Thesis Proposal: Fighting Phishing at the User Interface

Min Wu

Chapter 1. Introduction

As people increasingly rely on Internet to do business, Internet fraud becomes a greater and greater threat to people's Internet life. Internet fraud uses misleading messages online to deceive human users into forming a wrong belief and then to force them to take dangerous actions to compromise their or other people's welfare. The main type of Internet fraud is phishing. Phishing uses emails and websites, which designed to look like emails and websites from legitimate organizations, to deceive users into disclosing their personal or financial information. The hostile party can then use this information for criminal purposes, such as identity theft and fraud. Users can be tricked into disclosing their information either by providing sensitive information via a web form or downloading and installing hostile codes, which search users' computers or monitoring users' online activities in order to get information.

Phishing has a high increasing rate and is effective to fool users. In May 2004, Anti-Phishing Working Group (APWG http://www.antiphishing.org/) stated that reports of email fraud and phishing attacks increased by 180% in April 2004 and up 4,000% since November 2003. [APWG 2004-05-24] According to a recent study done by Anti-spam firm MailFrontier Inc, phishing emails fooled users 28% percent of the time [Sullivan 2004].

A real attack

Let us look at a real phishing attack first. APWG archives examples of phishing attacks. One example describes a recent attack against the eBay customers, reported on March 9, 2004 [APWG 2004-03-09]. Ebay Inc. is the world's online market for Internet users to sell and buy or auction and bid their items. The victim first received an email that informed her that her account information stored in eBay database was invalid and she should click the embedded link in the email to an eBay web page to update her account information. The link led the victim to a faked eBay web page that asked her for her credit card, contact information, SSN, and eBay account information. Figure 1 shows the phishing strategy, which contains two steps:

1. In the first step, a phishing email, as shown in figure 2, looked like a legitimate email from eBay. The sender (From: S-Harbor@eBay.com) had the domain name as ebay.com, which is the legitimate domain for eBay Inc, registered ten years ago in California, USA. The email subject and content presented a convincing request that asked users to provide information in order to reactivate their eBay account. The embedded link, visible as https://cgi1.ebay.com/aw-cgi/ebayISAPI.dll? looked legitimate too. It used SSL as an encrypted communication channel and the visible URL was from eBay web site. Based on the presentation the user formed a mental model that an eBay notification asked her to update her account at the eBay web site. The user then performed an action by clicking the embedded link, which she thought to direct her to the eBay web site. The user's action was then translated to a system operation of retrieving a web page of http://210.93.131.250/my/index.htm from the server with the IP address of 210.93.131.250, a server from a communication company registered in Seoul, South Korea. This company has no relationship with eBay Inc.
2. In the second step, the phishing web page, as shown in figure 3, looked like a legitimate eBay web page. The web page contains an eBay logo. The content and layout matches the format of the pages from the eBay web site. The input labels tell users what data they need to provide. Based on the presentation, the user formed a mental model that she was at the eBay web site and she need to provide the information about her credit card, contact information, SSN and her eBay account. The user then performed the actions by typing her personal and financial data and clicking the submit button. The user's action was again translated to a system operation of transferring a string of bits, which encoded the user's personal and financial data, to a server with the IP address of 210.93.131.250, but not a legitimate eBay server.

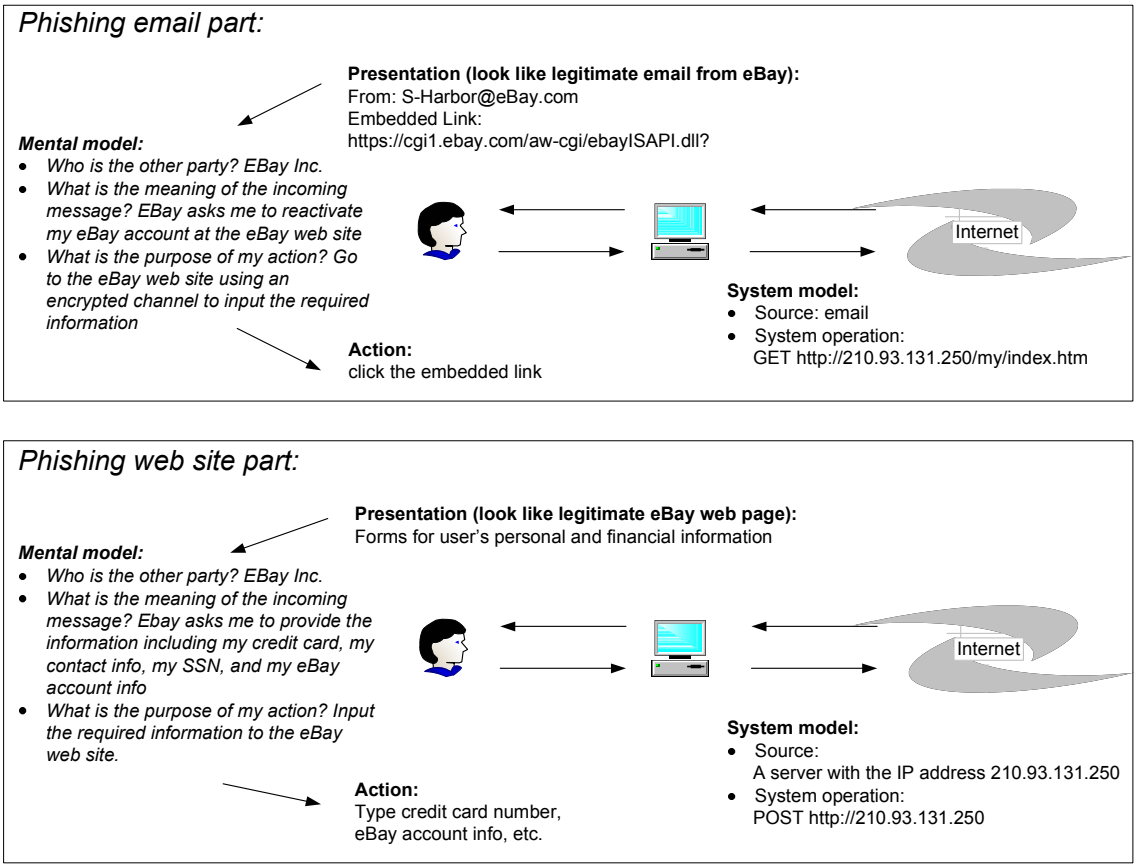


Figure 1: A real phishing attack

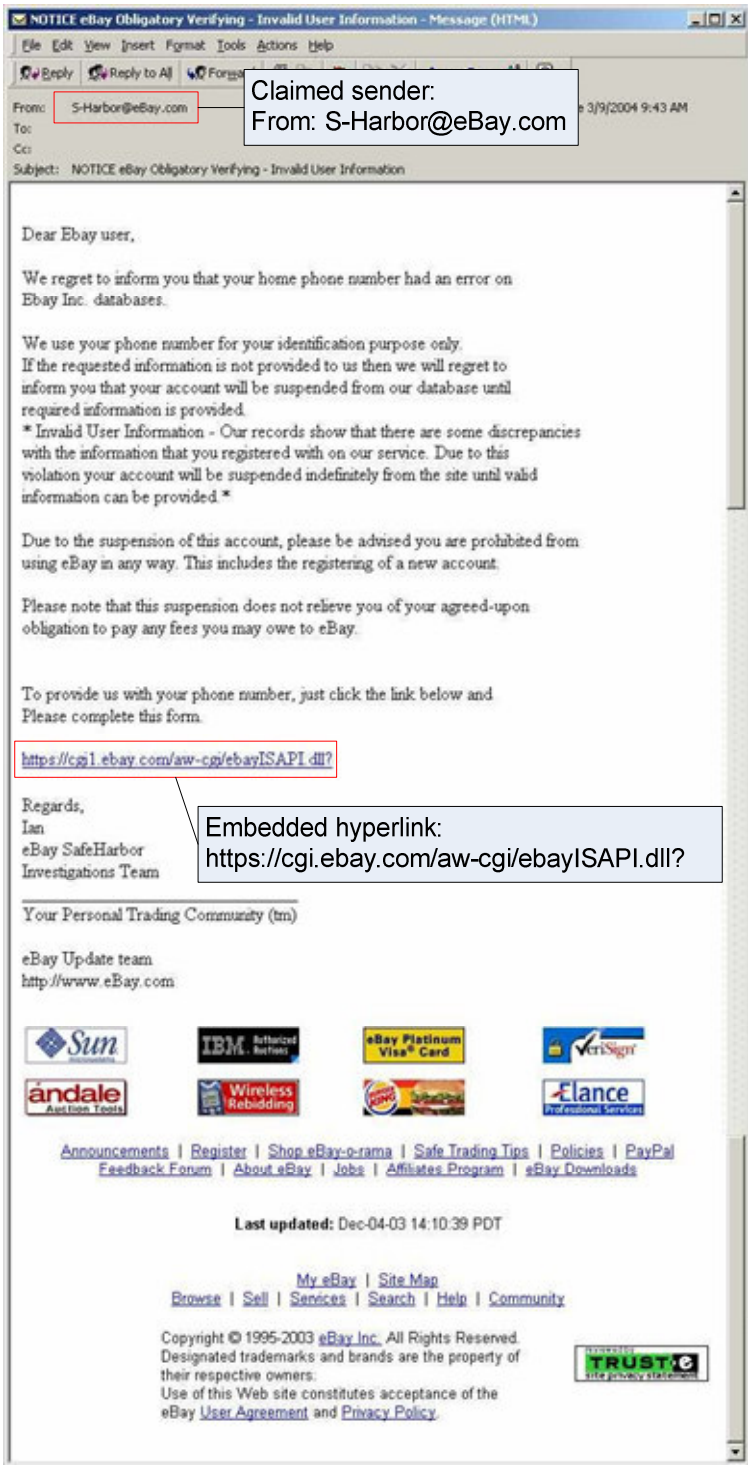


Figure 2: Screenshot of the phishing email (source: APWG)

