Cybersoft.com

CyberSoft White Papers

Security Problems Caused By Aggregate Data Of Client/Server Connections On The Internet

by Peter V. Radatti
CyberSoft, Inc.
1508 Butler Pike
Conshohocken, PA 19428 USA

radatti@cyber.com

URL: www.cyber.com

Web Server Logs

Most people do not realize that when they are browsing on the Internet that they are creating complex bidirectional client server relationships between their individual workstation, their corporate identity, their Internet Service Provider (ISP) and the remote server that they are connecting to. Generally in these relationships the remote server identifies who it is operated by with its content in addition to whatever information is contained in the InterNIC database about the domain and whatever information is in the IP registry. What this means is that it is easy to create remote servers whose ownership is easily hidden. At the same time, most client software such as Microsoft's Internet Explorer and Netscape's' Communicator reveal a good deal about the end user, even if the end user specifically configured the product to reveal incorrect or minimal information.

When a user connects to a Web Server they immediately reveal a good deal about themselves. Some of the information they may reveal are:

1. What URL they were coming from?
2. What URL(s) at the site they visited and how long they visited it?
3. What operating system they are using and consequently what type of computer?
4. What web browser they are using ?
5. The IP or domain address of their workstation?
6. What query they used in a search engine to locate the URL?
7. Internet Cookies ?
8. What day and time they accessed the site?
9. Configuration information about their web browser and system?

Most of these are standard log items from web servers. The careful use of some data reduction and sorting software will reveal a wealth from just these logs. Using some simple techniques it is also possible to learn the user's email address, name, company name and other sundry items. Below is an actual example of some log entries from the http://www.cyber.com web server which is running the Apache web server in a normal configuration. Information in the example has been changed to hide the identity of the users, otherwise this is from a live site with over 60,000 hits per week.

Annotated Web Server Log Examples

Example One:

customer.somedomain.net.il - -

[18/Nov/1998:15:27:20 -0500] "GET /images2/new_title.gif HTTP/1.0" 200 10365

"http://www.cyber.com/" "Mozilla/4.05 [en] (WinNT; I) via

NetCache version NetApp Release 3.2.1R1: Fri Aug 7 11:52:45 PDT 1998"

customer.somedomain.net.il

The user's address.

- - [18/Nov/1998:15:27:20 -0500]

The date and time that the user visited.

"GET /images2/new_title.gif HTTP/1.0" 200 10365 http://www.cyber.com/

What they are looking at.

"Mozilla/4.05 [en] (WinNT; I) via NetCache version NetApp Release 3.2.1R1: Fri Aug 7 11:52:45 PDT 1998"

This user is running Netscape Navigator 4.05(en) on Windows NT and appears to have a http caching program running called NetCache.

Example Two:

cache1.otherdomain.net - - [18/Nov/1998:15:39:33 -0500] "GET /images2/new_title.gif

HTTP/1.1" 200 10365 "http:\\www.cyber.com/" "Mozilla/4.0 (compatible; MSIE 4.01; Windows 95)"

cache1.otherdomain.net

The user's address.

- - [18/Nov/1998:15:39:33 -0500]

The date and time that the user visited.

"GET /images2/new_title.gif HTTP/1.1" 200 10365 "http://www.cyber.com/"

What they are looking at.

"Mozilla/4.0 (compatible; MSIE 4.01; Windows 95)"

This user is running Microsoft Internet Explorer 4.01 on Windows 95.

Example Three:

cache-dc03.proxy.aol.com - - [18/Nov/1998:15:48:40 -0500] "GET /personal/pete/specs.html
HTTP/1.0" 200 1888 "http://netfind.aol.com/search.gw?search=www.cyber.com&lk=
excite_netfind2_us&nrm=n&pri=on&xls=b&xll=40&test=Find%21" "
Mozilla/4.0 (compatible; MSIE 4.01; AOL 4.0; Windows 98)"

cache-dc03.proxy.aol.com

The request came from here.

