Global Information Assurance Certification Paper

GSEC version 1.2d
Big Brother is Watching. An update on web bugs
Steve Nichols
July 3, 2001

What are web bugs?
Web bugs are tracking devices embedded in web pages, executables or scripts that secretly monitor your activity on the web and send the information back to a 3rd party. According to a report published by Intelytics, there were close to 16 million web pages that had some type of web bug out of 51 million web pages that were scanned. In an online article in Smart Computing Magazine, Andrew Rodgers described four general types of web bugs, GIF (Graphics Interchange Format), executable, script based, and application. The first type, which is the most common, are also called beacon GIFs, tracker GIFs, clear GIFs, and invisible GIFs. Web bugs can be used in web pages, newsgroups, email and programs that allow images to be retrieved from the internet including the MS Word, Excel and PowerPoint.

What information do they track?
The typical clear GIF web bug "phones home" with: the user's IP address, the time it was displayed, the user's browser type, the URL of the host site, the URL of the image on the hidden web site, and previously set cookie values. Cookies can contain personal information input into a web form by the user. Some types of script and executable web bugs can retrieve almost any information that the programmer wishes to obtain from the user's computer. Application web bugs can retrieve a User's IP address, hostname, version of IE, OS name and version, and web browser cookie information.

How do they work?
Clear GIF web bugs use a 1x1 pixel GIF image that is planted in the web page of host web site. It can be seen as an HTML IMG tag when the source of the web page is viewed and will typically have a URL that is different than the rest of the page. It is important to note that not all 1x1 GIFs are web bugs because some small GIFs are used for alignment purposes. When the web bug image is fetched by an HTTP GET query, it sends information about the user to the hidden website. When the server responds, it sets a cookie on the user's machine that is sent back with any other requests to the same domain. Each GIF web bug acts like a trip wire on the Internet. When a user visits a site with a web bug, the web bug sends information back to the hidden web site that the intended website has been visited. The web bug will send back information in the cookie if the cookie was set using the same domain. The web bug would be of limited value if only a few web bugs were placed on the net. However, a wide range of bugged web sites provides a third party with enough information to profile a user's surfing habits.

In general, this is how a web session would work for a clear GIF web bug. User A visits web site B that contains a web bug. The web bug phones home to web site C without the user’s knowledge. The bug carries with it information about the user and the visited

web site. The third party web site (site C) collects information on user A as he encounters other web pages with site C web bugs.

Clear GIF bugs can infect email messages in the same way they infect web pages. A 1x1 pixel image is placed into the body of the email. The target email address is included in the web bug URL. When the message is opened, the bug phones home and takes selected information with it. Clear GIFs can also be used in newsgroups when Outlook or Netscape are used to read HTML newsgroup messages.

An example of an email web bug would start with site C sending out a bugged email to a list of possible recipients. Whenever user A opens the email, the embedded web bug image is fetched and sends the user's email address, IP address, and cookie information back to site C.

Executable web bugs are also called Trojans because they perform operations not intended by the user. They infect your system similar to a computer virus and can do a variety of tasks including: monitoring traffic, tracking web sites, and forwarding the information back to a hidden web site. Two common ways that an executable web bug can infect a system is by running an executable program from an email attachment or by running downloaded software. They can require various degrees of user action to install. One level requires that the user actively installs a program or opens an attachment. A second level requires only that the user browse a web site which triggers an executable without the user's knowledge. An example of a Trojan would be where a user runs an executable file that tracks and stores the user's movements on the web and then uploads them to a hidden web site. Intelytics demonstrated an example of the second level of executable bug to a group of US Congressmen in March by stealing an email address book off a system that had just visited a bugged website.

Script based web bugs rely on ActiveX, JavaScript, Perl, Java or other script language to do their dirty work. Typically, browsers are configured to process the scripts automatically, but scripting can be turned off. Exploits that abuse the properties of frames are considered "cross-frame scripting". They allow a third party to track your web activity after a bugged frame on a web page has been visited. An example of a script based bug can be seen when a user initiates a Javascript that collects information about the users browsing habits in another window or frame.