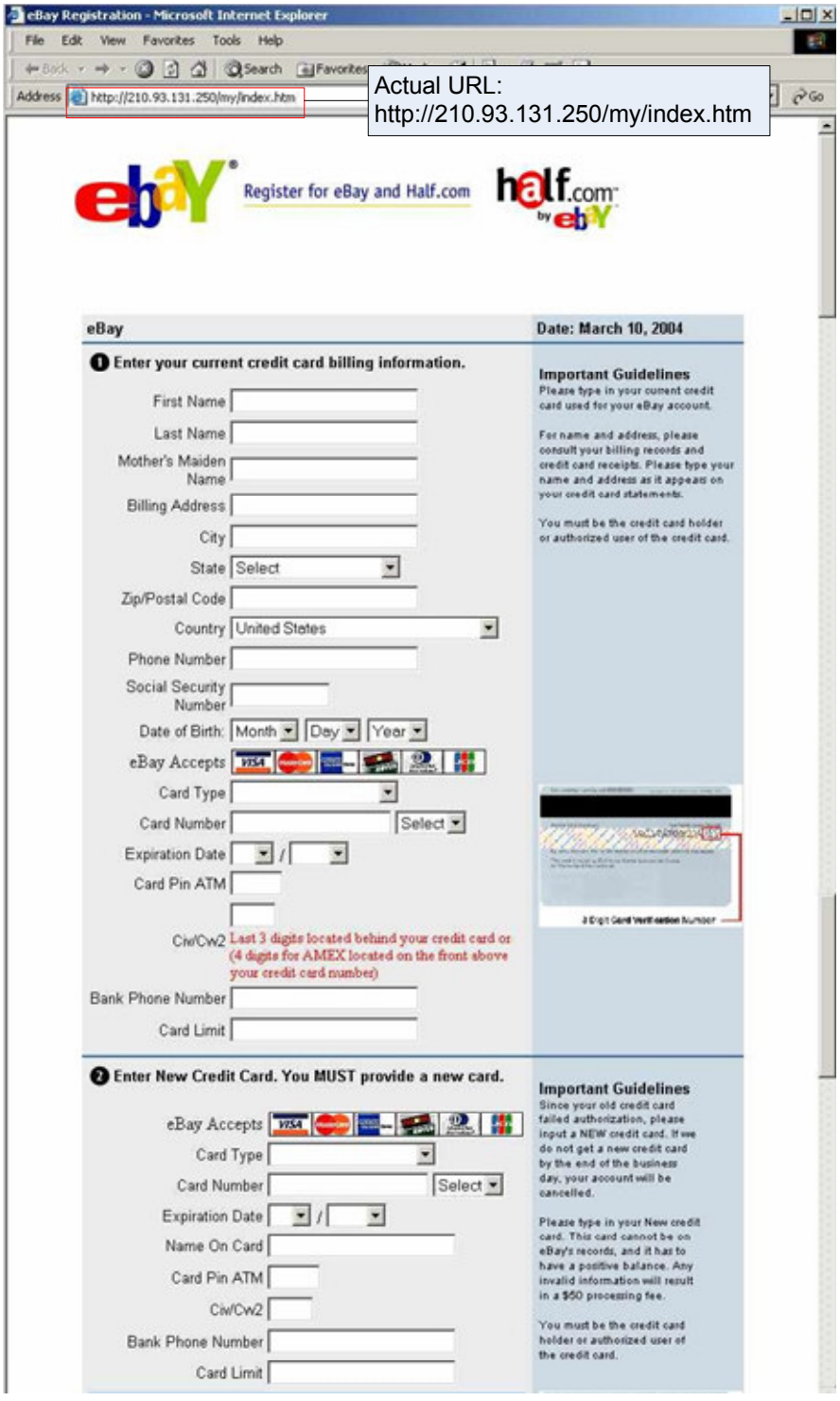


Figure 3: Screenshot of the phishing web page (source: APWG)

Analysis

Phishing is a semantic attack. Bruce Schneier once said that "the third wave of network attacks is semantic attacks: attacks that target the way we, as humans, assign meaning to content." [Schneier 2000] Successful phishing depends on a discrepancy between the way a user perceives a communication, like an email message or a web page, and the actual effect of the communication. Figure 4 shows the structure of a

typical Internet communication, dividing it into two parts. The system model is concerned with how computers exchange bits—protocols, representations, and software. When human users play a role in the communication, however, understanding and protecting the system model is not enough, since the real message communicated depends not on the bits exchanged but on the semantic meanings that are derived from the bits. This semantic layer is the user's mental model. The effectiveness of phishing indicates that human users do not always assign the proper semantic meaning to their online interactions.

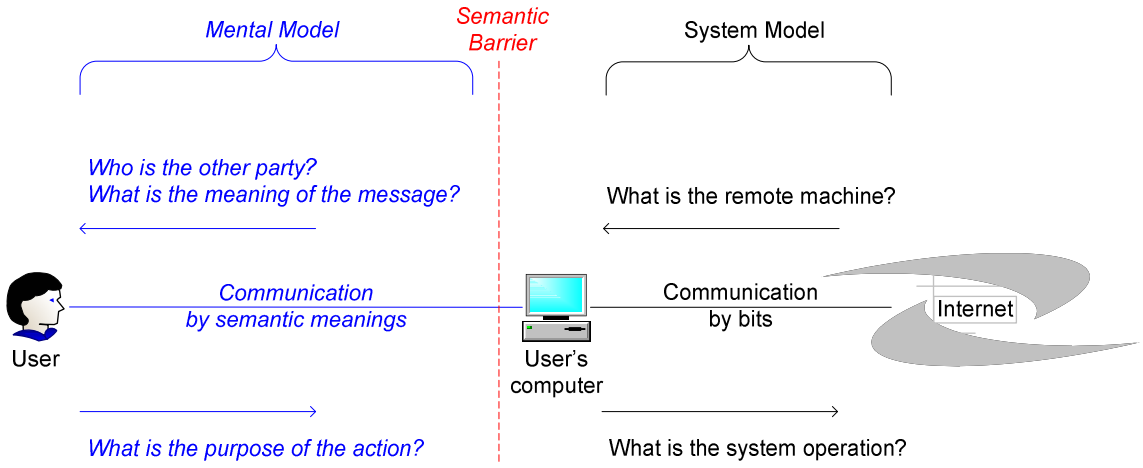


Figure 4: Two models in the human-Internet interaction

When a user faces a phishing attack, the user's mental model about the interaction disagrees with the system model. For example, the user's intention may be "go to eBay", but the actual implementation of the hyperlink may be "go to a server in South Korea." It is this discrepancy that enables the attack, and it is this discrepancy that makes phishing attacks very hard to defend against. Users derive their mental models of the interaction from the presentation of the interaction—the way it appears on the screen. The implementation details of web pages and email messages are hidden, and generally inaccessible to most users. Thus the user is in no position to compare their mental model with the system model, and it would take extra effort to do so. On the other hand, email clients and web browsers follow the coded instructions provided to them in the message, but are unable to check the user's intentions. Without awareness of both models, neither the user nor the computer is able to detect the discrepancy introduced by phishing.

One extreme solution to the phishing problem would simply discard the presentation part of an Internet communication–the part that produces the user's mental model—since it can't be trusted. Instead, a new presentation would be generated directly from the implementation. If the user's computer is trustworthy, then, the presentation seen by the user would be guaranteed to be related to the actual implementation. Unfortunately, the cost of this idea in both usability and functionality would be enormous. Most online messages are legitimate, after all, with the presentation correctly reflecting the implementation. Phishing messages are rare (but pernicious) exceptions. So this solution would improperly sacrifice the freedom of legitimate senders to present and brand themselves in order to block a small number of wrongdoers.

So we must accept the fact that users will see messages with mismatched presentation and implementation. Attempts to fight phishing computationally try to enable the computer to bridge the gap between the user's mental model and the true system model. But the human user must be the final decision maker about whether a message is phishing or not. The reason is that phishing targets how users assign semantic meaning to their online interactions, and this assignment process is outside the system's control.

Strategy

Figure 5 shows how human users interact with the Internet. We separate the interaction into four steps:

1. In the message retrieval step, a message arrives at a user's personal computer from the Internet.
2. In the presentation step, the user perceives the message displayed in the user interface and forms the mental model.

3. In the action step, the user performs an action to the user interface guided by her mental model.
4. In the system operation step, the user's action is translated to system operations

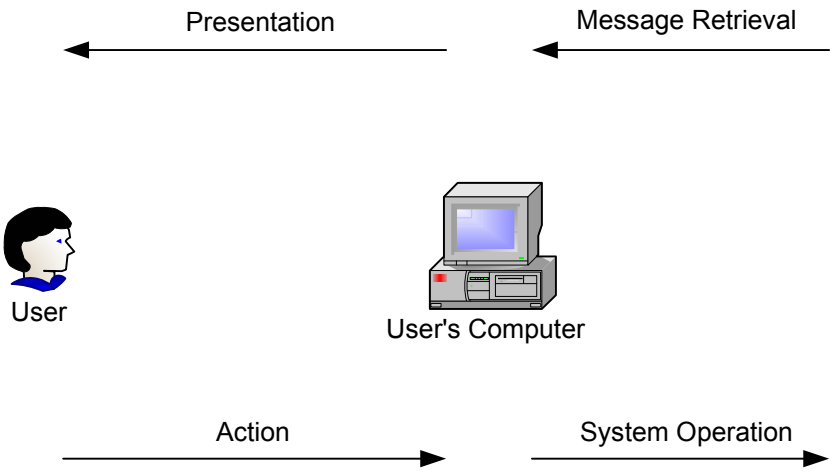


Figure 5: Human-Internet interaction

In order to bridge the gap between user's mental model and the system model during the online interaction, the system can help from two aspects. Since perception is the main way for users to form the mental model, the system can reflect the system model at the user interface through visualization so that the user's mental model can be affected by the visual cues of the system model so that the gaps between them can be shrunk. On the other hand, the system can try to derive the user's mental model and compare it with the system model already available to it in order to detect the gap. After a dangerous gap is detected, the system can either use visualization method to change the user model or generate new and safe system mechanism that matches the user's mental model.

Let us find out during the 4-step interaction procedure how to form the system model and the mental model and how to bridge the dangerous gap between them caused by phishing messages.

At the message retrieval step, the system should get the information about where the message actually comes from. Phishing, also as a social engineering attack, first targets on the process of how human users assign semantic meanings to the other party's identity. As Kevin Mitnick pointed out [Mitnick & Simon 2002], social engineering is getting people to do things they wouldn't ordinarily do for a stranger by pretending to be someone else and just asking for it. When human users interact with the Internet, they do not think that they are communicating with a specific machine attached to the Internet. Instead, they automatically assign semantic meanings to the machine that they are talking to. The system should help users to form the correct identity of the other party and derive how trustworthy it is. It can then present this information of trustworthiness through the visual cues at the user interface. In this way, the system information, such as the IP address of the web server, is assigned a degree of trustworthiness, whose visualization can affect the user's mental model since the users mainly identify the source of the online message through its appearance. The trustworthiness of a web server can be decided either by reputation of the server or the user's personal online history. The reputation of a server can be derived by a variety of ways, such as the domain registration date from the WHOIS data, the number of links from other servers as references, the SSL certificate from well known CA's, and the reliability report from authorities. The trustworthiness is also decided by user's online history: the more the user interacts with a site, the more trustworthy that site is to the user. Since one main characteristic of phishing messages is that the messages claim to be from reputed legitimate organizations but actually the actual web server is not trustworthy, we expect that integrating the source information with the message presentation can help users to detect possible phishing messages since users can easily notice that the claimed source is not same as the actual source.

The message is shown to the user at the presentation step. The information about identity, reputation and trustworthiness of a source derived from last step is not reliable enough block phishing at the system level. The human users have to decide by themselves if a message is trustworthy or not when they see the message. As Chaiken pointed out, heuristic strategy, by which people make decision of trustworthiness based on only the most obvious or apparent information, is identified as one of the two processing strategies by which an evaluation of trustworthiness may be made [Chaiken 1980]. Therefore, making the system-derived trustworthiness of the source as an obvious and understandable visual cue at the user interface can help users to make correct evaluation. We propose the content-integrated visual cues so that the security information will not be neglected by users since it is integrated naturally with the user's primary task but on the other hand will not block the user's primary task either. Moreover, trust has been broadly categorized into three layers: dispositional trust, the psychological disposition or personality trait of an agent to be trusting or not; learned trust, an agent's general tendency to trust, or not trust, another agent as a result of experience; and situational trust, in which basic tendencies are adjusted in response to situational cues, such as the amount and quality of communication, and therefore although one may trust an agent on the whole one may not do so in certain situations and under certain circumstances. These trusted layers may be seen to operate such that, in the absence of learned trust, for example, an agent's dispositional trust influences his behaviors, and where learned trust exists, then an agent's dispositional trust in less important in determining his behavior. Thus it follows that learned trust may be regarded as the experience born of a collection of past situational trusts. [Marsh & Dibben 2003] Phishing attacks exploit the learned trust by simulate the situations as the user's past experience. The phishing messages claim that they are from the organizations that users have contacted and formed trust before. Therefore, we propose the perception analysis so that the system can get more accurate information about the massage's claimed organization. The fundamental idea of perception analysis is to analyze the message's screenshot from the user's point of view using image analysis and visual clustering methods to group together the web pages that appear to be from the same organization or to be in the same type (e.g., a login page or a registration page). The system can categorize the incoming messages into a certain cluster according to its appearance. If the claimed source derived by the perception analysis is not related to the actual source at the semantic level, i.e., in the same organization or in the related organizations, the message is probably a phishing message and users are warned at the user interface.

