Introduction
Reports of phishing attacks have inundated the press for good reason. The cleverness of social engineering has again damaged the prospects for online commerce, and online banking in particular. The stakes are high and the consumer environment, one that is fraught with unprotected clients, is not one that is easily remedied. Without secure hardware, we must nod to the fact that absolute solutions to this problem are not currently possible.

However, with realistic risk assessment, leading to a differential approach to security, it is possible to build a cost-effective, scalable solution. We must attack on all fronts. The abilities of the user must be considered and the potential need for future changes must be built in from the beginning. The solution offered here to mitigate loss is generic but Real User’s technology, Passfaces, has some unique characteristics that allow for an innovative, effective solution that scales for consumer facing authentication needs and can be put into action NOW.

Definitions
For the purposes of this discussion, the following two attacks are key:

Phishing: A social engineering-based spoofing attack whereby a user is sent an email and spoofed into clicking on a link to a bogus website where personal information such as passwords and account numbers can be stolen. These can give the impression that the redirect is to the legitimate site (rather than a spoofed mock site) but in fact this is not the case.

Trojan Horse: Any executable code that is introduced to a computer without the owner’s knowledge or permission with the intent of doing harm. These can be introduced through attachments to emails or visits to unknown websites where security vulnerabilities in Operating Systems, browsers, and web server software are exploited. Spyware and adware can be the vehicles for Trojan horses.

These two attacks may be combined.

Phishing by Social Engineering Alone
In the cleverest of these attacks to date, the user is sent an email, ostensibly from a financial institution with which the user has a trust relationship, warning him of the dangers and proliferation of identity theft. It further states that the link provided is an attempt by that institution to correct and remove some of the individual's risk in that regard. The text associated with the link looks like the URL of the genuine site, however the actual link is to a spoof website. When this link is followed, the spoofing website serves the user a web page that is a small pop-up box for the purpose of authentication; the bogus box floats over the "real" institutional site's authentication web page so that the user is not likely to suspect anything abnormal. Even the fake URL has a name that seems legitimate in the context of the email as it includes words like “verify”. This pop-up window in fact has no title bar to identify it, has an SSL padlock (that is only a .gif file), and displays words about using 128-bit encryption.
Superficially it would be hard for even a skeptic doing a few checks to see the spoof. When the user enters his authentication data, the page sends the information back to the spoofing site. Phishing complete. The perpetrator has gained information that will allow him financial gain or access to information of value to him. The data collected is all usable without any expensive filtering or data mining techniques. The institution has no way to know directly that the spoof is occurring.

This type of attack is the latest in a long history of social engineering attacks. As Bruce Schneier is fond of saying, amateurs attack systems, professionals attack people. Its success is not dependent on any vulnerability in the end user’s software. People are part of any system they use and their vulnerability to social engineering is often the weakest link. Interfaces are always likely spots for intrusion but the one between man and machine is particularly susceptible because of basic incompatibility.

**Phishing With Trojan Horses**

Trojan horses, like screen scrapers and key loggers\(^1\), require control over the target computer. Many Trojan horses rely on social engineering to accomplish this. This can take the form of email attachments that the user is tricked into executing or can be included with other software that the user is persuaded they want to download and execute (along with adware and spyware). Software exploits can also be used in conjunction with social engineering: the user is persuaded to follow an email link to a site that exploits a known vulnerability in order to introduce a Trojan horse. Accepting a license agreement alone can allow for the downloading and installation of malware. Trojan horses by themselves are far more difficult to craft than a pure social engineering spoofing attack and take a lot more time to become effective, since they must first propagate themselves, install themselves and wait for the user to provide the correct trigger for activation. In addition, the filtering issue associated with Trojan horses for the attacker is huge. The data collected is not pure information when it is received. The attacker must sort through possibly megabytes of data to find the information being sought.

Given the complexity of the code in operating systems, browsers, web servers, and applications, it is not surprising that vulnerabilities exist. These vulnerabilities lead to worms, viruses, denial of service, and in this most recent case, to the acquiring of personal authentication information. Without processes in place (such as the Capability Maturity Modeling) during the development cycle for software, this situation is unlikely to be abated. Commercial pressures do not normally allow for the time commitment required by these processes.

Even with this caveat, discovering these vulnerabilities and making use of them has been limited mostly to benign worms that are more likely an expensive nuisance to clean up rather than a real threat leading to the long term loss of large sums of money. It is in the vendor

---

\(^1\) In reality, often these are not usually attacks directly on the hardware (keyboard or screen) but rather focus on collecting data in Help Objects that do the data collection work needed for the legitimate request. Screen scrapes are more often recursive walks through windows to identify the one formatted to accept a password.
community’s interest to be watchful and make repairs where software vulnerabilities are identified. Blocking the offending IP address from the Internet is not easy to do and requires process and time to accomplish. But this can be, and is, done to eliminate the source in identified cases.

What Can (and Can’t) Be Done?

The solution to either of these attacks rests on setting and accepting at least three expectations:

- Absolute security is a myth; only relative security is achievable. This means that we layer our protection, make differential changes when they are cost effective, and thereby achieve an increased level of security (reduced level of fraud).
- No solution is permanent. Security is a process and planning for it to be a process is equally important.
- The cost of the solution path must be in line with the likelihood of loss through the vulnerability being addressed.

To minimize the effect of attacks that are based purely on Trojan horses is difficult in an environment where the potential gain is high, coding is easy and positive control over the user’s computer is impossible. Fortunately these attacks are less prevalent because social engineering is currently easier and more effective.

