Threat Modelling in User Performed Authentication

Xun Dong, John A. Clark, and Jeremy L. Jacob

Department of Computer Science,
University of York
United Kingdom

Abstract. User authentication can be compromised both by subverting the system and by subverting the user; the threat modelling of the former is well studied, the latter less so. We propose a method to determine opportunities to subvert the user allowing vulnerabilities to be systematically identified. The method is applied to VeriSign's OpenID authentication mechanism.

1 Introduction

Criminals often seek to exploit a user's inability to distinguish the legitimate from the faked. *'Phishing'* attacks [1,13] are the most familiar examples; users are conned into taking actions that prove against their interests, typically resulting in the release of confidential and valuable information. Users may be regarded as complicit in such exploitation, but in many cases labelling of the user as 'the weakest link' merely covers up the fact that the systems are not designed to prevent such attacks or make them difficult. Users have reached their present exploitable state, aided and abetted by poor system design.

There is a deeper problem: there would appear to be little in the way of systematic *analysis* concerning the user's role in security. If users are now the weakest link then user-side threat modelling is as important as system-side threat modelling. In this paper we provide a user-focused threat identification approach for user authentication systems, particularly those used over the web. We hope our efforts will inspire further user-oriented threat modelling.

2 Background

Researchers have investigated security-relevant user behaviour addressing questions such as: How often are passwords reused [6]? How likely are users to choose secure passwords [6]? How effective are the security indicators shown on web browsers [3, 11, 14]? What factors influence users' trust in a website [2,4,5,7,8,9,10]? Why do users fall victim to phishing attacks and how can phishing interactions with users be modelled [3, 4]? The practicality of such research may be limited. Findings may be too closely linked to current technology and

L. Chen, M.D. Ryan, and G. Wang (Eds.): ICICS 2008, LNCS 5308, pp. 49–64, 2008.

could become invalid very quickly as systems, technologies, and user characteristics and behaviours evolve. Also, most studies are focused on specific problems, and there is little in the way of method to help system designers and security practitioners to systematically identify the threats to the users of systems.

Authentication is an interaction between a small group of entities (typically two) that aims to establish to each entity that the others have particular properties (often some notion of *identity*). We will focus on authentication between a *user* of a web service and an *external entity* (EE) that provides a service, or is a gateway to a service. We are interested in user-to-EE authentication and EE-to-user authentication. Both are typically achieved by proving ownership of certain objects (most typically secret knowledge such as passwords or encryption keys). While user-to-EE authentication is controlled by designers and consistently enforced by systems, designers often have little control over how EE-to-user authentication is carried out. Users must be able to first accurately establish the identity of the EE and then be able to judge whether this identity is entitled to request confidential information. The latter may seem straightforward for the user, but in some cases (as shown in section 4.2) it will be difficult. Failure of either part may lead to a user giving out authentication credentials to attackers.

Attacks against web authentication systems may be *passive* or *active*. Passive attacks do not require active victim involvement, often achieving their goal by analysing information available to attackers (e.g. that from public databases or websites, or even rubbish bin contents). Many are launched by insiders or people who have close relationships with the victims. Active attacks exploit the user's difficulty in authenticating EEs, requesting the user's authentication credentials whilst posing as trustworthy parties. Typical examples are phishing and pharming attacks. Mixed attacks are possible; some attacks have an initial passive phase to gather information and then use the information in a later active phase.

3 Vulnerabilities Exploited by Passive Attacks

3.1 Properties of Users' Authentication Credentials

A user and an EE share a collection of authentication credentials, typically including PINs, passwords, and so on. They also include an identifier unique to the user. Each credential can be classified along four axes: *mode*, *factor*, *assignment* and *losability*.

Mode. Primary or Emergency.

By *primary credentials* we mean those used to directly access assets and functionalities guarded or provided by external entities. Similarly *emergency credentials* denote those used to reset or recover the primary credentials. Attackers can masquerade as a user if they obtain primary or emergency credentials.

Factor something users know; something users possess; something users have access to; or characteristics of who users are.

The factor axis is based on the traditional way of classifying authentication approaches but with one addition. 'Something users have access to' is usually included in 'something users possess'. We distinguish it because it has different implications for the security of an authentication system: it creates an authentication security dependency relationship. It is not directly possessed by a user, but the user holds the keys (authentication credentials) to access them. For example, an email account is not directly possessed by users, it is typically the property of an email provider, but a user with the correct password can access the content in that email account.