Application web bugs infect applications, which allow images to be linked to the Internet. They are also called document web bugs even though they can infect more than documents. "Any program that allows images to be retrieved from the Internet could involve web bugs" [1]. Document web bugs contain the URL of the web bug and try to retrieve the linked image whenever the document is opened. In some word processing applications, the image itself is not saved in the application in order to save space. An application web bug example would start with site C embedding a bug into a document, spreadsheet, presentation file, or some other web-enabled application. When user A views the object, the web bug fetches the image and notifies site C that

the object has been viewed and when it was viewed. It also returns the IP address and host name that opened it along with any cookie information it can access.

Who uses them?
Advertising agencies are the primary web bug recipients. Online ad agencies pay to place advertisements on a network of websites. In order to maximize their return on investment, advertisers want to target their ads to those who will most likely purchase the product or service. For the ad to be most effective, the agency needs to know something about the online user, their interests, preferences and purchasing habits. One way to collect the information is to ask for it when a user visits the site or makes a purchase. The information can then be stored as a cookie. Another way is to discreetly monitor where the user surfs and relay that information back to a database. If the user enters information at one site, then some of the web bugs from the same domain can capture that information. The information in the user's profile can determine what ad is displayed to the user when he visits a web site.

The US federal government is also guilty of using web bugs even though tracking devices were prohibited on government web sites last year by the Office of Management and Budget. According to a report released June 15, 2001 Inspectors General found 42 web bugs on government agency sites. Senator Fred Thompson, the co-author of the legislation that requested the report, expressed concern that the report was not complete because only a small number of the total agency government agency web sites were included in the review. While checking out the Department of Commerce's policy on web tracking devices, I found it ironic that their site contained a web bug to <http://cgibin.erols.com> .

Web bugs have become so widespread that web bug reports can be obtained off the Internet. The web site <http://www.Securityspace.com> lists two different web bug reports. The web bug site count report shows the top 100 sites that benefit from web bugs. The report also lists the different types of web bugs each site uses including: img, iframe, a, frame, input, script, and ilayer. The web bug traffic count report shows the top 100 web sites that benefit from web bugs by assigning a relative traffic weight to each site based on the amount other sites that link to it.

Why are they used?
Web bugs are used by companies for a variety of reasons. Among the list of uses they: provide an independent tracking of web site hits, track web pages a visitor views within a website, track what web pages that are visited across different web sites, track search strings from search engines, transmit cookie information to a hidden site that can identify individuals, identify and capture the type of internet browser used by the visitor, and transfer demographic data that was previously entered by the user. They can "Match a purchase to a banner add that a person viewed before making the purchase. The web site that displayed the banner ad is typically given a percentage of the sale" [2]. Web bugs are a tool used by companies to determine their target audience and maximize their advertising investment by finding out all they can about a user's preferences and purchasing patterns. Many companies and apparently government

agencies who maintain websites can find out the information they want by using web bugs in their arsenal of data collection.

Document web bugs can be used to track when and who opened a document. Firms wanting to track newsletters or other important documents could plant web bugs into documents. Each time a document is opened, it fetches the web bug image and reports back to the hidden website user information.

Web bugs in email could be used to track who and when an email was read, along with the IP address. Marketing companies use web bugs to determine who has read an email and then remove the ones who did not read the message. People simply opening the email cause the web bug to fetch the image which notifies the sender that the address can be used for future email campaigns. Several well known companies have used web bugs in email marketing campaigns including Microsoft, Barnes and Noble and eToys.

Web bugs in newsgroups messages can also be used to monitor who is reading a particular newsgroup.