Given the nature of phishing attacks it is understandable that experts like Peter Tippett\(^2\) say that the use of SSL and long complicated passwords (among other techniques) add little to the security profile\(^3\) of a corporation or process. Employing these techniques just because they defend against known vulnerabilities fails to consider whether the REAL risks faced by a corporation are actually abated by these approaches. The key is to evaluate and weigh risk. What is the source of the real risk, the highest potential for harm and loss? For example today, social engineering risk far outweighs the risk of session interception in a switched network. This is part of the basis for Tippett’s remarks above. Add to this that the SSL padlock can be attached to a fake web page in the form of a .gif file, quickly eroding the trust associated with SSL, as our initial example described.

Where transaction loss, whether financial, informational, or life, is very high, expensive authentication mechanisms can be put into place to protect against this site-impersonation style of attack. Hardware tokens used with protocols that rely on one-time authentication data are in this category. However, hardware tokens for heterogeneous user groups using uncontrolled computers could be very expensive to install, manage, and protect long term.

---

\(^2\) CTO for TrueSecure Corp. Peter is considered to be the father of virus protection software.

\(^3\) See [http://infosecuritymagazine.techtarget.com/articles/june01/columns_executive_view.shtml](http://infosecuritymagazine.techtarget.com/articles/june01/columns_executive_view.shtml)
Remedial Action

This proposed solution recognizes that in large consumer-based user groups the solution set is different because of cost, usability and manageability issues. The strategy we propose here does not eliminate the problem but reduces risk and fraud to more acceptable levels by taking steps to:

- raise the bar for the attacker – make spoofing more work
- give the target institution a way to know an attack is underway
- shorten the life of the attack
- educate the user to be more active in the solution but at the same time consider his ability to comply with any policy or remediation you implement.

Steps

1. Use two authentication tokens: These could even be presented separately (although the security is lower if this is done). The first could provide limited authority (like viewing account balances, or moving money between accounts) while the second would be required for activities that culminated in the commitment of funds (such as authorizing payments or credit card charges).

2. Make one of the authentication sessions interactive: If the session requires user-specific information from the site, this raises the bar for the attacker. Generic attacks will not work. This also requires contact to the real server by the attacker, which could provide the real site with the knowledge that an attack is taking place.

3. To authenticate the site to the user, present user-specific data between the two levels of user authentication. This again raises the bar on the difficulty of the attack and gives more assurance to the user before full access is gained. It also necessitates a second interaction by the attacker with the true site.

4. Create a blacklist for offending servers: The use of user-specific data and an interactive authentication each necessitate communication between the real server and the spoof server. This makes it possible for the real server to use a blacklist. Building a blacklist would also require 1) an authentication mechanism that allows users to quickly discern that they are being spoofed and 2) a reporting mechanism. Rather than going through the onerous process required to close down an IP address, it could simply be blacklisted. This process is successful with spam blocking collaborative efforts. Some day it might even be possible for the browsers to check the blacklist!

5. Temporarily suspend the use of email as a marketing tool for financial and security topics so that customers receive no mixed messages. Begin an education campaign. Short simple messages to help users become more aware.

Passfaces™ – The Solution Maker

Passfaces, by Real User, plays nicely into the above solution set. Passfaces provide a scalable, cost effective, and reliable first or second level authenticator, along with the password. Because the Passface authentication process is interactive (user configuration data must be obtained), the bar is immediately raised for the attacker. The user cannot passively supply the correct token
because he has memorized nothing but rather must recognize self-prompting grids of faces in order to be authenticated. Although user’s Passfaces are randomly assigned from a library of faces, even more defense can be built by utilizing multiple libraries across a given population. The attacker would then need to come to the authentic site for the correct configuration data for each user. If the spoofer was content to trap, say, one in five persons by presenting a single library to all users, then four out of five users would know they were being spoofed and could report the activity. Together with a blacklist functionality, either way, the attack could be stopped before serious loss was suffered. The ability to change libraries also allows for a solution that can migrate. And finally, using Passfaces as the second factor places no additional burden on the user. There is no limit to the number of faces a person can remember once they are familiarized with them for this use. The hard part of the process is shifted to the server and the user is left to do only what he is innately good at – recognizing human faces. Passfaces are cost effective, reliable and scalable for large heterogeneous user groups without any changes in infrastructure. Because social engineering remains such a successful exploitation mechanism, using a technology designed to counter this threat dramatically improves the security profile of network or application access.

Conclusion

With the growing sophistication of hackers, complete elimination/prevention of the effects of phishing and Trojan horse attacks is likely impossible, or at the very least, cost prohibitive. But by taking a differential security approach, risk can be reduced by layering less costly approaches and thereby reducing the risk, not to zero, but to an acceptable level. By using a technology such as Passfaces that raises the bar on the attack components, by monitoring the resulting inbound requests more rigorously, and by instituting a blacklist capability, the security profile of an application can be raised to acceptable levels at a much lower cost.

About Real User

Real User Corporation is a resident of the Chesapeake Innovation Center in Annapolis Md. Real User’s Passfaces™ technology, patented worldwide, is a cognometric method of personal authentication, based on the measurement of the innate cognitive ability to recognize faces. Our products offer businesses, government agencies and OEMs a uniquely strong, reliable and cost-effective authentication solution that can be rapidly deployed to improve security and usability and generate an immediate return on investment.

For more information about the above PKI strategy, the science behind Passfaces, or the line of products, visit us at www.realuser.com or contact:

Patricia Lareau
V P Security Solutions
Tel: 805.544.1138,
Cell: 805.748.1034
pat@realuser.com
A Phish Story

Passfaces - The password you will never forget