



TOOLS AND TECHNIQUES

Active Cookies—A New Batch of Soft Tokens

How do you stop pharming? There's DNSSEC—which essentially mandates digital signatures for all Internet communications—but that calls for a major overhaul of the Internet and does nothing to help us now. Truth is, no one has a realistic way to unequivocally prevent pharming, but researchers at Indiana University and RavenWhite Inc.—a start-up that aims to secure online identity with unobtrusive tools—are working on a very promising countermeasure, dubbed “active cookies.”

Active cookies are, in essence, another layer of authentication, but they do not rely on a user's awareness (are in fact, transparent to a user) and are both inexpensive and easy to implement and maintain. They can combat pharming, phishing and other man-in-the-middle attacks and harden online identity management (which becomes increasingly important for regaining diminishing user trust and complying with stricter security regulations).

An active cookie does not prevent DNS cache poisoning. It doesn't prevent the user from being tricked by a spoofed site. It doesn't prevent an attacker from harvesting personal information.

What it does do is prevent an attacker from impersonating a user on the legitimate site, and helps a server learn that it's being spoofed.

What is it?

The researchers define an active cookie as “a piece of cached and sandboxed executable code, such as a JavaScript object, that helps authenticate an Internet browser to a server.”

So when you visit the legitimate site for the first time, it caches the cookie in your browser like usual. The cookie contains an ID for your browser, a private cryptographic key and a trustworthy IP address for the server.

The next time you visit, the server asks the cookie for your ID, which it then uses to look up your key and send back a cryptographic challenge. The cookie encrypts the challenge, and—here's the clever bit—sends it back to the server over two channels: the usual HTTP channel and the hard-wired, trustworthy IP address the cookie contains.

Sure, a pharmer might well alter a DNS entry and thereby insert himself into the middle of the HTTP communication, passing the challenge from the server to the browser, and the correct response from the browser to the server. But the cookie will still insist on sending a second response directly from the user's client system to the hardwired IP address (not the bogus one obtained from the poisoned DNS). This transmission will carry the user's IP address, not the IP address that the attacker in the middle has used. So,

The cookie encrypts the challenge, and—here's the clever bit—sends it back to the server over two channels: the usual HTTP channel and the hard-wired, trustworthy IP address the cookie contains.

if the server does not receive the correct response, through both channels and from the same IP address, it will know something is amiss, and will act accordingly—blocking account access, freezing the account, etc. It's a scheme that won't work if the attacker can interrupt the hardwired communication, perhaps by gaining control of the user's local network segment, but in that instance all bets are off in any case.

Markus Jakobsson, one of the authors of the research, offered a physical-world analogy. “Say I'm Chase Manhattan Bank. If I want to give you an active cookie in the physical world, I give you a piece of paper that says please call this number, and then say ‘rutabaga.’ That's the first step,” he said. “Then look up Chase in the phone book, call that number and say ‘gray tomato.’ Pharming basically is like someone replacing your phone book with a phony one, so you'll call a phony number and say ‘gray tomato.’ Chase is expecting two calls, both from the same number. So if the bad guy calls Chase, he may be giving the right password, ‘gray tomato,’ but Chase's CallerID tells them he's calling from the wrong number.”

Jakobsson describes an active cookie as a “soft cryptographic token”; easier and cheaper to implement than hard fobs or biometric readers, and much easier on users. They require insubstantial storage space and have nominal impact on performance.

Limitations

Active cookies have a number of limitations, to which the researchers readily admit.

First, the communication authenticates the browser, not the user, so the benefit is lost when a user switches browsers or machines. The cookies can also be easily purged from browser caches.

Also, the communication between server and cookie is initiated by the server. If the attacker simply wants to glean information, it won't

have any need to initiate communication with the cookie; so the user may pass over personal information with both the user and the cookie none the wiser. The cookie only springs to action when the attacker tries to access the user's account on the legitimate site.

To combat this, Jakobsson suggests an added feature to the active cookie that could clue an aware user that they're visiting a phony site. The cookie could contain an image particular to the ID, which would display only after successful authentication; thus, a savvy user may notice an absent or unfamiliar image.

“After the user has been verified, the server will send an acknowledgement to the active cookie, which will then display a picture,” said Jakobsson. “Maybe yours is an orange rabbit on top of Mt. Fuji and mine is a pig in a dungeon. Most users are going to be absolutely oblivious, but some of them may notice that my pig is there instead of their rabbit, or perhaps there's no rabbit at all.”

There's been no official deployment of active cookies as of yet, but RavenWhite continues experimenting in testbeds. The prototype has been tested mainly on Firefox operating on Windows XP and Mac OS X, but preliminary tests suggest that active cookies will also function in Internet Explorer.

The full whitepaper, “Active Cookies for Browser Authentication” will soon be made available at www.RavenWhite.com. ■

—S.P. and R.R.