- - [18/Nov/1998:15:48:40 -0500]

The date that this data was requested.

"GET /personal/pete/specs.html HTTP/1.0" 200 1888 "http://netfind.aol.com/search.gw?
search=www.cyber.com&lk=
excite_netfind2_us&nrm=n&pri=on&xls=b&xll=40&test=Find%21"

This user was reading our page from a search engine.

"Mozilla/4.0 (compatible; MSIE 4.01; AOL 4.0; Windows 98)"

This user has a Windows 98 machine and accesses the Internet via America Online
ver4.0 (which comes with Microsoft Internet Explorer 4.01 as the default web browser).

Honey Pot Servers

A method of collecting intelligence on the Internet that is well focused and targeted is to create Honey Pot Web Servers. Just as honey attracts bees, Honey Pot Web Servers contain selected information that will attract the target audience. For example, if you are interested in learning what companies are doing research on recombinant molecular DNA and you expect that this type of information will generally be a closely held corporate secret then a very good method of accomplishing this task is to create a Honey Pot Web Server on that topic. Interestingly, not only will you be able to tell which companies are visiting your web site but you should be able to learn who within that company is doing the research. In addition, by careful analysis of the server logs you may be able to learn which specific subsets of information the person is interested in and thereby reveal specifically what area of research and development the visitor is working in. This of course is a valuable piece of information that may help the server owner steer their research into the same areas as their competitors.

A secondary benefit of controlling a Honey Pot Web Server on a specific topic is that you may be able to solicit technical white papers on the focus topic. As the System Administrator for the site you could have up to a couple of weeks in which to use the paper for your own purposes prior to the actual posting. This is in addition to controlling a valuable topic site on the Internet. This control allows you to decide what to post on the site. If the site becomes trusted then its value as a source of disinformation and the resulting wasted resources of your competitors becomes priceless.

Finally, whoever controls the talent in a given field controls the field. Most of the time the people who you most want to hire are already working and locating them will be difficult. This is not true for the Honey Pot Web Site owner. They will can gather information on many potential employees including their email address. Once they have the email address they may be able to use standard Internet search engines or even finger processes to locate the person's full name, phone number and potentially their home address.

Of course, in order to facilitate this you will need to hide your identity. Fortunately, there are many Internet Service Providers who will host a web site for you and hide your identity as a by product of the process.

How can you fight a Honey Pot Web Server. Actually there are several ways. Using a proxy based firewall that is configured to hide the actual user and reject specific types of connections (such as finger deamon requests) can go a long way alone. Other things that can be done is to insure users do not configure their web browser in such a way as to reveal who they are. If you are only browsing then there is no reason to configure the email part of the web browser or you can configure it in such a way as to provide false information. Using the standard Unix sendmail aliases file can also be used to create false email addresses that can still accept and forward email. This is valuable because the false email address will not be associated with the person or company using it. An example of how this may be done is if company A controls the Internet domain names A.com and B.com. They can have their web server configured to use A.com while all of their browsers could be configured to reveal B.com. The email address xyzzy@b.com will not reveal that the actual user is radatti@a.com and since the

person xyzzy does not exist no amount of Internet or phone book searches will reveal any information.

Example of an /etc/aliases Configuration File With Comments

Aliases can have any mix of upper and lower case on the left-hand side,
but the right-hand side should be proper case (usually lower)
@(#)aliases 1.10 89/01/20 SMI
##

#######################
Local aliases below
#######################

xyzzy:hiddengateway!pete?Here we use direct addressing to a hidden system
sysop:root@a.com???Here we redirect to the internal domain
info:support@b.com??A public account stays on the known domain
marketing:harry@b.com