The user performs some actions at the action step. It is the step for the system to derive the user's intention about his actions. The fundamental characteristic of phishing messages is that they deceive the user to perform some dangerous actions, which the user expects to be normal and legal. Therefore, deriving user's intention in this step can effectively detect phishing messages. The action exploited by phishing attacks most is submitting sensitive information, either personal or financial, to a hostile party. The user's intention can be simplified into two pieces of information: what are the data types and where are the data supposed to be sent. The data types can be derived either by recognizing the labels of the input fields using optical character recognition technique or by monitoring user's keyboard typing when he fills the form. The data submission destination can be either derived implicitly by perception analysis of the whole message and the form part or requested explicitly from the human users through keyword typing by asking them where they expect their input data are sent through a popup text field. After the user's intention is derived, the system can either find out if the user is deceived by the message into forming such intention or provide optional safe system operations for users to take in order to fulfill their intention.

At the system operation step, the user's intention is translated into some system operation. If user triggers the given operation coded in the message, the derived user intention should act as a "semantic sandbox" to guide the system operation. Any detected dangers should be warned to the user.

We expect that this chain of detection and visualization mechanisms can accurately prevent users from deceived by phishing messages. On the other hand, we do not want our anti-phishing scheme to disrupt user's normal online activities. We propose to use different warning levels to display different phishing detection result. Since phishing attacks are at the semantic level and need user's involvement, the more the user interacts with the message, the more accurate the phishing detection is. Therefore, the overall trustworthiness of a message derived at the message retrieval step should be displayed less aggressive (e.g., the suspicious pages looks darker or fuzzier than trusted pages, with the "Oh yeah?" style button [Tim Berners-Lee 1997] with detailed explanation) than the discrepancy between user's intention and the system implementation detected at the action or system operation steps (e.g., content-integrated warning or modal confirming dialog box). Moreover, the system's incorrect decision should be easily overridden by users,

either implicitly or explicitly. If a user continues his task without affected by the warning, the system may interpret this case as the user discards the security advice because it is not meaningful to him and therefore when the user gets the same message later, the warning style should be decayed in terms of aggressiveness. Moreover, there should be an easy way for users to directly assign misclassified trusted sources into a "safe zone" to avoid later aggressive analysis and annotation.

The anti-phishing detection and visualization chain will be evaluated in two aspects. This chain should effectively prevent users from being deceived by real phishing messages. We can evaluate the effectiveness through the controlled user study and the field study using both the reconstructed existing phishing attacks from APWG archives and the new phishing attacks with novel tricks invented by us. On the other hand, this chain should not disrupt user's normal online activities. We can evaluation the degree of bothering through measuring the error rate of the detection methods on legitimate but un-reputed messages, the rate of distraction by the visual cues from the user's primary task, and the user's subjective satisfaction.

Thesis statement

My thesis is: The user interface can bridge the gap between the user's mental model and the system model of the human-Internet interaction by a chain of detection and visualization methods at the semantic level, providing accurate and effective defenses against phishing without disrupting the normal interaction.

Chapter 2. Related Work

Let's survey the security solutions to prevent Internet fraud according to the 4-step interaction procedure shown in figure 5.

Message retrieval

One possible solution for phishing is to block all phishing messages from showing to users. The requirement for this solution is that the computer alone can accurately differentiate phishing messages from legitimate ones. Computer-based filtering depends on message's properties that are understood by computers:

- Identity:
    - Black list is widely used to block potentially dangerous or unwelcome messages, such as spams. If the email sender is in user's spam black list, the email is categorized as spam or even dropped without user's notice. IE6.0's Content Advisor, as shown in figure 6, allows the user to create a list of web sites that are always viewable or never viewable. Company's firewall also restricts the employees to access only a subset of the Internet. In term of anti-phishing, EarthLink Toolbar [Earthlink] alerts users about web pages that are on a black list of known fraudulent sites. However, the source-based filtering is ineffective since it is easy to generate new identities in the Internet. Without constantly updating the list, the black-list filtering cannot warn users about dangerous messages from newly-setup sources. For example, figure 7 shows that the EarthLink toolbar fails to block a newly-reported phishing site [APWG 2004-08-06] from APWG.
    - White list allows users only access messages from a list of accepted sources. For example, Secure Browser [Tropsoft] controls where users browse on the Internet by using a list of keywords and full URLs. White list avoids the new identity problem since newly-setup sources are initially marked as unacceptable. However, there is a problem of how to define the white list. Since it is impossible to predict where users will browse beforehand, predefined static white list prohibits users from accessing legitimate websites but not on the list. On the other hand, dynamic list that needs user's involvement adds burdens to users since for every site that they first browse they have to decide if to put it into the white list or not. Moreover, if a phishing site can convince users to submit sensitive data to it, it can also convince them to put it into the white list.
- Content: Content analysis is widely used in anti-spam and anti-virus solutions. However, since such analysis mainly depends on heuristic rules, it is only good at catching well-known patterns, such as spam key words and virus code signatures. In order to beat content analysis, an attacker can tweak the content to bypass the well-known filtering rules. For example, encryption and compression are added to existing viruses in order to bypass anti-virus scans [F-secure 2004] and random characters are inserted into the emails in order to bypass spam filters. In term of anti-phishing, a sophisticated attack used images to display text messages in order to defeat text analysis [APWG 2004-02-24].
- Type: The types of a message can also affect user's access. For example, mail client applications always filter out the attachment which is executable. As another example, all common browsers have security settings to control the mobile code execution, such as ActiveX, JavaScript, and Java Applet. However, the message type alone cannot be treated as dangerous or not. It should be evaluated by combining with the information about who publishes the message (identity analysis) and what the message is about (content analysis).

Thompson et al. proposes a set of approaches to detect misinformation by analyzing the incoming messages [Cybenko & Giani & Heckman & Thompson]. For example, the reliability of an incoming message can be determined by the trust rating of the message source, by the reliability rating of the message content through collaborations with other redundant information sources, or by the linguistic analysis of the message in order to see if the message belongs to the claimed author.

Figure 6: Black/White List of Web Sites

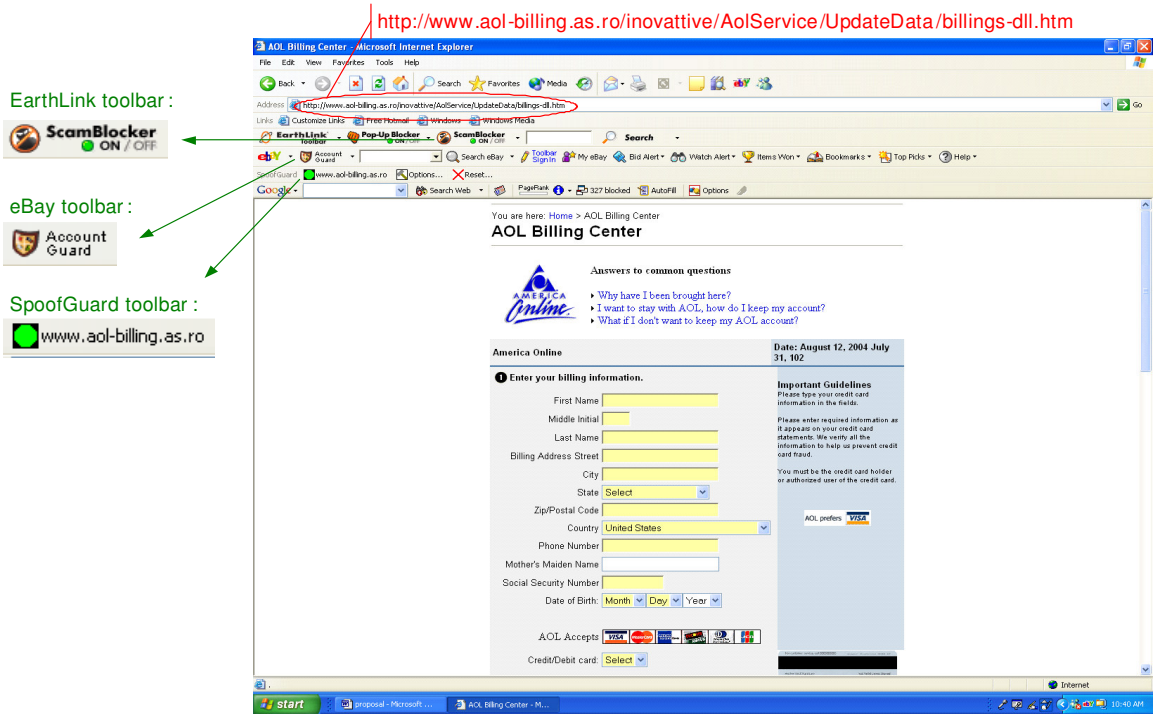


Figure 7: Third-party toolbars fail to block a newly-setup phishing site (tested on 08/12/2004)

Presentation

When a message is presented at the user interface, the user interface helps users decide if the current message is potentially dangerous or not through different visual cues.

Current browsers reflect the status information through a set of visual cues, for example, the address bar displays the URL of the retrieved web page and the secure connection icon in the status bar indicates if the current connection is encrypted or not. Mozilla FireFox changes the address bar's background color and adds another lock icon in the address bar in order to visually differentiate HTTPS connections from HTTP

ones, as shown in figure 8. Security tips on phishing ask users to always pay attention to these cues. However such visual cues are ineffective because:

1. The cues are displayed in the peripheral area of the browser and they are independent from the web content. Therefore, when users concentrate on the content, the display is hard to get user's attention and results in inattentional blindness.
2. It is possible to hide or fake all security cues (*e.g.*, address bar, status bar, authentication dialogue box, SSL lock icon, SSL certificate information) by using JavaScript and Java Applet [Tygar & Whitten 1996, Felten & Balfanz & Dean & Wallach 1996, Ye & Yuan & Smith 2002].

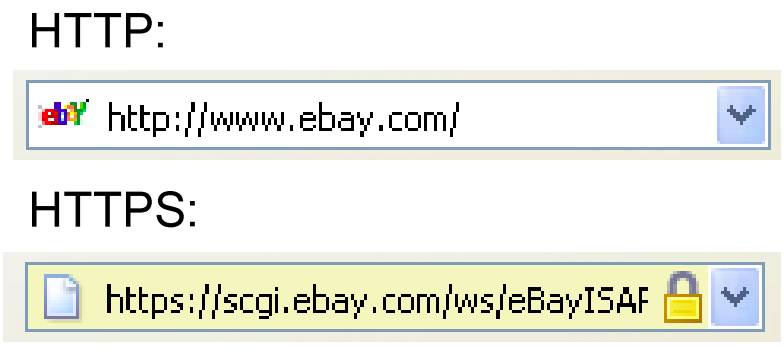


Figure 8: Visual difference between HTTP and HTTPS with Mozilla FireFox address bar

Ebay's Account Guard [Ebay-AG] uses the site indicator in a dedicated toolbar, as shown in figure 9, to show users whether the current page is dangerous or not. It separates the whole Internet into three categories: web sites of eBay or PayPal with green icon, known spoof web sites with red icon, and other sites with gray icon. However, eBay's Account Guard has a couple of problems:

1. The account guard mechanism is not scalable. Phishing attacks are not limited to eBay and PayPal. They target many other financial organizations. It is impossible to cram all the toolbars, each of which represents a single organization, into a single browser.
2. The account guard still uses the peripheral visual cue, an icon in the toolbar, to differentiate eBay or PayPal sites from other sites. It is still questionable if the cue is effective.
3. When a browser connects to a known spoof site, the account guard not only changes its toolbar indicator to red but also pops up a modal alert box. Users cannot get to the main browser without consenting to the alert box. Such design tries to ensure that users see the warning, but it is problematic:
    a. It does not effectively integrate the warning with the content being warned. The alert box in figure 9 contains a sentence "If the site is claimed to be eBay or PayPal and requesting your account information, do not provide it." But the actual web content is blocked by the warning. The users have to remember the warning in order to check if the site is claimed to be eBay or PayPal and if the site requests their account information.
    b. The alert box in figure 9 has a "Report This Site" button for users to report suspicious web sites. However, users have to decide whether or not to click that button or not before actually checking if the web content is suspicious.
4. The account guard fails to detect some phishing sites, as shown in figure 7.