New detection tools.
A research team from the University of Denver has recently developed a web bug detector call Bugnosis that alerts the user to hidden web bugs in web pages. Bugnosis is a browser extension for Internet Explorer 5 and above and only detects image bugs in web pages. It can be downloaded free at <www.bugnosis.org> . While it does not prevent web bugs, it visually and audibly alerts the user when they appear. It also displays contacts for some of the known web bug distributors. The tool can display the properties about the image file indicating why it determined it to be a web bug or why it is suspicious. Items in the analysis include: the size of the GIF, if multiple URLs are present, the length of the URL, protocols used, if the image comes from another domain, if it manipulates a third party cookie, and if it recognized the URL in its database of known web sties. With the tool loaded, it was interesting to note that some news web sites that reported on the dangers of web bugs were themselves bugging their web pages.

Intelytics also offers a web bug tool at <www.intelytics.com> that detects and cleans web bugs. Their tool called Personal Sentinel works on most windows operating systems. It has configurable filters along with the ability to alert the user of changes to the startup folder and registry. It graphically displays the level of privacy risk. It has the ability to clean up web bugs with manual intervention from the user. With Bugnois also running on the same machine it appeared that Personal Sentinel was able to detect bugs that did not use clear GIFs.

Other ways to protect yourself.
One defense against web bugs is to turn off cookies to untrusted sites in your browser's configuration. Turning off cookies will not prevent web bugs, but it will limit the amount of information they can pass back to the hidden site. Another way to block

clear GIF web bugs is to block all advertisements. Advertisement blockers include products like Guidescope, Webwasher or Adsubtract. Personal firewalls like Zone alarm can be used for protecting against web enabled application web bugs. When a document web bug
tries to fetch an image from the network, Zone Alarm can notify the user that the application is trying to access the Internet. To protect against executable web bugs, make sure any executable program that is run is from a reputable source. Be cautious when opening email attachments as they can contain hidden executable code. Script based web bug risks can be reduced by changing your browser preferences to disable scripting, Java, JavaScript and ActiveX parameters. Be aware that disabling some of these features may prevent web pages from displaying properly. In addition some sites require cookies to function properly.

Malicious use.
In an article entitled "Fun with Internet Bugs", Thomas Greene invited his readers to submit malicious uses of a GIF web bug. One submission he received described how a web bug could be used to imbed a porn picture. The picture could be reduced to a 1x1 pixel and then sent in an email. If the user happens to work at a company that checks for porn sites, he would be implicated for inappropriate web use when he opens the email and the web bug image is fetched from the porn site. The innocent user would not know why he is implicated because the image that was retrieved from the website was so small that he never saw it.

Another potential exploit sent in to Thomas Greene involved a bugged Usenet message. The embedded IMG tag would launch a CGI script that scans for open SMB shares for the purpose of installing BackOrifice2000.

Summary
The majority of web bugs are not persistent in the sense that once a user has visited a bugged site they are then tracked at every site they visit. A network of web bugs on many different sites act as trip wires to report back to the hidden web site what websites have been visited. A company interested in tracking user's web site habits plants a network of bugs on various sites. All of the bugs will phone home to the same location when tripped by the user. Each of these bugs can read cookies on the user's computer (if cookies are enabled). With the cookie and website information, the company can create a profile on the user's surfing habits and then target their ads for the user's specific interests. Email marketers use Web bugs in email to valid email addresses and to determine who read the message. Application web bugs can track confidential documents, newsletters and even spreadsheets. Script bugs can be more powerful than GIF bugs and can bypass some bug detectors. The most dangerous web bugs do not rely on images but on executables. These web bugs can bypass browser preferences and perform almost any task they are programmed to do.

The risks have to be weighed against functionality because many of the methods for reducing the risks of infection hinder functionality. Users can reduce their risk of privacy invasion from web bugs by turning off all scripting, turning off cookies, loading

bug detector/cleaners, installing ad blocking software and installing personal firewalls. However, all the measures are pointless if your computer is unusable. It is probable that more bug detectors will become available to the public to make it easier to protect user privacy. However, as more tools become available for preventing web bugs, marketers will be searching for new ways to obtain data from online users.

