡為例，使用者必須親自到銀行進行智慧卡的申請動作，申請完成之後仍須等待三至七日的時間，方可再到銀行領取屬於自己的智慧卡進行使用。所以 SSL 認證法的時間成本，相較於圖形化密碼認證方法或是智慧卡的認證方式，在時間成本效益上是相對迅速許多的，對於使用者的便利性有很高的提升。

在使用者的易用性上，當使用者使用智慧卡進行認證時，除了需要使用所專屬的智慧卡外，還需特定的智慧卡讀取設備，方能利用 PIN 碼取出藏於智慧卡中的認證資訊。當使用者使用圖形化密碼認證方式時，部分的圖形化密碼認證方法，需要特別的設備才可以使用，大部分的圖形化認證方法，只需利用滑鼠或是觸碰式介面，進行圖片的點選動作即可進行認證，但是所有類型的圖形化密碼認證法，卻都有以下的缺點，對於視力有問題或色盲的使用者將會難以辨別系統的圖片，另外對於肢體動作不方便或操控點選有困難的使用者，也將很難有效率的使用圖形化密碼認證系統。而 SSL 認證法，使用者可以利用普遍可見的 USB 隨身碟裝置，儲存自己的圖形金鑰檔案，即可使用；對於視力不好的使用者也無須擔心無法辨

識圖片檔案，因為該圖形金鑰可以由使用者自行選取，於是可以大幅降低對於視力不佳或色盲使用者的不便利性。因此相較於智慧卡認證方法和圖形化認證方法而言，本研究的認證方法有最佳的使用者便利性。

表 8 本研究與其他認證系統的比較表

	SSL 認證系統	智慧卡認證（蔡佳倫 2005）	圖形化認證（PassPoints）
需額外設備	否	是	否
認證訓練時間	無	無(有智慧卡傳遞時間)	有
缺點	使用者攜帶挑選圖片	需額外設備	對少數使用者不便利
使用者友善性	最佳	可	佳
鍵盤側錄	可抵抗	可抵抗	可抵抗
螢幕側錄	可抵抗	可抵抗	不可抵抗
密碼量	圖檔大小 嵌入技術	晶片容量	認證圖量

另外，由於惡意攻擊的推陳出新，螢幕側錄與鍵盤側錄等攻擊方法，對於竊取使用者的重要資訊，無疑是可怕的威脅，因此認證方式是否可以抵擋這些攻擊，也是很重要的，SSL 認證方法，在使用者登入時，除了以鍵盤輸入帳號與密碼之外，仍須以圖片檔案所取出的金鑰，做為認證的重要依據，而在圖片檔案選取階段的動作，搭配混淆法的使用讓惡意攻擊者無法取得金鑰圖檔，便可以有效的抵擋螢幕側錄與鍵盤側錄等攻擊。

SSL 認證法，在實作上已嵌入 256 Bits 長度的金鑰，因此在安全性強度上也有很不錯的效果，如搭配 512 Bits 的金鑰長度，在實作上亦可以達成。相較於圖形化驗證方式的安全性強度，大多是憑藉在認證的回合數上，當要達到更高的安全性效果，便需提高認證的圖形點選回合數，但如此一來，便讓使用者登入的時間拉長，將會降低使用者的使用意願。而智慧卡的容量，則是在卡片製造時便固定而無法隨意變更，雖目前已有 64KB 大小的晶片智慧卡，不過當安全性需求以致要增加卡片容量時，勢必只有透過更換新的卡片一途。而在本法中，嵌入圖片的資訊容量主要取決於嵌入的資訊隱藏技術和檔案的大小，在變更上只需系統端的技術改變或是提高檔案大小的要求即可達成，相對於更換新的智慧卡來說，不論是對使用者或是服務端都比較簡便，所耗費的成本也較少。

因此，SSL 認證法在使用性上是三種類型中最佳的方法。而且本法結合使用者容易記憶圖形的優勢以記憶金鑰檔案，又可以抵擋螢幕側錄與鍵盤側錄等多種攻擊法，在安全性效果的強度也很足夠，使用上也不需要額外的設備支援，相信

對於使用者或是系統服務端來說都是最佳的認證選擇。

5.3 安全性分析與比較

由於本研究的認證方法是改進以帳號、密碼為基礎的認證方式而來，以下將對幾種常見的攻擊方法做安全性的分析，以確認本研究提出的認證法之安全可靠性。

5.3.1 重送攻擊(Replay Attack)