Assignment by the system; by the user; or by a third party;

Assignment by the system can ensure that certain security requirements are met (for example, that values of the authentication credentials are unique, and difficult to guess).

A user may not find the value assigned by system usable. User defined values may have high usability, but the system has limited control over whether security requirements are met.

When the value is assigned by a third party, the security properties depends on the behaviour of the third party. If the value of the authentication credential is predictable or easy to replicate, then this vulnerability could lead to the compromise of the current system.

Losability losable; or unlosable.

Losability indicates whether credentials are likely to be lost. If an authentication credential is losable, the authentication system must provide methods for users to recover their ability to authenticate.

3.2 Authentication Credentials Vulnerable to Passive Attacks

Passive attacks require authentication credentials to be exposed to third parties. Credentials known only to the system and the user have low exposure; credentials accessible to the general public (for example when a value has been published on a web page) have high exposure; otherwise the exposure is medium. The exposure level can be determined by considering the authentication credential factor basis, how its value is assigned, and the choice of authentication credentials by other authentication systems.

The exposure level of authentication credentials that are based on personal data and who users are, can only be medium or high. For example, a user's date of birth or mother's maiden name are known to close friends and relatives, and may even be available on a public database. Data that describes who users are, such as finger prints, are inevitably exposed to objects users have touched and all systems which use finger prints as authentication credentials.

Password exposure level can be low if it is the system that assigns the value, and the exposure level is uncertain prior to the assignment if it assigned by the user (since personal practices will differ).

Any authentication credential with mdium or high exposure level is vulnerable to a passive attack. Attackers may obtain data from those parties to whom the credentials have been exposed. For example there are companies that sell users' personal contacts and other personal information they collect. Hence it is very difficult to determine who has access to any high or medium exposure credential.

The complete set of a user's authentication credentials can be divided into subsets, each of which is sufficient to prove the identity of the associated identifier. The subsets always include at least one subset whose members are all primary authentication credentials, and may also contain other subsets of emergency authentication credentials. If those emergency credentials can be used to recover and/or reset the primary credentials and the primary credentials are all assigned by users, then the compromise of the emergency credentials is as serious as the compromise of primary credentials. To compromise one's account, attackers must obtain one of the subsets.

To determine whether attackers can obtain any of the subsets by applying passive attacks, analysts must find out the exposure level of each subset. The exposure level of a set of credentials is the minimum of its members. For example, if a subset has three members with exposure levels of high, medium, and low, then subset's exposure level is low. If a subset has high exposure level then the subset can be obtained using passive attacks; if a subset has a medium exposure level, then parties to which the credentials have been exposed can obtain the credentials using passive attacks. For any authentication system, designers should make sure there is no subset whose exposure level is high. When there are subsets whose exposure level is medium, then designers must assess how likely the parties to which the credentials are exposed are to launch attacks against the user. Design improvements can be taken if the exposure levels and security requirements warrant it.

3.3 The Authentication Security Dependency Graph

Authentication security dependency relationships. If compromise of system 'B' directly leads to the compromise of 'A' we say that the security of 'A' depends on the security of system 'B'. If any of the user authentication credentials are in the category 'what you have access to' or is created or assigned by a third party then effectively the designers may have created a dependency of the current system on the third party. For example, the access right to a secondary email account is often used as one of the emergency authentication credentials to reset or recover primary authentication credentials. In those cases, compromise of the email account allows attackers to gain access to the authentication system by resetting or retrieving the primary authentication credentials.

Drawing the dependency graph. Analysts should identify the dependency relationships and represent them in *an authentication security dependency graph*. Each node in the graph represents a system, and the start node of the graph is the

system being designed. Directed edges are included from Node 'A' to Node 'B' exactly when Node 'A' depends on Node 'B'. If 'B' also has such dependency relationships then the graph can be expanded further.

If the value of the authentication credentials which create such dependency relationships are assigned by users then the dependency relationships may be unpredictable and the graph cannot be determined. For example, if a user provides an email account as an emergency credential then it is impossible to predict all the email service providers that the current system will depend on.