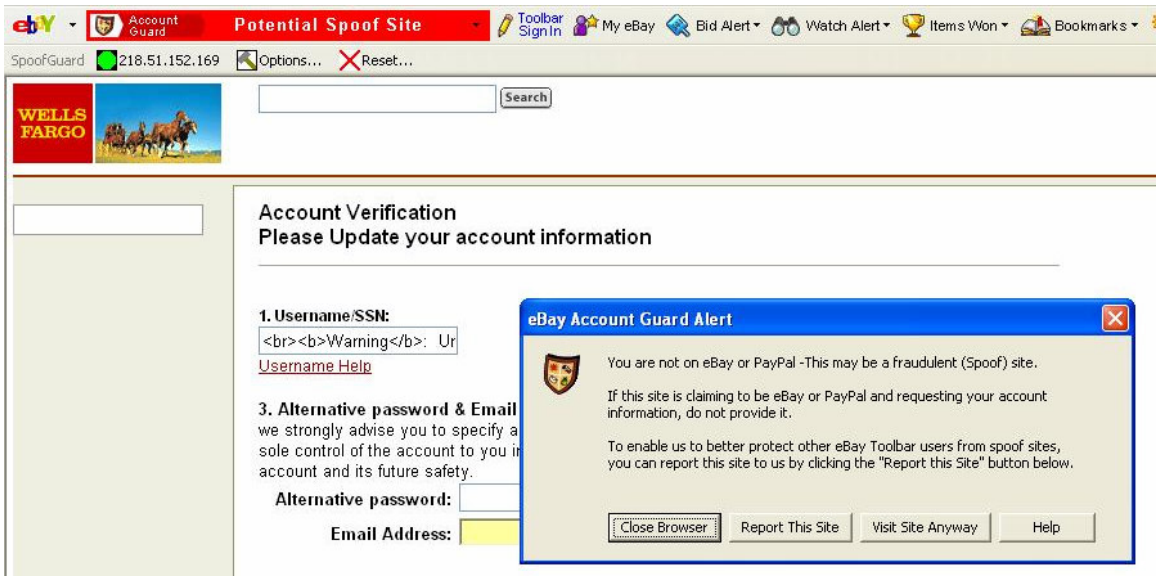


Figure 9: eBay Account Guard

SpoofStick [Spoofstick] is a browser extension that helps users detect URL spoofing by displaying only the most relevant domain information on a dedicated toolbar. For example, SpoofStick will display "You are on 10.19.32.4" when the current URL is http://signin.ebay.com@10.19.32.4/ and display "You are on intl-en.us" when the current URL is http://www.citibank.com.intl-en.us. This toolbar is easier for user to spot and examine than the address bar by using adjustable font size and color, as shown in figure 10. However, SpoofStick cannot solve domain spoofing problem. For example, is ebay-members-security.com a domain for eBay Inc. or is users-paypal.com a domain for PayPal? If the user's answer to either of these questions is yes, they will be tricked even with SpoofStick installed. Moreover, It is unknown if users will get suspicious when the address is in the IP format.

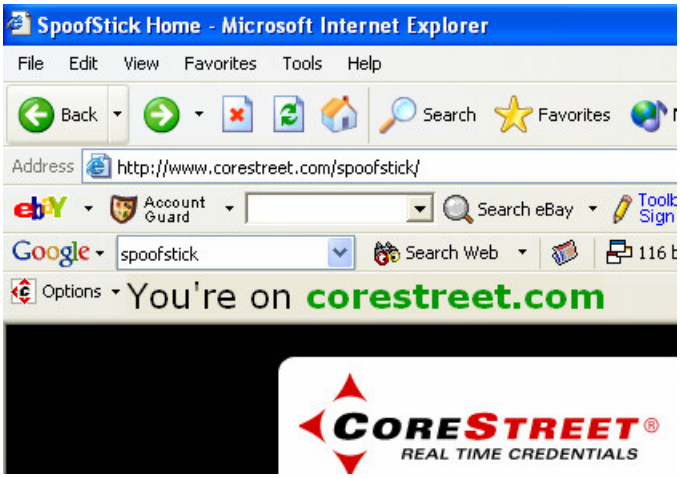


Figure 10: SpoofStick screenshot

TrustBar [Herzberg & Gbara 2004] is another browser plug-in to display SSL information, as shown in figure 11.

Ye and Smith [Ye & Smith 2002] proposed synchronized random dynamic boundaries. All the browser's legitimate windows change their border styles together at random intervals. Since a spoofed window from a remote site cannot get the random value generated by the local machine, its border does not change synchronously with the legitimate window borders. As a result, users are expected to detect spoofed

windows, such as the popup window display SSL certificate information. But the border style display is still the peripheral visual cue.

Personalized display shows users the positive feedback indicating that the current message is legitimate. For example, Amazon's greeting page with the user's registered name indicates that the user accesses the same web site as she registered before. Tygar and Whitten [Tygar & Whitten 1996] proposed to indicate legitimate windows in the local machine using a personal logo. And PassMark Security [Passmark] has introduced a new capability that organizations can embed a personalized image in outgoing emails and on their web site login pages to ensure users that the emails and web pages are authentic. Even security tips suggest using an impersonalized email greeting as a warning sign for spoof emails [Ebay]. However, I am suspicious with the positive visual cues because:

1. It is possible to fake the positive visual cues, which will provide users a wrong sense of security.
2. The absence of cues means dangerous. I expect that it is harder for human users to notice absence than presence.

Action

Since phishing needs user's action, existing security tips discourage users from performing potentially dangerous actions. For example, since most current phishing attacks use emails as a bait in order to trick the recipients into clicking the provided link in the email, which points to a phishing server, security tips [Justice 2003] suggest users to open a new browser and manually type the URL of a legitimate site instead of blindly clicking the given link in emails. However, with the very low probability of a user being phished, this suggestion sacrifices the efficiency of the hyperlinks in most legitimate emails in order to prevent users from clicking misleading links in very few phishing emails. Except users that are paranoid about security, nobody will follow this suggestion. A more aggressive suggestion discourages users from opening any email attachments, which will block the main channel for users to exchange computer files with remote parties.

System operation

In this step, the user's action is translated into the system operation. Since phishing does not exploit the system bugs, even under phishing attacks, the system operations are valid from the system's aspect. For example, it is a valid system operation to post data to a remote machine on the Internet. The evaluation of the safeness in term of phishing is dependent on the semantic meanings assigned to the operation. For example, data submission through HTTP is unhurt if the data is a searching keyword, but is dangerous if the data is credit card number or password. Warning solely based on operations will inevitably generate high rate of false positive errors, that is, warning users for innocent actions, as shown in figure 11. Such false positive errors will eventually cause users to disable the security features.

Figure 11: Warning solely based on system operation

Web password hashing [Boneh & Mitchell & Ross] is a project to solve the password phishing problem. A domain name is hashed by user's original password in order to generate a unique password per site. However, this solution has a couple of problems:

1. Phishing is used not only to steal passwords but also to get other types of data, such as credit card number or bank account information. Although browsers can recognize password input box, it cannot recognize the input for other sensitive data types.

2. Web password hashing assumes that users only type their password into the password HTML element. But we expect that sophisticated phishing web pages can trick users into disclosing their passwords via other channels, such as plain text area.

Case study – SpoofGuard

SpoofGuard is a project from Stanford trying to solve phishing problems at the user side [Chou & Ledesma & Teraguchi & Mitchell 2004]. Let's analyze it based on the four-step structure.

SpoofGuard calculates a message's total spoof score at the message retrieval step. The calculation is based on the common characteristics of the previous detected phishing attacks. For example

- If the URL of the current page contains misleading patterns, such as an @ or has low distance tolerance (the number of inserts or deletes required to change one host name to another, to a list of previously visited host names)?
- If the current page contains images (such as site logo) that look exactly like the images from a number of frequently spoofed domains?
- If the URL of the links in the current page contains misleading patterns?
- If the current page has password input fields but not uses https?

The problem for this calculation is that it is based on the superficial characteristics of current phishing attacks. Such publicly accessible heuristic rules can be bypassed by sophisticated attacks. Methods were proposed in the same paper to fool the password and image checks. Moreover, without deliberately circumventing SpoofGuard's detection rules, some phishing sites are not blocked by SpoofGuard, as shown in figure 7.

Based on the spoof score, SpoofGuard uses a traffic light (red, yellow, and green) in a dedicated toolbar to indicate if the current page is phishing or not. When the score is above a threshold, SpoofGuard pops up a modal warning box that needs user's consent. The problems of such presentation are shown as follows:

- The traffic light is a peripheral visual cue, which may result in inattentional blindness.
- The modal warning box does not integrate warnings with the content that is warned, which sometimes even generates confusions. As shown in figure 12, SpoofGuard warns misleading links without pointing out which links are misleading and explaining why they are misleading. Moreover, also in figure 12, the modal warning box alerts that the page contains password input field. But there is no password input field in the page's visible part. The page needs to be scrolled to the bottom to show the login box with the password input field.
- The heuristic spoof detection generates many false positive results, that is, warnings for legitimate sites, as shown in figure 13, which will interfere user's normal browsing.