Internet Cookies

Internet cookies as a computer technology sound safe, slightly boring and maybe even tasty. This paper will attempt to demonstrate that Internet cookies are actually mud pies with a good deal less safety and tastiness than you would have eating mud. First, you need to understand what an Internet Cookie is. If you remember old cowboy movies there was always a seen where cows were branded. An Internet cookie is the same thing except it is you that is being branded. If you are using Netscape, the browser arrives on your computer with a default of accepting cookies silently. That is, you never feel or even know that someone just smoked your hide. As a matter of good security policy I turned silent acceptance of cookies off. There is no option to turn off acceptance completely so every time a cookie request is made my system, a pop-up message window appears. The message window gives me the option of accepting the cookie and being branded or canceling the cookie. Since most people don't know what a cookie is, don't understand that there is any security issues in accepting them and may in fact be afraid of breaking something by press the button labeled "cancel" I assume that most people accept cookies. In fact they would never even know that they were being "cookied" unless they stumbled upon the button that disables automatic acceptance.

At this point in my paper I feel safe in the given assumption that most people are accepting cookies. So what is the big danger? The military knows. As long as there has been warfare, militaries have been concerned by something called aggregate data. Aggregate data may be as simple as counting the number of cars that enter the gates at a military reserve. If there is someone counting the number of cars entering a few dozen reserves across the country over a period of time then anyone who has access to the data from all of the reserves could in fact predict a major military engagement about to start. Simply put, if the number of cars entering all of the reserves had a sudden jump across the country and the people who entered didn't leave then conclusion is simple. They are about to go somewhere else, in mass. The same types of analysis can be done with your movements. There are

now large networks of Internet cookie data collection companies. They keep track of where you are, where you came from, where you went to, the kind of computer, browser and operating system you are using. In fact, they can also get your IP address, system name and if configured, your name, company name and email address. That is a lot of information about you in a single gulp but it is not the end. At some point you will come across a form or your will order something over the Internet. When you do, your real name, home address, telephone number, credit card number and anything else you tell them about yourself is now available to connect with your cookie. The interesting thing is that if the company kept all your old cookie information then they can track your past, present and future. This could be dangerous if you accidentally end up at an embarrassing web site.

So why does anyone try to brand you with cookies? The reason is simple, effective advertising. In fact, I feel that advertising is a useful thing since it helps me find things that I want to buy. The problem is that a billboard doesn't know who is looking at it but a computer does. If I were a member of a vegetarian household and I suddenly started receiving email, banner advertisements, postal mail and phone calls from meat producers that could be a real problem. At sometime in the past I might have bought a book from an on-line bookstore. I already had a cookie so a relationship now exists between myself as a person and my cookie. The cookie is issued every time I enter one of the cookie networks and they target advertising to me based upon my movements. Very quickly they know more about me than I do. I turned cookies on for a while and started looking for travel information at the Alta Vista search engine. They are part of a cookie gathering network. So is the web site devoted to the Dilbert cartoon strip as are many other sites. As soon as I did my first search on "airfare and Boston" I was presented with advertisements for travel agents. When I traveled to other cookie affiliated sites I received more travel related advertisements. This is fine but think about the implications. If I browsed several finical oriented sites I might start receiving unsolicited and unwelcome attention from sleazy stock brokers. If I searched for medical information I don't want anyone to know what my problems are. It is none of their business. If my doctor or stock broker shared that type of information about me I would have them in front of their respective state boards for unsavory behavior. The fact of the matter is a cookie tracker could learn my medial problems, hobbies, finical interests and a whole lot more depending upon what I did on the Internet. This is an invasion of privacy but legal.

One possible solution is to shut off automatic silent acceptance of cookies and just press the cancel button. It appears that the cookie monsters already thought of that. They have gotten pushy and rude. There are now many sites that enforce cookie branding by plastering you with dozens of cookie requests per page. Some of them plastered me with so many cookie requests per page that I lost count after 20. The message windows appear faster than I can cancel them, get in the way of what I wanted to do and waste my time. How rude! Department stores don't keep me out just because I refuse their "free" credit card and gift at the door. I don't mind one cookie request because I have the option of saying no but dozens of "requests" feels like getting mugged.