Vulnerabilities. The dependency graph has two implications for the system being designed:

1. The security of the current system is equal to the security of the weakest system reachable in the graph; and
2. Obtaining authentication credentials to the weakest system propagates access back up the reachability chain.

The first implication means that the security of the current authentication system could be reduced if there is a weaker authentication system in the dependency graph. Many financial related websites have educated users to choose strong passwords and pay more attention to security indicators when accessing an authentication web page. Most users are likely to behave cautiously and securely when dealing with web sites they categorise as financial-related and important, but tend to use weak passwords and pay less attention to security for the rest [6]. However, the security of authentication of such financial-related web sites may not be strong, if some reachable node in the dependency graph from the financial-related site is treated less seriously. In fact, it is common for such web sites to ask users to provide a secondary email account as an emergency credential, while most users think that the security of their email account is less important than that of the financial-related web site.

The second implication means that the dependency relationships create new channels through which an authentication system may be attacked. The new channels are the ones that attackers could use to compromise the other systems in the graph. Moreover, the new channels can not be mitigated by the design of the authentication system.

Any dependency relationship should be viewed as a vulnerability, especially those which are unpredictable, and they should be avoided or minimised at the design stages. For unavoidable dependency relationships analysts should design the authentication system in a way that the authentication credentials that create the relationship are not used alone to prove identity. For example, access to the email account must be used together with a number of security questions to prove a user's identity.

4 Vulnerabilities Exploited by Active Attacks

In the second stage, analysts should consider vulnerabilities that active attacks exploit. Our method considers phishing and pharming attacks.

4.1 Sensitivity of the Authentication Credentials

Section 4.1's reference to determining sensitivity level is OK in terms of indicating *what* needs to be done, but there is no information on *how* to do it, or indeed the practical feasibility of doing it. Sensitivity indicates the likelihood of the user being suspicious or alert when an external entity requests the authentication credential. If the user is very alert then the sensitivity is high, otherwise it is low. System designers should choose authentication credentials in a way that the sensitivity is as high as is practical.

User must be alert to the malicious request of authentication credentials. As mentioned in section 3.2, the user authentication credentials can be divided into several subsets. An analyst can predict the likely alertness of a user by examining the sensitivity level for each subset. The sensitivity of a subset is determined by the member with the highest sensitivity level. If there is a subset whose sensitivity is low, then there is a vulnerability. At least one subset should have its sensitivity at medium level, if possible every subset should have a high sensitivity level.

A credential data item's sensitivity level is subjective, and users' sensitivity levels for an data item may vary. In the process of determine the sensitivity level, analysts should use common sense, consulting a group of users if necessary.

4.2 Identify Potential Impersonating Targets

In active attacks, attackers need to impersonate a legitimate external entity (EE). It would be wrong to think that the impersonating target is only the current system. Attackers may impersonate three types of EE: the EE that the user has shared authentication credentials with; EEs that are entitled to request users' authentication credentials or initiate user-to-EE authentication; and the EEs that exist in the authentication dependency graph.

The first type are normally EEs with which the user has set up an account. However, there are some exceptions, for example, the single sign-on system such as OpenID. Here users set up an account with both an OpenID provider and the service provider website. The user shares its authentication credentials with the OpenID provider but not the service provider website.

The second type of EE can be difficult to identify. A company may have a number of websites, and users can use the same authentication credentials to access the services provided by all of them. It also happens when companies or organisations use the single sign-on system such as OpenID, in which users can use the same set of authentication credentials to access services provided by all participating companies. Another typical example is the credit/debit card authentication system: the card details are assigned and shared between the bank and the user, but online retailer websites may be entitled to request card details from its users. In all these examples there is no convenient mechanism for analysts and users to find out who are legitimate entities.

The third type of EEs can be identified by constructing the dependency graph.

Among the impersonating targets identified, if there are EEs whose authentication system designs cannot be influenced by the system designers, then there is

a vulnerability that may be exploited. If the authentication system of such an EE is not designed or implemented properly, attackers might choose to impersonate that EE instead of impersonating the EE that designers can influence.

If EEs other than those users have shared credentials with may request them, and there is no reliable method to conveniently prove an entity is entitled to do so, then a vulnerability will be created – attackers could acquire users' credentials by claiming to be one of those further entities.