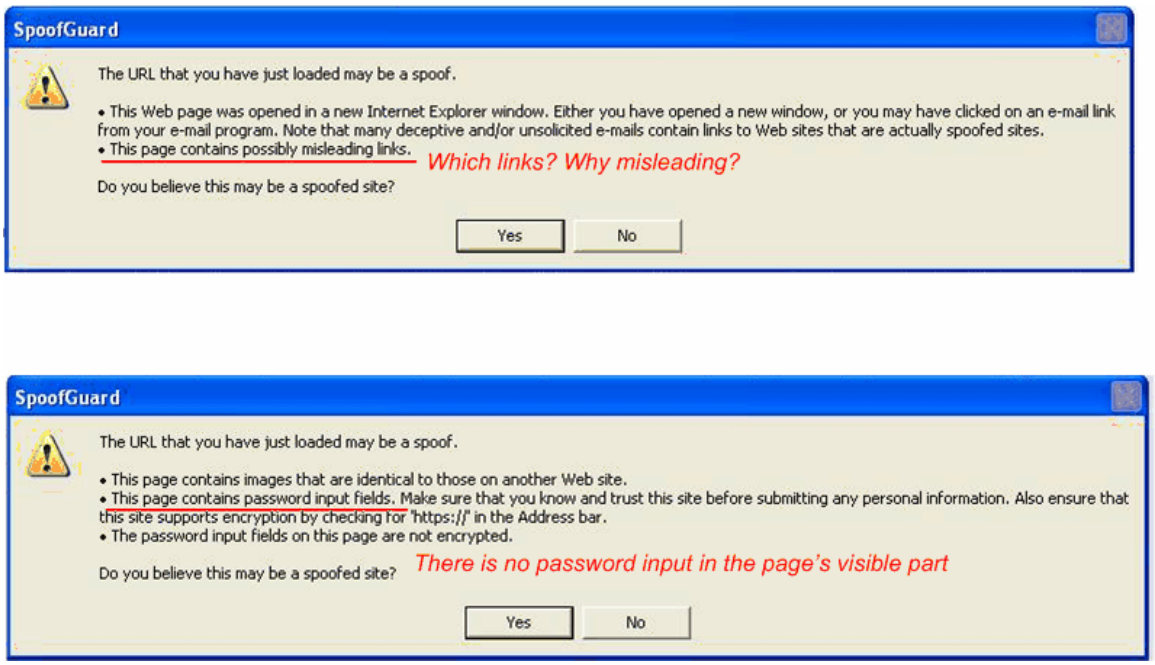


Figure 12: Confusing warnings from SpoofGuard

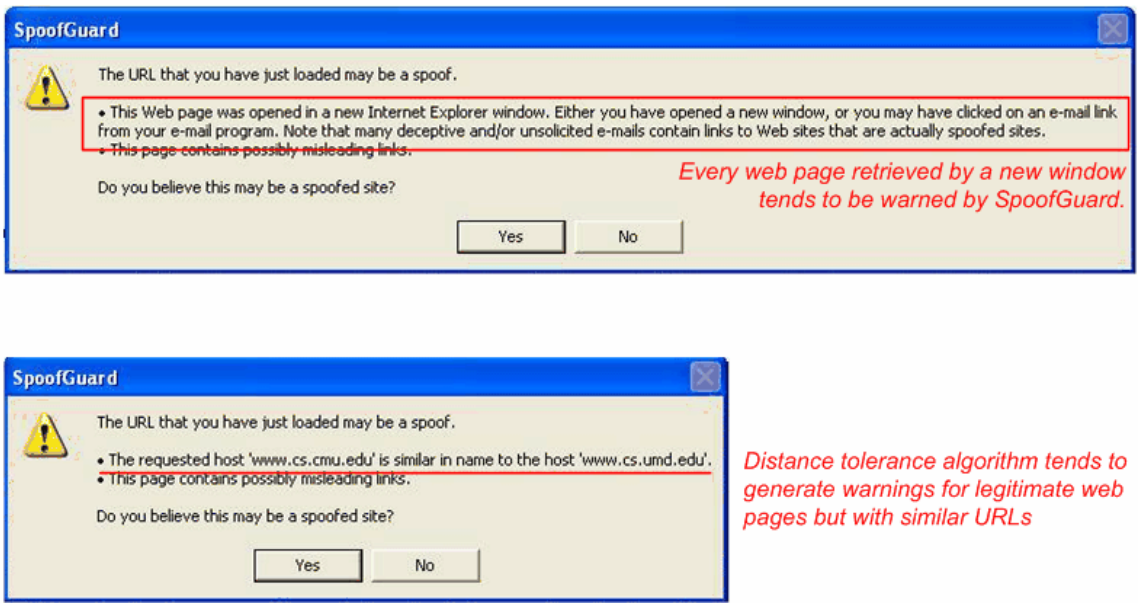


Figure 13: False positive warnings from SpoofGuard

SpoofGuard does not modify user's online behavior. Users can click whatever links or buttons in an online message.

SpoofGuard evaluates the post data before it is posted to a remote server. The evaluation tries to detect if sensitive data is sent to phishing servers. However, the evaluation is based on some assumptions which may not be valid with sophisticated attacks:

- Data is posted by HTML <form> tag. This assumption is invalid if the phishing page uses JavaScript to encode posted data or uses Flash form submission [Moock 2003].

- Each input tag represents an independent data unit, such as password or credit card number. The author mentioned how to trick this assumption in the same paper.

In general, SpoofGuard is a good start to fight phishing attacks at the client side. But if it only limits its solutions based on the superficial characteristics derived from existing phishing reports, although it is extensible, it will be eventually like current virus detection solutions that need constant update but still fail to detect the latest attacks.

Chapter 3. Proposed Solutions

Overview

We propose to use a detection and visualization chain to prevent users from interacting with the phishing messages. When a message arrives at the user's computer, it is under the source analysis. That is, the system tries to figure out if the message publisher is trustworthy or not. Moreover, since human user's always do visual identification of web sites and most current phishing attacks try to fake the appearance of well-known organizations, the system uses perception analysis to simulate user's visual perception process in order to detect the discrepancy between where the message is actually from and where the message claims to be from. The security analysis results will be presented to users with the message itself at the presentation step using content-integrated visual cues to get user's attention. It is possible that users will ignore such visual cues, especially our detection rules in source analysis and perception analysis are heuristic and will introduce errors. Then the system still has the opportunity to prevent the dangerous interactions at the action step. Since most current phishing attacks deceive users into submitting their personal data through web forms. Form analysis can be used to detect the user's expectation, such as what kinds of information the user expects to submit and which organization is expected to receive the data. If the user's expectation does not match the system implementation, warnings can be very aggressive to block user's current dangerous interaction. If the system cannot fully derive the user's expectation, it can explicitly ask user's intention in a natural and intuitive way. At last, the system has to use user's expectation as a "semantic sandbox" to guide the system operation. Although single step or single analysis will not work very well to block phishing attacks, we expect that a chain of them will effectively block phishing while not bother user's normal online interaction.

Source analysis

In order to prevent phishing, the first step that the system can do is to get the correct identity information of the message source and then decide if the source is trustworthy or not. From the system's point of view, every machine on the Internet is identified by its IP address. It does not have semantic meanings until humans assign some meanings to it.

- **Blacklist:** Blacklist indicates whether a machine with an IP address is good or bad. A bad machine means that it was known to be used by attackers to send fraudulent messages on the Internet. The blacklist publisher assigns the "goodness" (the machines' IP addresses are not in the list) and the "badness" (the machines' IP addresses are in the list) to all Internet machines. The problem of the blacklist is that it is hard to keep the list up-to-date since it is easy to register new IP addresses in the Internet. After an attacker gets a new IP address, he can broadcast solicit emails and wait for victims. Without constant updates, the blacklist gives human users a wrong sense of security to newly-setup phishing sources.
- **URL:** A URL is a "nickname" assigned to a machine for human users to easily remember. It is also used to derive the identity information of the machine. Even the security tips [Justice 2003] suggested users to "take note of the header address on the web site" and advised them that "if a website address is unfamiliar, it's probably not real." Although such derivation is correct most of the time (for example, www.ebay.com represents eBay Inc. and www.verizon.com presents Verizon company), there are some dangerous exceptions: 1) Does www.ebay-members-security.com represent eBay Inc.? 2) Does www.mycitibank.net represent Citibank, a legitimate financial organization? What about www.citibank.com.intl-en.us? 3) Is www.usefulbill.com a legitimate web site that helps users to solve billing problems? If you answer "Yes" to one of these questions, you are tricked by the real phishing attacks reported to APWG. Attackers tweak the URLs to make users derive wrong identity information from them.
- **WHOIS database:** The WHOIS database [Harrenstien & Stahl & Feinler 1985] stores the registry information of Internet domain names. A query provides organization information including company's name and address, contact information, and the domain's creation and expiration dates. The WHOIS data is designed for human users to read in order to form a sense of to which organization they are communicating. Figure 14 shows the result of the WHOIS query for domain name eBay.com. A common feature of phishing web sites is the short duration of their domain names. From APWG's archives, we find that the domain names of most of the phishing web sites were registered with a very recent creation time (a few days ago). The reason is straightforward

because in general the phishing site is short-lived. It is unnatural for a phisher to register a domain and set up a server and then wait for a year to lure the victims. Therefore, the creation time is an indicator for possible phishing sites, with exceptions of the newly registered legitimate domains. The problem of the WHOIS database is the reliability of its data. "A registrant only needs to fill out an online form, and a domain name is automatically reserved for him. As such, the process is ideal for cybersquatters or other scammers looking to defraud businesses or consumers." [Rosencrance 2002] For example, the domain name ebayco.org, used by a real phishing attack in March 2004 [APWG 2004-03-26], has the same registrant information as ebay.com. Based on the registrant information, it is natural but dangerous to think that ebayco.org is an extended domain name registered by eBay.

```
Registrant:
eBay, Inc. (EBAY-DOM)
   2005 E. Hamilton Ave., Ste. 350
   2125 Hamilton Ave
   San Jose, CA 95125
   US

   Domain Name: EBAY.COM

   Administrative Contact:
      Payable, Accounts  (PFZHEBJLGI)              hostmaster@ebay.com
      2145 E Hamilton
      San Jose, CA 95125
      US
      408 376 7400

   Technical Contact:
      Master, Host  (WZCTOKYPUI)                   hostmaster@ebay.com
      2145 E Hamilton
      San Jose, CA 95125
      US
      408 376 7400

   Record expires on 03-Aug-2010.
   Record created on 04-Aug-1995.
   Database last updated on 1-Sep-2004 05:07:14 EDT.

   Domain servers in listed order:

   SJC-DNS1.EBAYDNS.COM        66.135.207.137
   SJC-DNS2.EBAYDNS.COM        66.135.207.138
   SMF-DNS1.EBAYDNS.COM        66.135.223.137
   SMF-DNS2.EBAYDNS.COM        66.135.215.5
```

Figure 14: An example of WHOIS data

- **SSL certificate:** SSL certificate indicates that the communication with a web page is encrypted by a public key and is confidential to the key holder, whose identity information is in the certificate. Moreover, it provides a rule of thumb about whether a web site is trusted by finding out whether the site's SSL certificate is issued by a trusted CA or not, as shown in figure 15. Since SSL is involved in sensitive information transfer, we expect that a site is reliable if its SSL certificate is issued by trusted CAs. Moreover, we also need to differentiate different CAs installed in the browsers by default, since their practice may different [Microsoft 2004]. The problem of the SSL certificate is that only some big organizations apply certificate from trusted CAs. Some legitimate organizations act as their own CAs and use self-signed certificates.

Figure 15: SSL certificate indicates the reliability of a web site

- **Catalog information:** Catalog information indicates the purpose of a web site. Such information can be retrieved from the Open Directory Project at www.dmoz.org or the Yahoo! Directory at www.yahoo.com. In order to have a web site listed in the online directories, one has to submit the URL of the web site to these directories and has some experts inspect the web content. These directories have the policy against the inclusion of sites with illegal content. "Examples of illegal material include child pornography; libel; material that infringes any intellectual property right; and material that specifically advocates, solicits or abets illegal activity (such as fraud or violence)." [Dmoz] The reliability of a directory depends on its editor's discretion. A site's placement in a directory is subject to change or deletion at any time if the content is mismatching with the placement. The problem of the catalog information is that some legitimate organizations may not bother to apply for the catalog.
- **Link popularity:** Link popularity score of a web site is the total number of external sites that link to it. We treat each link as a reference to the site and we expect a site can only be referred after its content is examined. The link popularity score thus indicates how well the content is inspected under publicity. The higher the score, the more reliable the content since more external entities inspect it. One problem of the link popularity is link farms [Thomason 2002]. We expect that search engines deal with it and they return the score indicating the true popularity of a web site on the Internet. Another problem of the link popularity is that small or newly-setup legitimate organizations do not have high popularity scores.
- **Databases from trusted third parties that inspect Internet business practice:** Better Business Bureau (BBB) information system (http://search.bbb.org/search.html) can be used to find reliability report of a web site. The report is for human users to read in order to form a sense of reliability of the other party that they are communicating. Figure 16 shows the reliability report of www.ebay.com. Since BBB is dedicated to decide good or bad business practices, its report is expected to be trustworthy. There are even suggestions that "if you hear some offer through email or even by telephone, you can always check it out with the Better Business Bureau," [Smalls 2004] which means that BBB is a possible good third party authority for trustworthiness. One problem for the BBB report is that the information is for human to read but not for computers to parse and analyze. TRUSTe (http://www.truste.org/about/member_list.php) is another third-party authority.

