Designing and Evaluating a Petname Anti-Phishing Tool

Ka-Ping Yee
University of California, Berkeley
Berkeley, CA 94720
ping@zesty.ca

ABSTRACT

Phishing is a misidentification problem. To defend against it, users must have a reliable way to identify the websites they visit. The Petname Toolbar is a proposed solution. This brief paper examines the nature of the problem, discusses the Petname concept and implementation, and describes the design of a user study to evaluate its effectiveness.

1. INTRODUCTION

Gartner Research estimated that US$1.2 billion in direct losses was suffered by U. S. banks and credit card issuers due to identity theft fraud against phishing attack victims in 2003 [3]. The rate of attacks increased steadily during 2004 [2].

The problem is fundamentally a matter of mistaken identity. In a successful attack, the user arrives at a fraudulent website F, believing it to be a legitimate site L, and enters secrets previously shared between with L. Some examples of the ways that users come to identify F as L are:

- The user arrived at the site via a link in an e-mail message that appeared to come from L.
- The graphics, text styles, and overall appearance of F match what the user has come to expect from L.
- The domain name at which F resides is similar to the domain name the user expects to see for L.

It is interesting to note that this class of problem was anticipated well in advance. The designers of the SSL protocol for encrypted communication on the Web were well aware of the potential for spoofing. They addressed the problem by introducing SSL certificates. Any site that wishes to use encrypted connections must obtain and present an SSL certificate. The certificate must be purchased from a certificate authority (CA); it contains information about the holder, digitally signed by the CA.

However, today's centralized certificate infrastructure has several serious trust management problems:

- The centralization of registries introduces a high-risk point of failure and a locus of political control.
- The user is required to place trust in a third party, the certification authority (CA). But the CA is selected by the website (i.e. the potential adversary), not the user.
- Today's browsers come with a list of CAs that are trusted by default. This list usually contains dozens or over a hundred trusted certificates. The user could not possibly memorize them all and may not know any of them. The browser assumes it can place the user's trust in organizations the user has never even heard of.
- What the CAs actually certify is unclear. They do not, as a rule, guarantee that the holder of a certificate for a particular company or individual name represents that company or individual.

Moreover, it is plainly evident that certificates have failed to prevent spoofing. They fail for two reasons:

1. Certificates don't actually help the user identify the site. None of the certificate information is shown to the user unless the user asks for it. It usually takes three or four clicks to get to the information, and even then the accuracy of the information is not guaranteed.
2. Certificates are not used for verification. The March 2005 issue of the APWG report on trends in phishing activity [2] finds that more than 96% of phishing sites use plain, unencrypted connections on port 80.

2. FAILURE MODES

The APWG report for July 2004 [1] lists the targets most frequently attacked by phishing. Here are the top five:

Number of attacks in July 2004 (total: 1974)	Target site	SSL on login page
682	Citibank	yes
622	US Bank	no
255	eBay	yes
147	PayPal	no
41	AOL	no

Three of these five do not enable SSL on the page where the user enters their password, though the password itself is encrypted. This and the figures in the March 2005 report [2] yield estimates of maximum potential effectiveness for some classes of solutions:

- Any solution that expects users to enter passwords only on SSL-enabled pages will fail to address at least 41% of phishing sites (US Bank, PayPal, and AOL above).
- Any solution that judges legitimacy based on the content of SSL certificates will fail to address at least 96% of phishing sites (those known to use port 80).
- Any solution based on correcting variants of well-known domain names will fail to address at least 69% of phishing sites (the 48% that use IP numbers and the 21% that use domain names unrelated to the target).
- Any solution based on generic rules applied to the URL and page content will fail to address at least 21% of phishing sites (since they have unrelated domain names and the attacker controls the page content).

3. THE PETNAME CONCEPT

As things currently stand, the attacker controls all the elements that are used to identify the site. Since the problem is a failure to identify, a solution is to give the user a means to identify.

A *petname* is a user-assigned name [5]. The defining feature of a petname is that its namespace is completely user-controlled. For example, the names of files on a GUI desktop are petnames, whereas Internet domains are global names.

Waterken, Inc. proposes a petname-based solution to phishing [6]. The user simply assigns names to known sites, after which they are clearly distinguishable from untrusted sites. Waterken has implemented this solution as an extension for the Firefox browser, the Petname Toolbar [7], which provides a text field where the user can enter a petname for the current site. When the user navigates to a site, the text field updates to show the site's petname or "untrusted" if no petname has been assigned.