4.3 Active Attack Entry Point Analysis

Analysts should document all impersonating targets, and then carry out active attack entry point analysis for the targets that are within the control of the current system's designers. This is achieved by first identifying the entry points, and then analysing vulnerabilities at each entry point.

The lifecycle of authentication credentials. Figure 1 shows the states in the general lifecycle of authentication credentials and the transitions between them. The optional state and transitions between states are represented by dashed lines. There are seven states authentication credentials could be in:

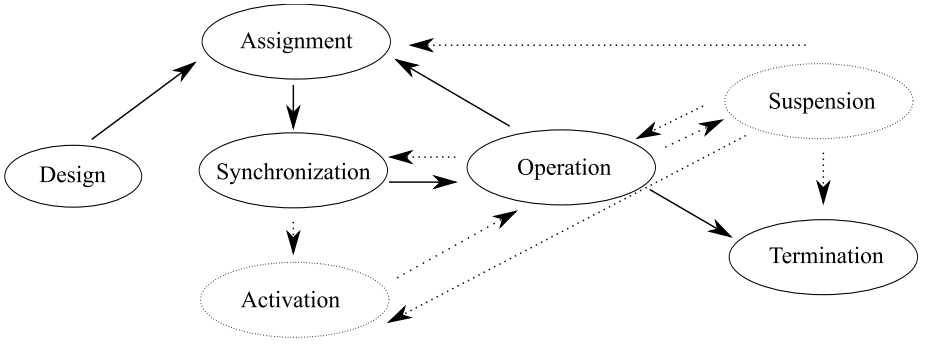

Fig. 1. The Lifecycle of Authentication Credentials

Design: Designers decide three things at this state : over which communication channel the authentication will be carried out; what authentication credentials should be used; and their lifecycle. The decision should be made based on the requirements for security, usability, constraints on economics and the properties of the authentication credentials (described in section 3.1). Our threat modelling should also be carried out in this stage.

Assignment: In this state the value(s) of the authentication credential(s) for a particular user will be created. Only the assigner knows the value of the credential(s).

Synchronisation: The party who assigned the value informs the other party of the value it has chosen. The time taken varies with the communication channel

used. If users supply credential values via webpages then synchronisation could be immediate. For credentials exchanged by the postal system (e.g. a PIN number for a new cashpoint card), then the synchronisation could take a couple of days.

Activation: Some systems may require users to activate authentication credentials before they can be used.

Operation: Users supply their primary authentication credentials to authenticate themselves to external entities.

Suspension: The current authentication credentials temporarily cease to function, e.g. 'lockout' after three failed authentication attempts. Upon satisfying certain requirements authentication credentials can be tranformed to other states.

Termination: Here current authentication credentials permanently cease to function. The account may have been terminated by the system or the user.

Analysts should identify over which communication channels the authentication credentials are exchanged during each state and each transition between states. For the transitions originating from the operation state, analysts should also check whether the transition requires proof of identity. A vulnerability is created if the transition can be carried out without authentication, as attackers can request the transition without possessing any authentication credentials.

The activation and suspension states are optional. Any authentication credential should pass through the remaining five. However, transitions between states vary for different authentication credentials. A typical authentication credential's lifecycle starts at the design state before moving to assignment and synchronisation. Depending on the actual authentication system, there might be an activation state before the operation state. From operation it can reach four states: suspension, termination, assignment, and synchronisation. It often reaches assignment because the user or system decides to reset the current value of the authentication credentials. Not every system allows authentication credentials to transition from operation to synchronisation. But when it does, it is often due to loss of authentication credentials. For example, when a user forgets his password, the user ask the system to send him/her the password.

Transitions from the operation state can be triggered by events from both users and systems. When it is triggered by users, users normally are required to prove their identities by using emergency authentication credentials. On the other hand when the event is triggered by the system, the system will need to inform its users about the transition between states and may require users to complete the transition. If there is no rigorous EE authentication mechanism for users to reliably prove the EE's identity in the communication, then phishers could impersonate the trusted EE. That's why it is necessary to analyse the vulnerabilities of EE authentication within the communication between users and external entities.

Entry points. Active attacks can only obtain user's authentication credentials when they are exchanged. By using the lifecycle analysts can identify in which states and in which transitions this occurs and focus vulnerability analysis

on those entry points. Using the lifecycle, we have identified the following six situations where a user's authentication credentials could be exchanged: 1)Synchronisation State; 2) Operation State; 3)state transition from operation to assignment; 4)state transition from operation to synchronisation; 5)state transition from suspension to assignment; 6)state transition from suspension to operation.