End notes:
1 FAQ: Document Web Bugs.

2 Bugnosis: Web bug FAQ.

References:
"A glossary of digital privacy terms." Privacy Foundation
http://www.privacyfoundation.org/resources/glossary.asp (29 Jun 2001)

"Bugnosis: Web bug FAQ." http://www.bugnosis.org/faq.html (29 Jun 2001)

Craig, Kimberly. "Web bugs." October 2000
http://www.sans.org/infosecFAQ/covertchannels/bugs.htm (3 Jul 2001)

"CYBERSPACE AND THE LAW." 23 Oct 2000.
http://www.nd.edu/~pbellia/cookieshand.pdf (1 Jul 2001)

"FAQ: Document Web Bugs." Privacy Foundation.
http://www.privacyfoundation.org/resources/docbug.asp (29 Jun 2001)

Farrow, Rik. "Big Brother's Sneaky Browser Tricks." Interesting overview of the darker side of Web Surfing. 3 Dec 2000. http://www.mumbai-central.com/nukkad/dec2000/msg00016.html (1 Jul 2001)

Festa, Paul and Barnes Cecily. "Word documents susceptible to "web bug" infestation." 30 Aug 2000. http://news.cnet.com/news/0-1005-200-2652562.html (3 Jul 2001)

Geene, Thomas C. "Fun with Internet Bugs," 12 Dec 2000.
http://www.theregister.co.uk/content/6/15423.html (30 Jun 2001)

Krebs, Brian. "Government Sites Still track Visitors," 19 Jun 2001.
http://www.computeruser.com/news/01/06/19/news2.html (30 Jun 2001)

Lew, Jacob. "Memorandum for the Heads of Executive Department and Agencies." Office of Management and Budget. M-00-13. 22 Jun 2000.
http://whitehouse.gov/omb/memoranda/m00-13.html (29 Jun 2001)

Olsen, Stephanie. "Nearly undetectable tracking device raises concern." 12 Jul

2000. http://news.cnet.com/news/0-1007-200-2247960.html?tag=st.int.3761-7-2426166 (3 Jul 2001)

Olsen, Stefanie. “New tools hatch for sniffing out Web bugs.” 5 Mar 2001. http://news.cnet.com/news/0-1005-200-5008849.html?tag=tp_pr (3 Jul 2001)

“Policy 4: Policy on the use of Web bugs.” 24 Apr 2001 http://www.doc.gov/webresources/Printable.html (3 Jul 2001)

Rodgers, Andrew. The Biggest Privacy Threat You’ve Never Heard About.” Jul 2001. http://www.smartcomputing.com/editorial/article.asp?article=articles%2F2001%2Fs1207%2F35s07%2F35s07%2Easp Checking For Bugs (3 Jul 2001)

Smith, Richard. "Bugnosis." 7 Jun 2001. http://www.privacyfoundation.org/commentary/tipsheet.asp?id=43&action=0 (1 Jul 2001)

"Surf Safely: Protect Private Information." http://home.cnet.com/internet/0-3761-7-2426166.html?tag=st.int.3761-7-24261 (29 Jun 2001)

”Web bug site count report." 1 Jul 2001. http://www.securityspace.com/s_survey/data/man.200106/webbug_site.html (1 Jul 2001)

”Web bug traffic count report." 1 Jul 2001. http://www.securityspace.com/s_survey/data/man.200106/webbug_traffic.html (1 Jul 2001)

Whalen, David. "The Unofficial Cookie FAQ." Version 2.54 http://www.cookiecentral.com/faq/#3.2 (3 Jul 2001)

“What are the four types of web bugs?” http://www.intelytics.com/webbugtypes.html (3 Jul 2001)

Last Updated: July 26th, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event