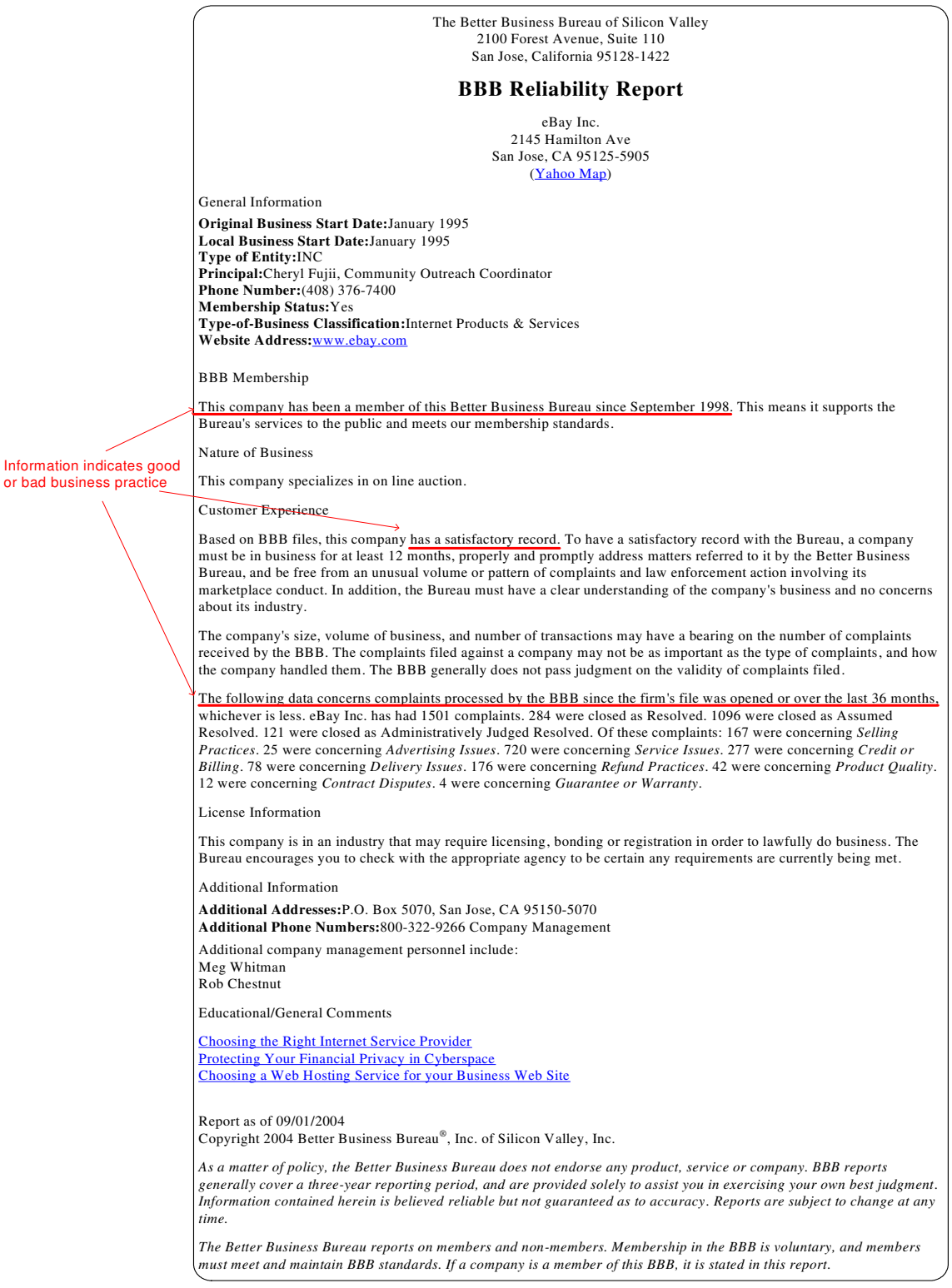


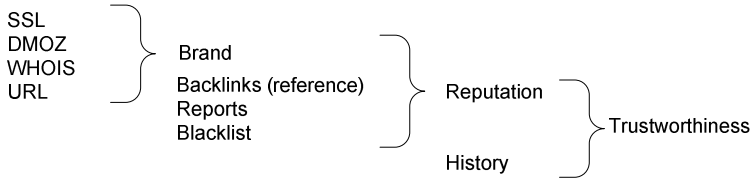
Figure 16: An example of reliability report from BBB

- **User's online history:** The trustworthiness of a web site is also decided by the user's online history: the more the user interacts with a web site, the more trustworthy the web site is to the user. Here the interaction means that the user submits data at the web page from the web site and

the basic assumption here is if a user submits data at some web site, no matter what types of the data, the user trusts the site for later interactions.

Therefore, there are already many available online data sources that provide information for a given machine and the system can decide the trustworthiness of the machines based on these data sources. We propose to take the SpamAssassin's approach [Mason 2002]. Although a single source cannot mark a source is trustworthy or not (since attacker may spoof WHOIS data, may increase his site's Google link value using link form, and may even fool Verisign to sign certificate for his phishing server), many sources together can reliably decide if a source is trustworthy or not. The heuristic rule can be built based on both the real case analysis from APWG archives and trustworthiness status of the current Internet on a whole, which is constantly monitored by our web crawlers.

Every heuristic rule has error rate, both false positive and false negative. In term of phishing, the false positive error is that a site is legitimate but is not reputed so that it fails all the reputation tests in the heuristic rule. The heuristic rules can be fine-tuned by users and also are constantly updated based on current Internet status. We also want to find out the statistical percentage of the first-time-accessed web sites in all sites that are accessed by users every day. If users only access to a few new sites and if we treat the sites in the users' online history is trusted, even we make the criteria of reputation rigid the users will not be bothered too much. On the other hand, the false negative error in term of phishing is that a site is phishing but is categorized as non-phishing. This effect of this error can be significant since it gives users a wrong sense of security. We expect that our rule is strict enough to make the false negative rate as low as possible. This rule can be continuously tested by newly recorded phishing cases from APWG. We also need to figure out what a phishing site can do to bypass the heuristic rule and if it is worth to bypass the rules by trying to fool multiple third-party authorities, such Google, DMOZ, WHOIS server, etc.



1. + creditability report + SSL from well known CA	1. highly reputed (blue)
2. – creditability report + SSL from well known CA + (DMOZ, WHOIS, Backlinks)	2. reputed (green)
3. – creditability report – SSL from well known CA + (DMOZ, WHOIS, Backlinks)	3. well-known (yellow)
4. others	4. not well known (red + 2-way switch)
5. – creditability report – brand – backlinks	5. no reputation (black + 3-way switch)
6. + blacklist	6. danger (block)

Browser is a frame for web pages. There should be different actions for clicking within the "site" and clicking out of the "site". Clicking out of the "site" will bring trustworthiness information of the coming site to the user.

The source analysis is just the first link in the detection and visualization chain. A message from alleged trustworthy source may be phishing as long as the site can find way to bypass the source analysis rule. Therefore, we need to examine the message itself.

Perception analysis

The trustworthiness of the message publisher is not reliable enough to block all phishing messages. The content of the message will be analyzed since all phishing messages make a claim (*e.g.*, eBay requests users to update their account) that is not related with the actual data source (*e.g.*, a server in South Korea). Since a phishing message tries to make itself look like a message from a well-known legitimate organization, the system can simulate the user's visual perception process to derive what the organization

the message is claimed to be from and to check if there is any relationship between the claimed source and the actual data source. If there is no relation, the message is highly likely to be phishing. But how does the system form the visual perception process? We propose to visually cluster the web pages.

We assume that users interact with the authentic web site of an organization before they receive a phishing message pretending to be from the same organization. At the first time when users interact with the web site, which is assumed to be authentic, all web pages from the authentic web site are crawled and supervise-learnt and then clustered and generalized by some feature vectors. The supervised learning is guided by the saliency model, which specifies which part of the message presentation is most noticeable to users and should be assigned high learning weight.

Later, we expect that the phishing message which looks like legitimate messages from an organization will be categorized to the same cluster representing messages from that organization and thus the claimed source is available to the system. The system can then compare the message's actual source with the claimed source derived from the visual clustering for relationship. If the system can reliably detect the deception, it can aggressively warn the user as shown in figure 17. The system does not use modal warning box because it hinders users the ability to examine the web page. Instead, the system blocks all the outgoing connections, e.g., links and forms, which can be triggered by the current page until the user responses to the warnings. Importantly, since the warning pane contains recommended actions, i.e., "close the browser" and "go to the authentic eBay site", this warning should not be spoofed. One solution is to use a dedicated toolbar that is out of the remote party's control.

The proposed project is to visually cluster web pages from well-known organizations, e.g., eBay or Citizen Bank, to see if such visual clustering is possible and if yes, which clustering algorithm is most accurate and efficient.

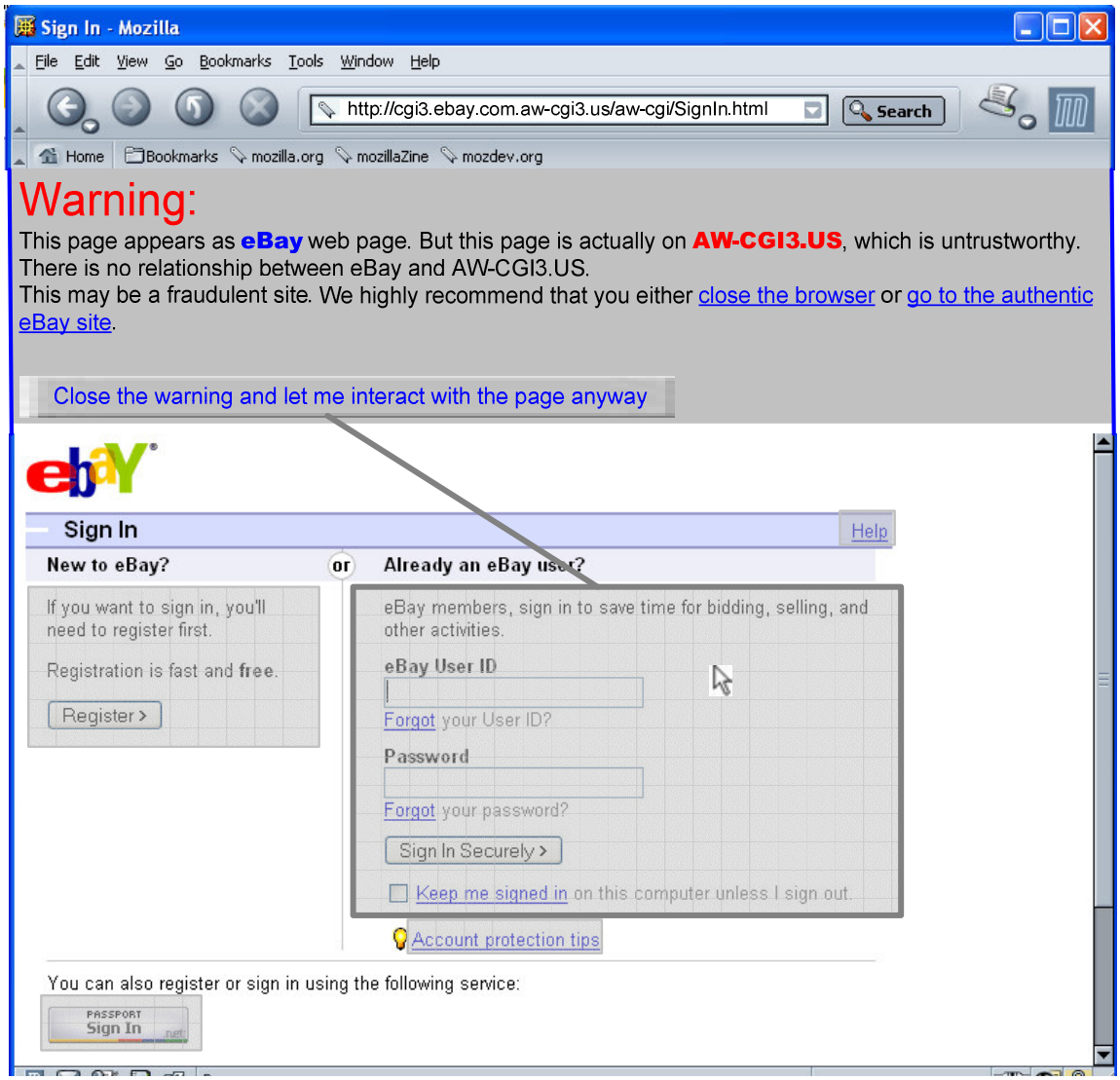


Figure 17: Warning based on perception analysis

Content-integrated visual cues

Current web browser uses a set of peripheral visual cues to reflect the system model to the user. We expect that all the peripheral cues are not effective because users only concentrate on the content of the message in order to finish their main task and to perform task securely is always their secondary goal. Therefore we propose to move peripheral visual cues to the center of the user's attention, that is, to integrate those cues with content, so that as long as users see the content, they see the cues as well. However, since the cues are integrated with the content and the content display is totally controlled by the remote party, the cues should not be hidden (if the cue is negative to reveal that the current page is dangerous) or faked (if the cue is positive to support that the current page is safe). We propose to develop additive cues that are guaranteed to appear and make discrepancies between the mental model and the system model visually apparent. The hue, value, size and position of the additive cues should be adaptive to the current message presentation by using saliency models to guide the presentation of the cues. Since phishing messages ask users to submit their data through web forms, web forms need to be highly annotated. Figure 18 shows an example to alert users that the data source is heuristically untrustworthy.