So how can you deal with cookies? Actually it's easy. Turn on silent acceptance of cookies. Enter the ".netscape" directory and delete the file named "COOKIE". There are all kinds of dire warning not to edit or delete the file but I do it anyway. Unfortunately, Netscape keeps recreating the cookie file and I

have to keep deleting it. On the UNIX computer that I use to browse the web I could put the "rm /export/home/radatti/.netscape/COOKIE" in my ".login" and ".logout" files but I found a better way. From your home directory enter the ".netscape" directory. Remove the COOKIE file and put in a logical line to "/dev/null" (ln -s /dev/null COOKIE). As fast as the web browser creates new cookies the UNIX system throws them away. I no longer get bothered with pop-up windows and I clog the cookie monster with hundreds of fake identities per day. In fact, as far as the cookie trackers go they must think that 80 different people visit each page without finishing to download the page.

Cookies are only one way for people to gather aggregate data on you when using the Internet. In addition cookies are not restricted to Netscape, Microsoft Explore also processes cookies. Finally, your Internet service provider can gather all of this information and a great deal more about you. It's a dangerous world.

Trojan Horses and Covert Communications Channels

Covert communications channels can continue to provide intelligence gathering long after a specific client/server connection has been broken. There now exists Internet web server based Trojan horses that can access private information stored in your computer. One of the most well known but still new attacks is known as the Cache Cow Trojan. The Cache Cow program allowed Web Servers to extract a client's cookies. This could have dangerous results since some financial institutions use cookies as part of the authentication process. Potentially, a Web Server which downloaded cookies for an individuals stock trading account could then attempt to pose as the legitimate account owner and move stock to their own accounts. The Cache Cow problem was corrected in Netscape 4.07.

The Son of Cache-Cow is a set of Trojan Horse programs that continues to work in all versions of Netscape including version 4.07. These programs are called "Cookie Monster", "File List" and "Cache-Cow 4.07". The Cookie Monster program steals cookies from arbitrary locations. The File List program steals the contents of local directories. I tried this program pointing it to my home directory and then pointing it to the root directory of my Unix workstation. It correctly showed me a complete file list! The Cache-cow-4.07 program will steal the contents of the browser cache. It has the same effect as the original Cache-Cow program.

References:

"Cookie Monster, The risks of Internet Cookies and Aggregate Data"

By Peter V. Radatti, January 1998

Web Server Log extracted from http://www.cyber.com

MSNBC Web Site information on Cache Cow.

Dan Brumleve's Web Site on Son of Cache-Cow.

CYBER.COM™ , VFIND™ and AVATAR™ are registered trademarks of CyberSoft, Inc. CIT™ , THD™ , UAD™ , MVFILTER™, JDIS™, ROBOTMODE™ , NTI™ , NTI-CRYPTO™ and RMI™ are trademarks of CyberSoft, Inc. Documentum™ is a registered trademark of Documentum, Inc. All other trademarks and copyrights are the property of their respective holders.
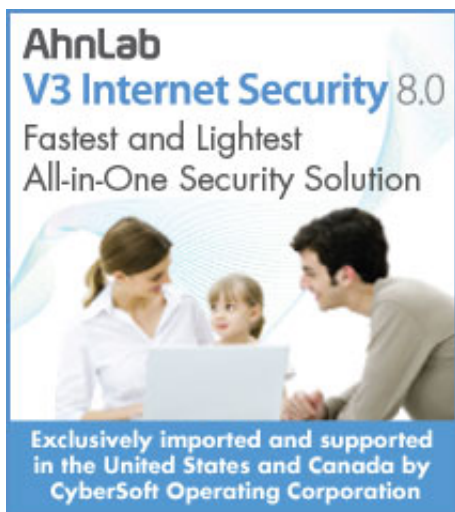
Home | Products | Support | Purchase | Contact | News | About

This site certified 508 Compliant