本研究的方法為有效的避免重送攻擊(Replay Attack)，因此在使用者每次提出認證的連線階段中，在使用者端、伺服器端都會產生一隨機亂數(Random Nonce)，讓攻擊者無法藉著竊聽使用者、伺服器之間的一次連線認證資訊之後，將所竊聽到的認證資訊重送到伺服器端，而獲得認證並進入系統之中。由於每次的認證連線時，所產生的亂數是不一樣的，因此所傳送的資訊都不相同，攻擊者便難以在少數幾次的竊聽之下，破解並重送認證資訊到系統端，進而假冒使用者身分進入系統，況且伺服器端在驗證數次以內均錯誤，便會將該使用者判斷為非法的使用者，系統隨即終止該使用者的登入請求，所以本認證機制可以避免惡意的重送攻擊(Replay Attack)。

5.3.2 伺服器偽裝攻擊(Server Spoofing Attack)

由於本研究所提出的驗證機制中，要求在伺服器驗證使用者的合法身分之前，會先進行伺服器的確認動作。因此，攻擊者如果想要假冒伺服器以騙取使用者的認證資訊，則該攻擊者必須取得 $O_S=H(U_{PW})\oplus H(P_S)$，即雜湊函數 H( )、使用者的密碼 U_{PW} 及使用者的圖形金鑰檔案 P_S 等資訊，才能成功假冒伺服器來欺騙使用者而獲取使用者驗證用的資訊。一旦使用者端發現到有假冒伺服器欺騙動作進行時，使用者端的程式便會中斷整個驗證流程，因此本研究所提出的驗證機制可以有效的避免伺服器偽裝攻擊(Server Spoofing Attack)。

5.3.3 驗證表被竊攻擊(Stolen-Verifier Attack)

本研究的驗證機制中，伺服器存放的使用者驗證表 $O_S = H(U_{PW}) \oplus H(P_S)$，僅為認證的部分資訊，攻擊者仍須取得使用者的圖形密碼金鑰 $H(U_A||S_A) = P_{key}$，方能有效取得使用者權限。且圖形金鑰 P_{key} 並無存放驗證表於伺服器系統中，僅存放該圖形金鑰 P_{key} 產生所需的資訊，攻擊者還得取得系統資訊 S_A，使用者資訊 U_A，雜湊函數 H( )演算法等，方可有效取得使用者權限，因此本研究所提出的驗證機制可以有效的避免驗證表被竊攻擊(Stolen-Verifier Attack)。

5.3.4 階段鑰匙的安全性(Session Key Security)

本研究的驗證機制中，使用者與伺服器之間的階段鑰匙(Session Key)的決定因素是利用 $H(U_{PW}) \oplus H(P_S)$、使用者端隨機亂數 N_u、伺服器端隨機亂數 N_s 三個參數所組成，其中隨機亂數 N_u 和 N_s 在每次的登入連線時都不一致且獨立，所以即使攻擊者透過攻擊而得知該使用者前一次登入的階段鑰匙(Session Key)的值，也無法預測出次一回或其他登入階段的階段鑰匙。另外，當這些資訊在傳送時，都會先經過雜湊函數的處理，因此攻擊者更加難以破解，因此本研究所提出的驗證機制在階段鑰匙(Session Key)的安全性上是非常安全的。

5.3.5 使用者假冒攻擊(Impersonation Attack)

本研究所提出的 SSL 驗證方法，當攻擊者想要利用使用者假冒攻擊(Impersonation Attack)來假冒成合法的使用者入侵系統時，SSL 方法首先可以藉著伺服器所產生的隨機亂數 N_s 和使用者所產生的隨機亂數 N_u 保護之下，來避免攻擊者得知 $H(U_{PW}) \oplus H(P_S)$ 的值。即使該攻擊者，透過方法取得合法使用者的帳號密碼與 $H(U_{PW}) \oplus H(P_S)$ 的值，攻擊者仍須取得該使用者的圖形金鑰 P_{key}，方能有效的取得登入系統的權限，更何況上述的幾個數值，在傳送時會藉由雜湊函數的保護。而使用者的圖形金鑰 P_{key}，其取出方法，僅存有在伺服器端，攻擊者想要直接利用圖形檔案得知圖形金鑰 P_{key} 也不可能。因此本研究所提出的驗證機制對於使用者假冒攻擊(Impersonation Attack)是可以有效避免的。

5.3.6 肩窺(Shoulder Surfing)

肩窺(Shoulder Surfing)對於圖形化密碼認證，實為安全性上的重要隱憂。

因此提出混淆法，來降低肩窺的安全性威脅，如圖 27 所示。由於經由 SSL 系統產生的圖形金鑰檔，其檔案大小與原始檔案相同，因此使用者可以利用相同圖形檔案，以不同檔案名稱儲存，但其中僅有一個檔案為真正藏有金鑰的圖形檔案，來混淆肩窺者的判斷，提升安全性。也由於系統認證端，對於使用者認證的錯誤次數多有限制，因此也建議使用者在該資料夾下圖形檔案的數量，可以大於錯誤次數的兩倍甚至三倍以上的檔案數量，才能有較好的安全性效果。使用者也可以利用系統所提供的假金鑰檔案產生動作，來方便達到此一效果。