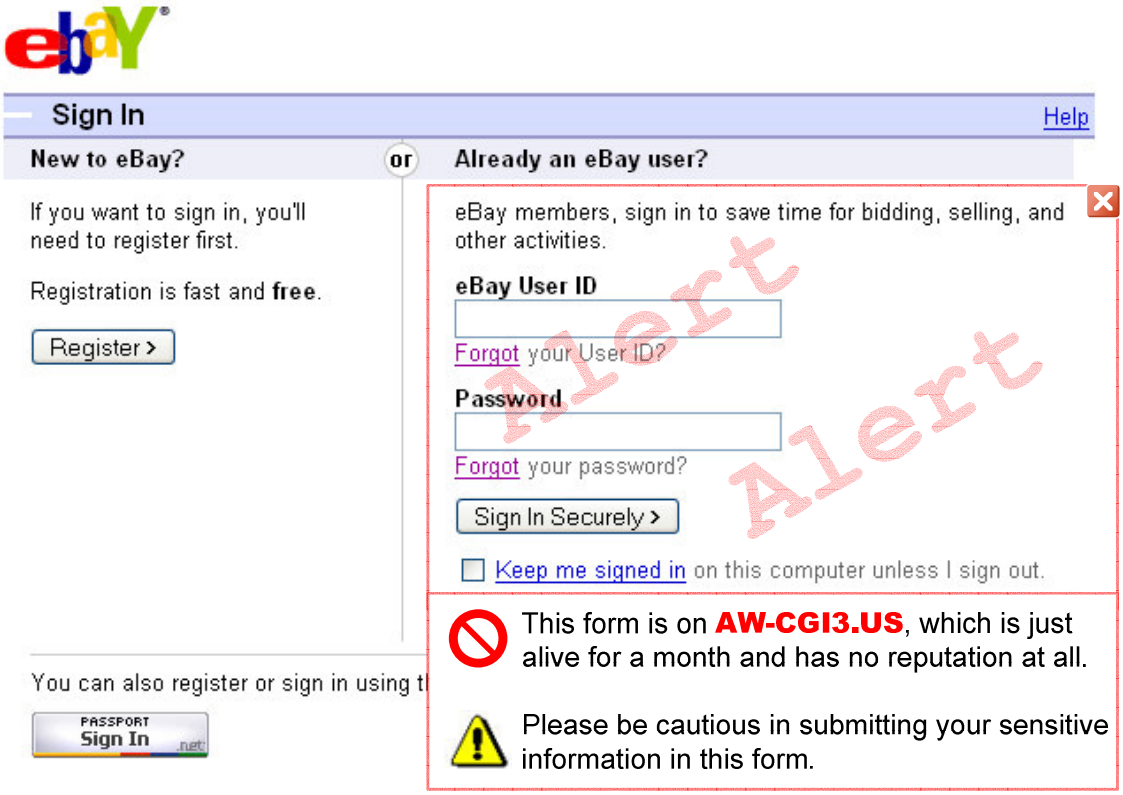


Figure 18: Warning wrapping around form with watermark style

Currently, most phishing attacks use email as a lure to trap users into the phishing web sites. Therefore, the system can help to block the phishing attacks at the first place, *i.e.*, the email message, by displaying the link destination information since most current phishing sites are not trustworthy. We expect this display can raise the user's suspicions on phishing emails. As shown in figure 19, the warning is automatically inserted into the email when the user moves the mouse over the dangerous link. In order for the user to aware the warning, the given link is disabled and a warning with a new link is attached. In this way, we also change the user's action on the dangerous link, if he wants to continue anyway.

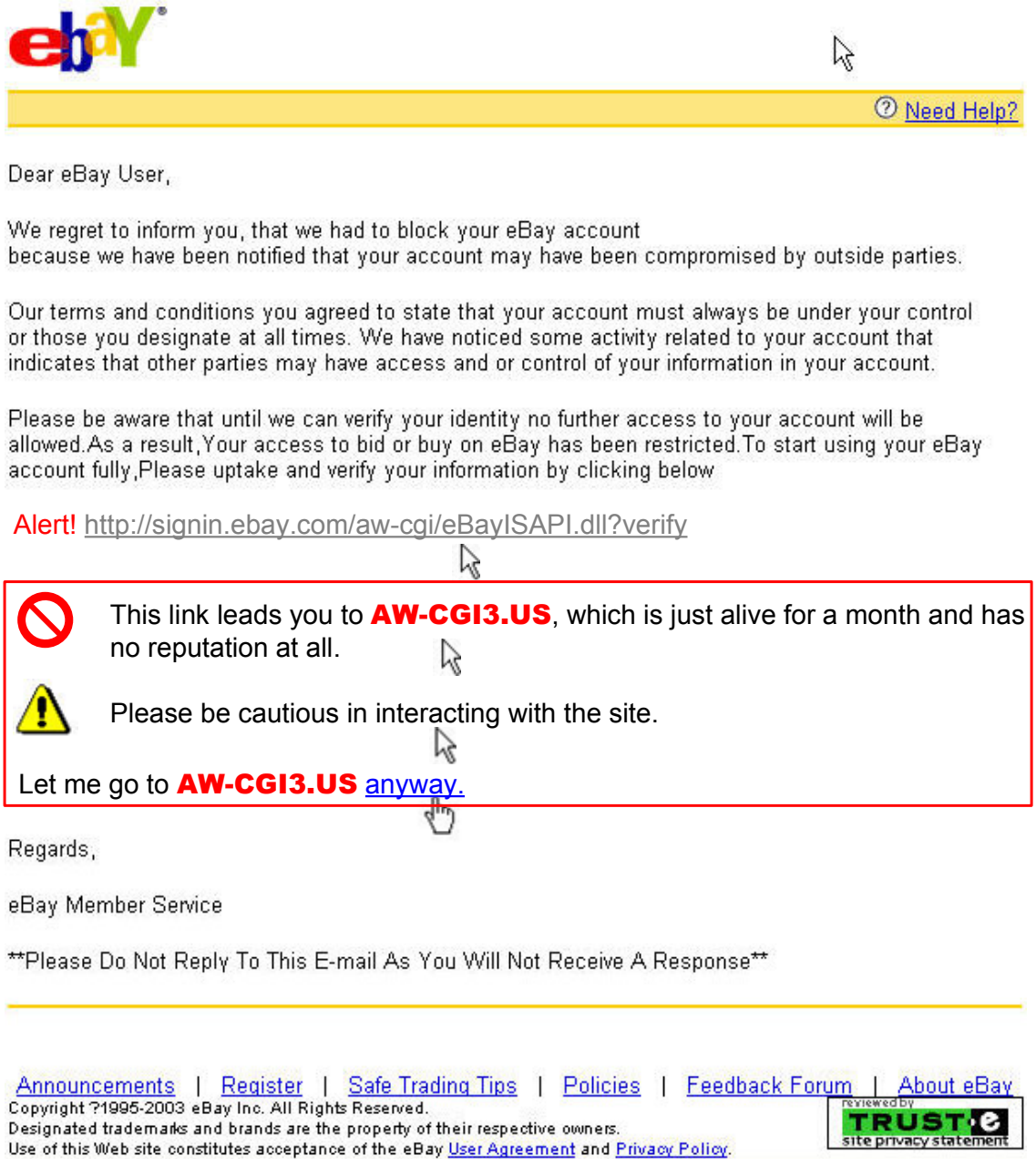


Figure 19: Warning inserted into email in the HTML format

Form analysis

A true phishing attack actually request personal information from the user, so forms requesting personal information should be subject to stricter scrutiny by the browser in order to figure out what the form is requesting. On the other hand, a phishing attack might use a variety of techniques to conceal this information: e.g., using image instead of textual label, or using JavaScript or Java applets or even flash-based forms that are less amenable to content analysis. We propose two ways to figure out the request from the form: perception analysis of the form appearance and action analysis of the user's keyboard typing.

In term of the perception analysis of the form appearance, there are four steps:

1. Use image analysis techniques to find out the form section in the screenshots of both the current page and the popup windows. If there is HTML <FORM> element, content analysis of the page in

HTML format can help to decide the location and the size of the form. This analysis may also postpone until the keyboard input is focused on a particular text entry because the location of the entry is a good reference point to find out the location of the form that contains it.

2. Use image analysis with help of the content analysis of the HTML page to group the labels with their corresponding entry fields.
3. Use optical character recognition (OCR) techniques to figure out what the labels are.
4. Assign the recognized labels into different personal information categories.

In term of the action analysis of the user's keyboard typing, the browser can first learns the user's personal information by observing how the user fills in forms on legitimate web sites over time, using simple analysis to identify legitimate forms requesting passwords, credit card numbers, security questions, and other sensitive fields. When a user types data at an unknown site, his input is constantly monitored and searched for matches with the stored sensitive values. Moreover the analysis of the keyboard input can in turn improve the recognition accuracy of the labels. For security, the user's personal information discovered by the browser is not stored directly, but instead as secure hash values. The browser can also keep track of which sites have received which of the user's personal information, which it can display to the user by the identity theft forensics after the fact.

We also propose that forms should reflect its system operation to the user by using content-integrated cues. Two pieces of information we want to display with form is where the data goes and if the data is transmitted securely. Both pieces of the information is hard to get reliably by analyzing the source code of the web page since the attackers may use the sophisticated JavaScript code to prevent code analysis. In order to reliably get where and how the data goes, we propose to simulate form submission when the user moves his mouse over the "submit"-kind button and is about to click it and then the system stop the submission immediately before the real net connection happens. At this point the system can find out the correct answer of where and how the data is transmitted and then it can reflect the information to the user. The same display method as shown in figure 19 can be used. The provided button is disabled and the data transmission information (to where and how) is inserted into the form. Moreover, through the content analysis of the web page source code and the perception analysis of the web page appearance, the browser may derive with some reliability the site that the users expect their input data to go to. Then the browser can add the derived site as an alternative choice for the user. Therefore, the user has to choose their expected site from a list of sites that includes the actual destination, the expected destination plus some heuristically-chosen sites to which the user has submitted personal information in the past. Wu, *et al* [Wu & Garfinkel & Miller 2004] have found out that users make fewer mistakes and thus the system is more secure when users have to actively choose their intended action from a list of choices then when they visually check the security-related status display and then confirm or give up their action based on the warnings.

Intention request

When there is a gap between the user's intention of his action and the system implementation of the same action, the user can express his intention explicitly to the system in a system-understandable way in order to bridge the gap. Security tips suggest users to explicitly express their intentions by opening a new browser and manually typing the URL of the site that they want to contact instead of blindly clicking the given links in emails or web pages. However, the URL typing does not always correctly reflect the user's intention. For example, when user types http://manager-earthlink.com, does the user mean to contact to EarthLink or does the user want to contact a phishing site at manager-earthlink.com [APWG 2004-10-26]? We expect that using keywords from the natural language, the same as Google keyword search, can help users to express their intentions correctly. Think about when you want to go to an organization's web site but you do not know the exact URL, what should you do? Google is a good choice for you to input the organization's name and hopefully the intended URL is displayed in the first 10 results. We propose to use the same idea here. When the browser cannot derive what the user intended to do after content analysis and perception analysis, it can simply prompt a dialog box asking users: "where do you expect this information sent to?" Based on the user's answer, the browser can decide if it sends the answer as a Google search item directly or if the answer needs some pre-processing with common knowledge or context information before it is sent to Google. In this way, Google translates user's natural language to the URL that the system can understand. Actually, Google has done this thing all the time.