It is quite obvious the why a user's credentials are exchanged in both synchronisation and operation states. The transitions from the operation state can take place only when users have proved their identities. As a result users' emergency credentials will be exchanged. For example, when a user loses his primary credentials, such as password or USB token, the user needs to prove his identity to reset a new one.

Communication channels. The communication channel (CC) is a system or a method through which external entities can interact with human users. The most typical CCs are: 1) Emails; 2) Mobile phone messages; 3) Phone calls; 4) face to face communication; 5) webpages; 6) Instant Messenger; 7) Physical letters or notes. Authentication is a special type of interaction and it also operates via a CC. Each channel carries a different type of information, identifies entities in different ways, incvolves different agents, etc. As a result, EE-to-user authentication has different characteristics and vulnerabilities in different CCs. For a channel the following factors should be considered:

- How external entities are identified on the CC;
- How identifiers can be proved on the CC;
- Which Agents are involved in this CC.

The characteristics and vulnerabilities in EE-to-user authentication mainly depend on the communication channels over which the authentication is carried out. A limited number of CCs exist, so analysis of CCs could be carried out independently and the results can be used as a reference. Web page and email interactions are examined below:

Web page Interactions. Web pages are identified by their URLs. The integrity of website domain names are often proved by using an SSL/TSL certificate or an Extended Validation certificate. The agents involved in this communication channel can be classified as: client agents, server agents, and infrastructure agents. The client agents include:

- web browser
- operating platform (including the client computer's hardware and operating system)
- client side networking components: local router, gateway)

The server agents are the website servers. The infrastructure agents include:

- Domain Name Servers;
- Data delivery components;
- Certificate validation servers;

Email. An email is identified by its sender's address. To prove the email originates with the claimed sender SSL/TSL certificate can be used. The agents involved in this communication channel can be classified into three categories: Sender's agents, Receiver's agents, and infrastructure agents. The receiver's agents include:

- Email Client;
- Operating platform (including the client computer's hardware and operating system)
- Client's side networking components; (local router, gateway)
- IMAP/POP3 servers;

The sender's agents are responsible for delivery or relaying of the emails, for example, SMTP or MX Servers; The infrastructure agents include:

- Domain Name server;
- Data delivery components;
- Certificate validation server;

Entry points vulnerability analysis. For each attack entry point analysts determine the EE authentication vulnerabilities. These may arise due to:

- no reliable and sufficient authentication information is provided to users;
- users lack knowledge; and
- security design assumptions concerning users do not hold in practice;

Reliability and Sufficiency of Authentication Information. For successful EE-to-user authentication users must have reliable and sufficient authentication credentials. Because users mainly rely on authentication information presented to them to establish an external entity's identity, without them they can not accurately distinguish legitimate entities from the rest [4].

First analysts must determine the reliability of the authentication information provided. The authentication credentials the entity used to establish its identity, and all the agents involved on the channel on which the authentication system is implemented should be identified. This can be easily done by referring to the analysis of communication channels. The most important step is to check whether the compromise of any agents would make any EE authentication credentials untrustable. Then based on the protection of those agents, analysts can estimate how likely those agents would be compromised. If all agents were protected properly, then the reliability of that authentication credential would be high, otherwise it would be low.

Analysts can find out whether enough authentication information has given to users by checking first if users have been given the external entity's identifier. If the identifier's reliability is low given the previous analysis (which means it can be easily spoofed), then check whether users have been given additional reliable authentication credentials. If not, then users have insufficient information to carry out EE authentication.

Knowledge. Users need both technical and contextual knowledge to decide whether to release the credentials requested by an external entity. Technical knowledge helps users recognise and prove an external entity's identity based on given authentication information, while contextual knowledge help users to decide whether the external entity is entitled to request user's authentication credentials. Previous literature (e.g. [3]) have addressed only the user's need for technical knowledge, but contextual knowledge is equally important [4].