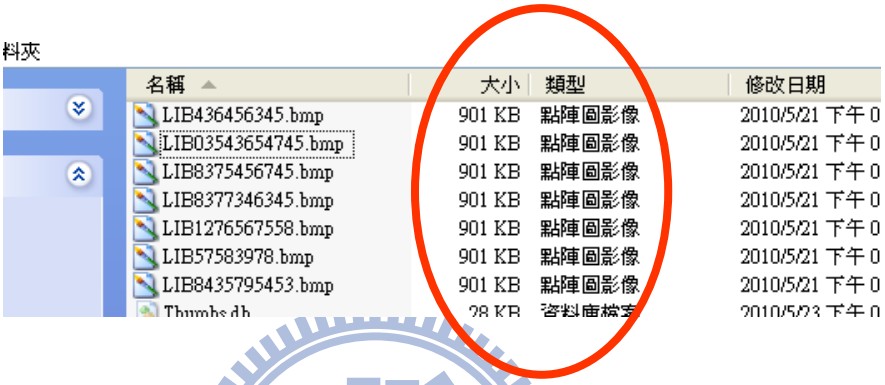

圖 27 混淆法圖示

而此處提出的混淆法，除了可以抵擋肩窺攻擊外，由於只有使用者自己知道圖形金鑰檔案的選擇規則，而該圖形檔案又有使用者自行選擇的優勢，因此對於色盲等視力不佳的使用者而言，在使用上的困擾性相較於 PassPoints 等圖形化密碼系統而言是比較低的。甚至當使用者利用隨身碟儲存圖形金鑰，卻又不慎遺失時，也不需要擔心圖形金鑰遭人盜用的問題。

第六章 結論與未來方向

近年來網路服務愈來愈多，接踵而來的使用者認證問題和伺服器攻擊行為也與日俱增，這些問題一直困擾者服務提供者和使用者，雖然利用帳號密碼認證的模式仍佔多數，但對於安全性的考量之下，如圖形化密碼、智慧卡、生物資訊辨識的認證方式有愈來愈多的系統服務者採用，但對於使用者來說，記憶煩雜的高安全強度密碼，實在難以達成，因此本研究採用圖形化密碼的優點結合智慧卡認證安全性的效果，以較低的成本、較高的使用者便利性和記憶性，來提供使用者簡便的方式和服務提供者較高的認證安全性。

6.1 結論

本研究為改善使用者和服務提供者雙方的身分認證問題，利用人對於圖形有較好的記憶能力與資訊隱藏金鑰的方法，在小幅度變更當今多數的使用者認證系統環境、不增加額外設備下，提出一個減少使用者記憶負擔，方便系統服務提供者改進，而且可以提高安全性的一種認證機制，其特點重述如下：

1、認證系統利用PHP程式語言即可達成，而原為帳號密碼認證模式的系統，僅需小幅度變更，即可獲得較高的安全性效果以及有良好的擴充性。

2、利用人類對於圖形有優越的記憶性，取代冗長難記憶的強密碼字串，又沒有過去圖形化密碼認證的訓練時間花費，降低使用者的時間成本，對於使用者有高度便利性。

3、價格低廉，無需額外特定設備，使用者也可利用USB隨身碟，便可以簡便攜帶圖形密碼檔案，並且達到防制肩窺攻擊的效果。

6.2 未來方向

本論文提出的 SSL 認證方法可以有效提高使用者便利性與系統認證安全性，但仍有未臻完美之處，未來本研究可朝以下三個方向進行：

1、支援多類型之圖片檔：

本研究所提出的方法只針對單一類型的圖片檔案進行應用，在檔案格式繁多的資訊時代，對於使用者的選擇性和有心攻擊者的破解可能性都有影響，如能提

供使用者多樣類型的圖片格式選擇，相信必能更加提高使用者的使用意願，以及讓攻擊者也更難以找出固定模式而加以破解。

2、單一登入(Singel Sing-On)：

在網路使用環境日益便利的現今，各式各樣的網路服務充斥，使用者所需記憶的帳號和密碼越來越多。因此，如果可以利用本研究的機制，加以改進而達到單一登入的效果，甚至將使用者的帳號與密碼等登入資訊，直接嵌入在檔案中，相信對於使用者而言，是非常方便的。

3、大量使用者登入的改善機制：

由於本研究僅有少量使用者進行認證效果測試，對於使用者數量眾多的服務提供者而言，在系統遇到大量使用者同時登入時的負載效能，必須再行測試以求更好的服務效果。

參考文獻

[1] 謝續平，交通大學網路安全授課資料，Available from:http://dsns.csie.nctu.edu.tw/course/netsec/2004fall/。

[2] Rachna Dhamija, Adrian Perrig, "Déjà Vu: a user study using images for authentication", Proceedings of the 9th conference on USENIX Security Symposium, p.4-4, Denver, Colorado, 2000, August.