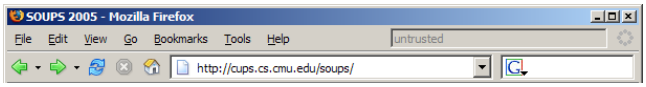

Figure 1. The Petname Toolbar is disabled for non-SSL sites.

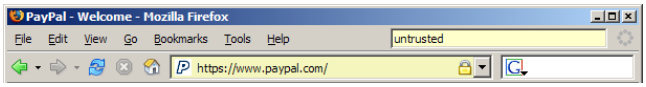

Figure 2. The toolbar is yellow and displays "untrusted" for SSL sites with no assigned petname.

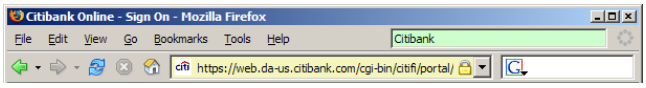

Figure 3. The toolbar is green for sites with a petname.

4. TOOLBAR DESIGN

I propose to test a toolbar like Waterken's, with a few changes. While the Waterken toolbar only works for SSL sites, the APWG data suggests a design that also assigns names to non-SSL sites.

4.1 User Training

The Waterken petname field is a single unlabelled text field in the toolbar area. I propose to add a "Name" label to its left to indicate what it is for and provide a drop-down menu. The first item on the drop-down menu, "Quick Help," will run a short animation explaining how to use the toolbar.

4.2 Assignment of Petnames

If the user can be persuaded to assign a petname to a fraudulent site, the security of the petname system is compromised. Users must be instructed to enter petnames only at sites whose URLs the user knows and has entered on their own, or at sites that can otherwise demonstrate their authenticity.

4.3 Attention to Petnames

If the user ignores the petname field (especially when it displays "untrusted"), the petname system serves no good. One way to address this is to make the toolbar a stronger color when there is no petname, such as red. Another way is to provide a non-interrupting notification when the user is entering a form or about to submit a form, such as "Form will be submitted to: [*petname*]" or "Form will be submitted to an unknown site."

5. EVALUATION PLANS

The design of the toolbar will undergo an initial low-fidelity user test, to be followed by a high-fidelity test designed to estimate the toolbar's effect on resistance to phishing.

The key issue in phishing success or failure is how the user decides whether or not to enter login information. Designing an ethically sound study that accurately mimics the context of a phishing attack requires carefully identifying the test condition. For phishing, I define the test condition as follows:

1. The subject must arrive at a website based on what the subject believes to be a legitimate recommendation.
2. The arrived-at website must be distinguishable in some way from the site claimed in the recommendation.

The following study design duplicates this test condition without deceiving or harming the subjects:

1. Users are asked to participate in a study of how they protect themselves from online fraud. They opt in and complete an initial survey about browsing habits.
2. Subjects are randomly assigned to test group A or B.
3. Subjects receive e-mail from the experimenter with software to install. Group A gets just a button for reporting suspicious sites; group B gets a petname toolbar including such a button. Both pieces of software offer an explanation of phishing and how users can protect themselves.
4. Several weeks later, subjects receive e-mail from a trusted source that asks them to log in to a website. The site actually belongs to the trusted source, but it is hosted at a different domain than usual.
5. The user either logs in to the site or clicks the button to report the site as suspicious.
6. The site then gives the user a survey asking how they made the decision to log in or click the button.

The test condition is duplicated in step 5, even though no phishing or other misrepresentation takes place. Thanks to Marti Hearst and Tyler Close for their suggestions regarding this work.

6. REFERENCES

[1] Anti-Phishing Working Group. Phishing Activity Trends Report, July 2004. http://antiphishing.org/APWG_Phishing_Attack_Report-Jul2004.pdf

[2] Anti-Phishing Working Group. Phishing Activity Trends Report, March 2005. http://antiphishing.org/APWG_Phishing_Activity_Report_March_2005.pdf

[3] Gartner, Inc. Gartner Study Finds Significant Increases in E-Mail Phishing Attacks. Press release, 6 May 2004. http://gartner.com/5_about/press_releases/asset_71087_11.jsp

[4] Mark S. Miller. Lambda for Humans: The Petname Markup Language. http://www.erights.org/elib/capability/pnml.html

[5] Marc D. Stiegler. An Introduction to Petname Systems. http://skyhunter.com/marcs/petnames/IntroPetNames.html

[6] Waterken, Inc. Trust Management for Humans. http://www.waterken.com/dev/YURL/Name/

[7] Waterken, Inc. Petname Tool. http://www.waterken.com/user/PetnameTool/