The technical knowledge required depends on the authentication communication channel. Users need knowledge to recognise external entity's identifier and understand associated authentication credentials. The knowledge must suffice to avoid falling victim to sophisticated spoofing techniques. The set of entities that are entitled to request the authentication credentials (AC) is obvious when the entity that requested the AC is the entity that users have shared ACs with. However, it would be ambiguous if the legitimate entity has delegated or shared the right to request the AC to other entities. Users need knowledge about how to distinguish (more than recognise) the set of entities that have been delegated from others. A typical example is an online shopping payment system. There is no clearly defined set of entities that could accept credit card details: online mechants, or some shops' own processing systems can all request card details. To steal users' financial credentials phishers could simply appear as a trustworthy online shop with its own payment processing system.

When an entity has a large number of identifiers users must know how to determine whether the identifier he/she currently sees is legitimate. Authentication on the telephone is an typical example of this. If a company has many telephone numbers users will have difficulty recognising whether the caller's number is one of those.

During the threat modelling practice, analysts should document the all expected knowledge from users. As those expected knowledge may change in the future especially those contextual knowledge. When those changes do happen, analyst can quickly identify the possible vulnerabilities by identifying the emerging knowledge gap.

Assumptions. The security of EE-to-user authentication assumes that users perform certain required actions correctly and consistently. System designers need to know how plausible such assumptions are. Results of existing empirical studies may prove useful [3, 5, 8, 11, 14, 12, 9] or further user studies may be carried out. If asssumptions prove implausible the system design must be altered. Users' behaviours are are affected by how systems are designed, education users receive, etc. As a result the legitimacy of assumptions on users is not static. Threat modelling analysts should document the user assumptions designers make for each entry point and periodically revalidate them. Invalid assumptions are clearly vulnerabilities.

4.4 External Entity Authentication in Communication Matters

Many active attacks lure victims by first impersonating the EEs in communication, such as masquerading as legitimate entities to send emails to users. The

trust, expectation and perception constructed in communications could reduce users' ability to authenticate the EE in the following authentication session [4]. As a result, it is important to study vulnerabilities within the communication between legitimate entities and users.

The method used to analyse the vulnerabilities at entry points can be applied to analyse the vulnerabilities in communication. Here, there is more contextual knowledge users need to be aware of: 1) What are the communication channels the external entity would choose to initiate communication with users, if any? 2) In which circumstances the external entity would initiate communication with its users?

5 Case Study

We illustrate elements of the approach with reference to OpenID as a case study. OpenID is a decentralised, free and shared identity service, which allows Internet users to log on to many different web sites using a single digital identity, eliminating the need for a different user name and password for each site. It is increasingly gaining adoption among large sites, with organisations like AOL, BBC, Google, IBM, Microsoft, Orange, VeriSign, Yandex and Yahoo! acting as providers. We apply our method to analyse the default OpenID solution provided by VeriSign.

5.1 Passive Attack Analysis

Properties of authentication credentials. The complete set of user authentication credentials in this system is : {user name, password, access to a secondary email account} The property of the users' authentication credentials are listed in Table 1. Both authentication credentials are losable, but as long as not both credentials have been lost, their values can be recovered or reset.

Table 1. User Authentication Credential Properties

Authentication Credential	Mode	Factor	Assigned By	Losable
Password	Primary	Users know	User	True
Access to a chosen email account	Emergency	Users possess	User	True

Authentication credentials vulnerable to passive attacks. The password exposure level is uncertain, because it is user-assigned and there is no mechanism to ensure strong password choices. The exposure level for the access right to the email account is medium, because apart from the users and the system, the email service providers can also access the email messages in the email account.

Two sets of authentication credentials can be used to prove a user's identity, and each set has only one member: {password}, and {access to a chosen email account}. The exposure level for the password set is uncertain. This introduces a vulnerability, because the system cannot influence whether users choose weak

passwords or reuse their passwords. Insiders are likely to be able to guess the password if it has been chosen poorly. The exposure for the second set medium, as its only member has medium exposure level. The parties to which the credential is exposed are limited, and it would be safe from general passive attacks.

Authentication security dependency graph. The authentication credential – access to a chosen email account – is in the category of what users have access to, so it has created the dependency relationships between VeriSign's OpenID authentication solution and the email providers which users have chosen. This relationship is unpredictable from the analysts' point of view, because there is no way to predict which email providers users would choose. Even worse, the access to the email account alone can complete the recovery and reset of the password.