[3] Helene Intraub,"Presentation rate and the representation of briefly glimpsed pictures in memory", Journal of Experimental Psychology: Human Learning and Memory,6(1):1–12, 1980.

[4] Steganography，Available from:http://en.wikipedia.org/wiki/Steganography。

[5] F. A. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information Hiding—A Survey", Proceedings of the IEEE 87, No.7, 1062–1078, 1999.

[6] L. F. Cranor and S. Garfinkel, "Secure or Usable? ",IEEE Privacy and Security, Vol. 2, PP. 16-18, 2004.

[7] S. Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy, N. Memon, "PassPoints: Design and Longitudinal Evaluation of A Graphical Password System", International Journal of Human-Computer Studies, Vol. 63, PP. 102-127, 2005.

[8] A Habibi Lashkari, R Saleh, S Farmand, OB Zakaria,"A wide-range survey on Recall-Based Graphical User Authentications algorithms based on ISO and Attack Patterns",(IJCSIS) International Journal of Computer Science and Information Security, Vol. 6, No. 3, 2009

[9] Symmetric Encryption，Available from:http://en.wikipedia.org/wiki/Symmetric-key_algorithm。

[10] Asymmetric Encryption，Available from:http://en.wikipedia.org/wiki/Public-key_cryptography。

[11] 卡威科技，「IC智慧卡(Smart Card)與電子商務」，Available from:http://www.cardweb.com.tw/304ICS/ICCardInfo/ic_EC.htm。

[12] G. E. Blonder, "Graphical password", United States Patent 5559961, Lucent Technologies, Inc., Murray Hill, NJ, August 30, 1995.

[13] L. Standing, J. Conezio, and R.N. Haber. "Perception and memory for pictures:Single-trial learning of 2500 visual stimuli".Psychonomic Science, 19(2):73–74, 1970.

[14] W. Bender, D. Gruh, N. Morimoto and A. LU: "Techniques for data hiding", IBM System Journal, 35, pp 313-336, 1996.

[15] D. Weinshall and S. Kirkpatrick, "Passwords You'll Never Forget, but Can't Recall," in Proceedings of Conference on Human Factors in Computing Systems (CHI). Vienna, Austria: ACM, 2004, pp. 1399-1402.

[16] 郭信東，「一個新的圖形密碼方法」，亞洲大學，碩士論文，2007。

[17] J.K. Lee, S.R. Ryu, and K.Y. Yoo, "Fingerprint-based remote user authentication scheme using smart cards", Electronics Letter, 38, 12, pp. 554-555, 2002.

[18] 蔡佳倫，「遠端使用者身分驗證之研究」，國立交通大學，碩士論文，2007。

[19] Analysis of 32 million breached passwords， The Imperva Application Defense Center (ADC)，Available from:http://www.net-security.org/secworld.php?id=8742。

[20] John R. Anderson and Christian Lebiere, "The Atomic Components of Thought", Lawrence Erlbaum Associates Inc., 1998.

[21] Telegraph Media Group，Security risk as people use same password on all websites，存取於 2009 年 9 月 27 日，http://www.telegraph.co.uk/technology/news/6125081/Security-risk-as-people-use-same-password-on-all-websites.html。

[22] Josh Catone，Bad Form: 61% Use Same Password for Everything，存取於 2009 年 9 月 27 日，http://www.readwriteweb.com/archives/majority_use_same_password.php。

[23] Steve Ragan，Internet users still using same password for all Web sites，存取於 2009 年 10 月 1 日，http://www.thetechherald.com/article.php/200911/3184/Internet-users-still-using-same-password-for-all-Web-sites。

[24] Adrian Perrig and Dawn Song, "Hash visualization: A new technique to improve real-world security", In Proceedings of the 1999 International Workshop on Cryptographic Techniques and E-Commerce (CryTEC '99), 1999.

[25] Adams A., Sasse M., & Lunt P. , "Making passwords secure and usable", People and Computers, 1-20,1997.

[26] Wu T. , " A real-world analysis of Kerberos password security" , Paper presented at the Proceedings of the 1999 Network and Distributed System Security Symposium, San Diego, California, USA , 1999 .

[27] 台灣網路資訊中心，「台灣寬頻網路使用調查」，存取於 2010 年 4 月 10 日，http://www.twnic.net/ibnews.php

[28] L. Standing, J. Conezio, and R.N. Haber. "Perception and memory for pictures:Single-trial learning of 2500 visual stimuli".Psychonomic Science, 19(2):73–74, 1970.

[29] Ian Jermyn, Alain Mayer, Fabian Monrose,Michael K. Reiter, and Aviel D. Rubin, "The design and analysis of graphical passwords", In Proceedings of the 8th USENIX Security Symposium, August 1999.