Guided system operation

At the end of the human-computer interaction, certain system operation is triggered and should be guided. If the data source of the message is not heuristically trustworthy, the browser has several mechanisms to alert the users about their interactions with the potential dangerous message. And if the user ignores all the implicit and explicit warnings, that data source should be treated as trustworthy. On the other hand, if the data source of a message is heuristically trustworthy, either because it is reputed or because it is in the user's interaction history, the system operation coded by this message should still be monitored since it is possible that the message is altered by attackers. That is, if the system operation triggered by this message is totally different from the typical operations provided by the data source before, (*e.g.*, the form submission in the current eBay message sends user's data to a server in South Korea instead of an eBay's server) it is highly possible that the message is altered for nasty purposes. In a word, the triggered system operation by the messages from trustworthy data sources should be monitored for the discrepancy between what it is doing and what it used to do and the discrepancy between what it is doing and what it is supposed to do.

Chapter 4. Evaluation

The anti-phishing techniques developed in this work will be evaluated in two aspects:

1. Security: Can the anti-phishing techniques effectively prevent current phishing attacks as well as the attacks that have not existed yet but we anticipate that they may appear in the near future?
2. Usability: Do the anti-phishing techniques distract users from their normal tasks?

Since our anti-phishing solution is based on a chain of detection and visualization schemes, we will evaluate the following parameters in term of security:

- The error rates, especially the false negative rate (*i.e.*, treat phishing messages trustworthy) of the detection schemes, including source analysis, content analysis, image analysis, perception analysis and action analysis
- The effectiveness of the content-integrated visual cues to alert true phishing messages
- The spoof rate of the content-integrated visual cues

We can test them by both reconstructing the existing phishing attacks from APWG archives and creating new phishing attacks with novel tricks in order to bypass the heuristic detection rules and to spoof the security visual cues.

In term of the usability, we will evaluate the following parameters:

- The false positive rate of the detection schemes: how many legitimate sites are treated dangerous when users first interact with them?
- The distraction of the content-integrated visual cues with legitimate sites and the degradation of the anti-phishing schemes to the system performance
- The user's subjective attitude towards our solutions: is it too bothersome so that the user tends to disable them?

We can test them by implementing our solutions as web browser or email client plug-ins so that users can use and evaluate them with their daily Internet interactions.

The implementation of the detection and visualization chain and its evaluation should be in an iterative design loop.

Chapter 5. Time-line

November, 2004

- Submit thesis proposal
- Start the first round of design-implementation-evaluation loop
    - Design and implement heuristic rules for source analysis in order to decide if a source is trustworthy or not using a web crawler and anonymous HTTP tracing log
    - Design and implement the intention-requesting user interface
    - User study for effectiveness and distraction of different security visual cues, including our proposed content-integrated visual cues

January, 2005

- Design and implement the guided system operation
- Based on the result of the user study in November, design and implement content-integrated visual cues for both form submission and link clicking in email

February, 2005

- Study and implement the algorithm to do visual clustering of web pages based on the saliency model
- Design and implement form analysis mechanism with OCR and keyboard input monitoring

April, 2005

- Start to evaluate the first version of phishing detection and visualization chain

June, 2005

- Improve the both the security and usability of the phishing detection and visualization chain based on the evaluation result
- Extend the chain to prevent users from downloading hostile code as email attachment

August, 2005

- Start to evaluate the improved version of phishing detection and visualization chain

October, 2005

- Start the final round of design-implementation-evaluation loop based on the evaluation result from last round
- Think about phishing with mobile phones using SMS or WAP and the solutions to prevent it with limited computation power and display screen

December, 2005

- Final evaluation of the phishing detection and visualization chain
- Write the thesis

References:

[APWG 2004-02-24] *MBNA "MBNA informs you!"*. Anti-Phishing Working Group, February 24, 2004. URL http://www.antiphishing.org/phishing_archive/MBNA_2-24-04.htm

[APWG 2004-03-09] *eBay - "NOTICE eBay Obligatory Verifying - Invalid User Information"*. Anti-Phishing Working Group, March 09, 2004. URL http://www.antiphishing.org/phishing_archive/eBay_03-09-04.htm

[APWG 2004-03-26] *eBay - "Email regarding pre-indefinitely suspended from eBay"*. Anti-Phishing Working Group, March 26, 2004. URL http://antiphishing.org/phishing_archive/03-26-04_eBay(Email_regarding_pre-indefinitely_suspended_from_eBay).html

[APWG 2004-05-24] *Reports of Email Fraud and Phishing Attacks Increase By 180% in April; Up 4,000% Since November.* APWG press release, May 24, 2004. URL http://www.antiphishing.org/news/05-24-04_Press%20Release-PhishingApr04.html

[APWG 2004-08-06] *AOL - "Urgent message from AOL member services"*. Anti-Phishing Working Group, August 06, 2004. URL http://antiphishing.org/phishing_archive/08-06-04_AOL_(Urgent_message_from_AOL_member_services).html

[APWG 2004-10-26] *Earthlink - "EarthLink Account Expired - Update Now"*. Anti-Phishing Working Group, October 26, 2004. URL http://antiphishing.org/phishing_archive/26-10-04_Earthlink(EarthLink_Account_Expired)/26-10-04_Earthlink(EarthLink_Account_Expired).html

[Benway & Len 1998] Jan Panero Benway, David M. Lane. *Banner Blindness: Web Searchers Often Miss "Obvious" Links.* Internetworking. December 1998. URL http://www.internettg.org/newsletter/dec98/banner_blindness.html

[Boneh & Mitchell & Ross] Dan Boneh, John Mitchell, Blake Ross. *Web Password Hashing*. URL http://crypto.stanford.edu/PwdHash/

[Chaiken 1980] Chaiken, S. *Heuristic Versus Systematic Information Processing and the Use of Source Versus Message Cues in Persuasion.* Journal of Personality and Social Psychology. 1980.

[Chou & Ledesma & Teraguchi & Mitchell 2004] Neil Chou, Robert Ledesma, Yuka Teraguchi, John C. Mitchell. *Client-Side Defense Against Web-Based Identity Theft.* 11th Annual Network and Distributed System Security Symposium, 2004. URL http://theory.stanford.edu/people/jcm/papers/spoofguard-ndss.pdf

[Cybenko & Giani & Heckman & Thompson] George Cybenko, Annarita Giani, Carey Heckman, Paul Thompson. *Cognitive Hacking: Technological and Legal Issues.* Dartmouth College. URL http://www.ists.dartmouth.edu/IRIA/projects/semantic/lawtech.doc

[Dmoz] *How to suggest a site to the Open Directory.* Dmoz Open Directory Project. URL http://www.dmoz.org/add.html

[Earthlink] *EarthLink Toolbar: Featuring ScamBlocker.* EarthLink. URL http://www.earthlink.net/earthlinktoolbar/download/

[Ebay] *Tutorial: Spoof (fake) Emails.* eBay. URL http://pages.ebay.com/education/spooftutorial/

[Ebay-AG] *eBay Toolbar.* eBay. URL http://pages.ebay.com/ebay_toolbar/

[F-secure 2004] *F-Secure Virus Descriptions : Bagle.N*. F-SECURE®. URL http://www.f-secure.com/v-descs/bagle_n.shtml

[Felten & Balfanz & Dean & Wallach 1996] E. Felten, D. Balfanz, D. Dean, D. Wallach. *Web Spoofing: An Internet Con Game*. 20th National Information Systems Security Conference, 1996.

[Harrenstien & Stahl & Feinler 1985] K. Harrenstien, M. Stahl, E. Feinler. *RFC 954 - NICNAME/WHOIS.* Network Working Group. 1985

[Herzberg & Gbara 2004] Amir Herzberg and Ahmed Gbara. TrustBar: *Protecting (even Naïve) Web Users from Spoofing and Phishing Attacks.* URL: http://eprint.iacr.org/2004/155.pdf

[Justice 2003] *FBI Says Web "Spoofing" Scams are a Growing Problem*. Federal Bureau of Investigation, Department of Justice, 2003. URL http://www.fbi.gov/pressrel/pressrel03/spoofing072103.htm

[Mack & Rock 1998] Arien Mack, Irvin Rock. *Inattentional Blindness.* MIT Press. 1998

[Marsh & Dibben 2003] Stephen Marsh and Mark R. Dibben. *The Role of Trust in Information Science and Technology.* Annual Review of Information Science and Technology. 2003.

[Mason 2002] Justin Mason. *Filtering Spam with SpamAssassin.* HEANet Annual Conference. 2002.

[Microsoft 2004] *Microsoft Root Certificate Program Members.* Microsoft, April 2004. URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnsecure/html/rootcertprog.asp

[Mitnick & Simon 2002] Kevin Mitnick and William Simon. *The Art of Deception*. John Wiley & Sons. 2002.

[Moock 2003] Colin Moock. *A Study in Flash Form Submission.* O'Reilly Network. 05/20/2003. URL http://www.oreillynet.com/pub/a/javascript/2003/05/20/colinmoock.html
[Passmark] *Protecting Your Customer from Phishing Attacks.* PassMark Security. URL http://www.passmarksecurity.com
[Reisberg 2001] Daniel Reisberg. *Cognition: Exploring the science of the mind. 2nd Edition.* W.W. Norton & Company, 2001.
[Rosencrance 2002] Linda Rosencrance. *Task force report looks at accuracy of Whois data.* Whois Source, December 05, 2002. URL http://www.whois.sc/news/2002-12/whois-accuracy.html
[Schneier 2000] Bruce Schneier. *Semantic Attacks: The Third Wave of Network Attacks.* Crypto-Gram Newsletter, October 15, 2000. URL http://www.schneier.com/crypto-gram-0010.html#1
[Smalls 2004] YaVonda Smalls. *High-speed heist: Internet scams, identity thieves may leave students with empty wallets*. The Ball State Daily News. November 22, 2004.URL: http://www.bsudailynews.com/vnews/display.v/ART/2004/11/22/41a18a3830cd4
[Spoofstick] *SpoofStick.* CoreStreet. URL http://www.corestreet.com/spoofstick/
[Sullivan 2004] Bob Sullivan. *Consumers still falling for phish.* MSNBC. July 28, 2004. URL http://www.msnbc.msn.com/id/5519990/
[Thomason 2002] Larisa Thomason. *Promotion Tip: Link Farms Grow Spam.* NetMechanic. April 2002. URL http://www.netmechanic.com/news/vol5/promo_no7.htm
[Tim Berners-Lee 1997] Tim Berners-Lee. *Cleaning up the User Interface*. World Wide Web Consortium. February 6, 1997. URL http://www.w3.org/DesignIssues/UI.html
[Tropsoft] *Secure Browser.* Tropical Software. URL http://www.tropsoft.com/secbrowser/
[Tygar & Whitten 1996] J. D. Tygar, Alma Whitten. *WWW Electronic Commerce and Java Trojan Horses*. Proceedings of the Second USENIX Workshop on Electronic Commerce. 1996
[Wu & Garfinkel & Miller 2004] Min Wu, Simson Garfinkel, Robert Miller. *Secure Web Authentication with Mobile Phones.* DIMACS Workshop on Usable Privacy and Security Software. 2004.
[Ye & Smith 2002] Zishuang Ye, Sean Smith. *Trusted Paths for Browsers*. Dartmouth College. 2002
[Ye & Yuan & Smith 2002] Zishuang Ye, Yougu Yuan, and Sean Smith. *Web Spoofing Revisited: SSL and Beyond*. Technical Report TR2002-417, Dartmouth College. 2002