According to our method, this design has at least two vulnerabilities: have created uncertain dependency relationships; and the system does not try to minimise the relationship by asking users to provide more authentication credentials together with the access to the chosen email account.

5.2 Active Attacks Analysis

Sensitivity of the authentication credentials. Both the password and the access right to the email account have high level of sensitivity.

Impersonating targets. The user has shared its authentication credential with VeriSign. The entities that are entitled to initiate the user authentication are not well defined and there is no mechanism for users to effortlessly and accurately know whether the entity that requests the use of OpenID authentication is legitimate or malicious. The email providers that existed in the dependency graph could also be the targets of impersonating. This is a vulnerability, as designers of VeriSign cannot patch or eliminate the vulnerabilities that existed in the EE-to-User authentication in the email systems.

Lifecycle of authentication credentials & entry points. Among all the possible targets, the only one designers can influence is VeriSign's authentication system. The lifecycle for the user's authentication credentials (including the communication channels) are shown in figure 2. There are three possible phishing attack entry points: 1) Synchronisation State; 2) Operation State; 3) State transition from operation to assignment;

5.3 Vulnerabilities at Each Entry Point

The methods and processes at each entry point analysis are the same, so for demonstration purposes, we explain only how the analysis is carried out for the operation state.

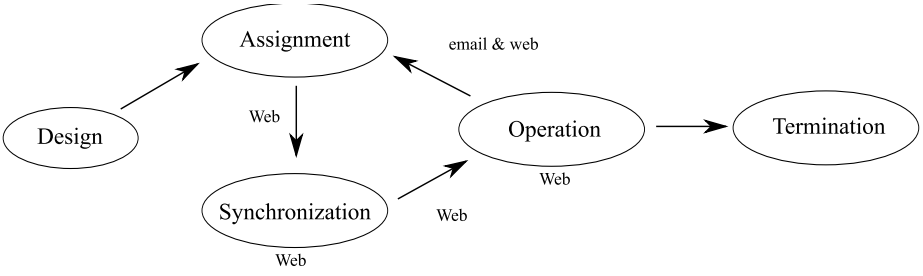

Fig. 2. The Lifecycle of Authentication Credentials

Operation state. In this state, the user first visits the service provider website and requests to sign in with his/her OpenID user ID. Assuming the user ID belongs to VeriSign, the user is directed to VeriSign's website, and is asked to enter the correct user name and password.

Reliability and Sufficiency of External Entity Authentication Credentials. In the operation state, all authentication actions are carried through the web page communication channel. VeriSign identifies itself on web by its domain name and URL. Its URL is `https://pip.verisignlabs.com/`. The compromise of the client side agents (operating system, web browser, networking components) and infrastructure agents could make the domain names displayed on the web browser no longer trustable. Among those agents, the client side agents are most vulnerable, least protected and exposed to the Internet. So its compromise is very likely, and the domain name alone is not reliable enough to identify the entity. VeriSign has used an SSL/TSL certificate to prove that the domain name is genuine. As a result, we can consider that VeriSign has provided sufficient and reliable authentication credentials.

Knowledge. To decide whether a URL belongs to VeriSign or any other entities (especially when some URLs are made to look as if they come from VeriSign) users must understand the syntax of URLs. Users also need knowledge to understand the SSL/TSL certificate to identify which entity really owns this domain. Both sets of knowledge are not possessed by many users [3]. This could be considered as potential vulnerability.

As for the contextual knowledge, it is not clear which websites are entitled to request users to use OpenID authentication. As a result, attackers could set up a phishing website which looks identical to VeriSign's and then lead users to this phishing website to steal the authentication credentials.

assumption. It is assumed that users check the URL and SSL certificate when they are in this state. Based on previous studies, we know a lot of users pay no attention to them. Users are unlikely to pay attention to the content of the SSL certificate, and they care only about their existence. [3, 8, 9, 11, 14]. As a result, the assumption on users has weak validity and vulnerability has been created.

5.4 Vulnerabilities in Communication

VeriSign uses only emails to communicate with its users. We will apply the same method that we used to analyse the entry point to analyse the vulnerability within the external entity authentication in the email system VeriSign uses.

Reliability and sufficiency of external entity authentication credentials. An email is identified by its sender's address. The compromise of client agents and any server agents could make the sender's address untrustable. In fact, the sender's address can be spoofed even without the compromise of any agent, as shown in [8]. So it is extremely unreliable to identify the real sender of the email. VeriSign has not used any other authentication credentials to prove that the email indeed comes from VeriSign, which has created a serious vulnerability and allows attackers to impersonate VeriSign in email communication channels. In conclusion, users have not been given reliable and sufficient authentication information to prove verisign originated the email.

Knowledge. Most users will be able to recognise the sender's email address. Users must have (contextual) knowledge needed of: the email address used by VeriSign to communicate with its users; and under which circumstances VeriSign would contact its users. VeriSign has not made clear to its users which email address it will use to communicate with users. As a result, email addresses whose semantic meaning is close to the email address VeriSign really uses could be accepted by users.

The last one is unclear as well, because VeriSign has not made this explicit to its users. As a result, it gives chances for phishers to create a scenario to lure victims to phishing websites.

Assumptions. It is assumed that users will check the email sender's address. This assumption is realistic and it is helped by the user interface design that users automatically read the sender and titles first.

6 Conclusions

User–side threat modelling is as important as system–side threat modelling, but it is much less well studied. This paper describes a method to systematically identity threats to web user authentication from user and social perspectives. Besides the VeriSign OpenID solution we have also used this method to identify threats to other user authentication systems: the UK national grid system, and Google websites. However, our method should not be viewed as complete; it is our initial effort towards developing a threat modelling method that can be used by system designers with moderate security knowledge. In future we will further refine this method and evaluate its usability by system designers. The provision of analysis tools for investigating threats to the user is important and we recommend the area to the research community.

References

1. Anti-phishing work group home page (2007), http://www.antiphishing.org/
2. Flinn, S., Lumsden, J.: User perceptions of privacy and security on the web. In: The Third Annual Conference on Privacy, Security and Trust (PST 2005), St. Andrews, New Brunswick, Canada, October 12-14 (2005)
3. Dhamija, R., Tygar, D., Hearst, M.: Why phishing works. In: CHI 2006: Proceedings of the SIGCHI conference on Human Factors in computing systems, ACM Special Interest Group on Computer-Human Interaction, pp. 581–590 (2006)
4. Dong, X., Clark, J.A., Jacob, J.: A user-phishing interaction model. In: Conference on Human System Interaction (2008)
5. Downs, J.S., Holbrook, M.B., Cranor, L.F.: Decision strategies and susceptibility to phishing. In: SOUPS 2006: Proceedings of the second symposium on Usable privacy and security, pp. 79–90. ACM Press, New York (2006)
6. Florencio, D., Herley, C.: A large-scale study of web password habits. In: WWW 2007: Proceedings of the 16th international conference on World Wide Web, pp. 657–666. ACM Press, New York (2007)
7. Friedman, B., Hurley, W.D., Howe, D.C., Nissenbaum, H., Felten, E.W.: Users' conceptions of risks and harms on the web: a comparative study. In: CHI Extended Abstracts, pp. 614–615 (2002)
8. Jagatic, T., Johnson, N., Jakobsson, M., Menczer, F.: Social phishing. ACM Communication (October 2007)
9. Jakobsson, M., Tsow, A., Shah, A., Blevis, E., Lim, Y.-K.: What instills trust? a qualitative study of phishing. In: USEC 2007 (2007) (Extended abstract)
10. Nikander, P., Karvonen, K.: Users and trust in cyberspace, pp. 24–35 (2001)
11. Schechter, S., Dhamija, R., Ozment, A., Fischer, I.: The emperor's new security indicators: An evaluation of website authentication and the effect of role playing on usability studies. In: 2007 IEEE Symposium on Security and Privacy (2007)
12. Whalen, T., Inkpen, K.M.: Gathering evidence: use of visual security cues in web browsers. In: GI 2005: Proceedings of Graphics Interface 2005, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, pp. 137–144. Canadian Human-Computer Communications Society (2005)
13. Wikipedia. Phishing. web, http://en.wikipedia.org/wiki/Phishing
14. Wu, M., Miller, R.C., Garfinkel, S.L.: Do security toolbars actually prevent phishing attacks? In: CHI 2006: Proceedings of the SIGCHI conference on Human Factors in computing systems, pp. 601–610. ACM Press, New York (2006)