

# How Black-hatters Artificially Inflate their Alexa Ranking

## Overview

As sure as rain will fall in Seattle, you'll find that not long after a legitimate internet technology is developed, some enterprising soul will develop a method to abuse it. This article aims to address one commonly employed technique to artificially increase Alexa rankings.

## Background

Alexa Internet is a subsidiary of Amazon.com that collects information related to the traffic patterns of individuals who have installed the Alexa toolbar. Much discussion has occurred regarding the relevancy of results offered by Alexa. For example, it's important to note that the assumptions reached by Alexa are not representative of the internet population as a whole, but rather only of those individuals whom have installed the Alexa toolbar, be it by hook, or crook. Further, there is no reliable information to suggest what percentage of the of the web-using population actually uses browsers with toolbar installed. In addition, the toolbar itself is only offered in English, and only for Internet Explorer and Firefox (since July 2007) browsers, and only on the Microsoft Windows platform. Computers and workstations used in the workplace, college laboratories, and internet cafes are often highly managed, and the alexa toolbar, if ever installed, often doesn't last long.

Despite this, it's not uncommon for website operators to reference Alexa rankings when evaluating their own websites, or the websites of prospective or current clients.

In the course of this article, I am using version 7.2 of the Alexa toolbar, the most recent available. For the packet analysis, I used Wireshark v0.99.8, a freely available and open-source network protocol analyzer. For more information about Wireshark, consult their website. [Technical Analysis](#)

Every time a request for a webpage is made, the toolbar sends a parallel request to Alexa's server (data.alex.com) to retrieve stored information about the site in question. Below is an example of such a request:

```
GET /data/j6HV718Dy0g2GJ?cli=10&dat=snba&ver=7.2&cdt=alx_vw=20&wid=23976&act=000000000000
&ss=1536x960&bw=749&t=0&tll=4000&vis=1&rq=6&url=http://www.cybernac.com/ HTTP/1.1
Accept: */*
UA-CPU: x86
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.50727; Avalon
6.0.5070; InfoPath.2; Alexa Toolbar)
Host: data.alex.com
Connection: Keep-Alive
Cookie: __utma=115222615.545764629.1206300112.1206300112.1206301378.2; __utmz=115222615.1206300112.1.1.
utmcsr=(direct)|utmccn=(direct)|utmcmd=(none); ygho=6\5144785674; AlexaVersion=7.2
```

The server probably records the request and the data which it contains for use in its analytics, but also returns an XML formatted document that is used by the toolbar to display key information about the website, including it's Alexa Rank.

Using the above, we can make very reasonable assumptions as to the function and method as to how the Alexa toolbar works, and the what type of information is transmitted to the Alexa servers. More specifically, the first line contains a standard HTTP/1.1 GET request, the same method in which a browser requests a webpage from a server. Coupled with the "Host" string; string a few lines below, we see the particular "web page" being requested is: data.alex.com/data/j6HV718Dy0g2GJ?cli=10&dat=snba&ver=7.2&cdt=alx\_vw=20&wid=23976&act=000000000000&ss=1536x960&bw=749&t=0&tll=4000&vis=1&rq=6&url=http://www.cybernac.com/

We can further break down this URL into smaller chunks by analyzing the query string. By observing successive queries, certain patterns regarding their use become apparent:

j6HV718Dy0g2GJ

The above string of characters varies depending upon the installation, but is probably an randomly generated value to uniquely identify the computer performing the query. This code could then be used by Alexa to establish long term browsing habits. The string is always 14 characters, and contains the following characters: [A-Z], [a-z], [0-9].

cli=10  
This value is always 10, while its use is unknown, "cli" is probably short for "client" and may be related to the browser version in which the toolbar is installed, though the same information could also be obtained from the user-agent string.

dat=snba  
This value is always snba, and its value is also unknown.

ver=7.2

'Ver', short for 'version', correlates with the version number of toolbar transmitting the request. The current version is 7.2.

cdt=alx\_vw=20

The purpose of this value is also unclear, but fortunately remains the same with each request. On a side note, it's interesting to note the presence of an equal-sign in the value, which suggests (to me at least) a possible bug.

wid=23976

This value is apparently random in nature, but appears to be initialized when the browser is started. All requests will use the same number, until the browser is restarted, at which point a new number appears. It may be short for 'Window ID', or something similar.

act=0000000000

'act' is probably short for 'account' and may be used to track installs of the Alexa toolbar when it is installed by third party software. However, it is usually 11 zeros.

ss=1536x960

'ss' is most certainly an abbreviation for 'screen size' and corresponds to the screen size of the computer which the browser is running on.

bw=749

'bw' is probably short for 'bandwidth'. Doing some research online, the value of this variable varies, but is usually very close to a common bandwidth speed, such as 768kps, or 1,544kbs.

t=0

The purpose of this variable is unknown. Still, every request that has been observed has the same value, zero.

ttl=4000

'ttl' is a common abbreviation for TTL, and could be related to the amount of time since the last toolbar report, or a variety of other timing purposes.

vis=1

This purpose of this variable is unknown. However, it is always the same value, one. 'vis' could be short for 'visibility', and the number one is often used to represent the value 'true', so one possibility is that this value reports whether or not the toolbar is actually visible to the user; some webpages (for example, popups) display in windows without toolbars visible.

rq=6

'rq' probably is short for request, and this number will increment with each successive report sent from the toolbar. It resets to 1 when the browser is restarted.

url=http://www.cybernac.com/

The URL being accessed. The rationale behind this value should be self-evident.

The Alexa service contains no ability to validate information sent to its clients. This is to say, toolbar reports sent from a client are never verified as actually coming from a valid Alexa Toolbar client. As such, the data can easily be replicated to appear to come from virtually any client- even those without the toolbar installed.

The easiest method to replicate the report is to simply embed the URL in a standard HTML <IMG> tag, for example: 

By doing so, the browser will send a report that is structurally identical to the report given by the Alexa Toolbar.

Logically, one would expect the Alexa Internet organization to take reasonable steps to prevent erroneous reports. It's conceivable that a connection problem could cause a legitimate client to transmit multiple reports. To this end, reports containing essentially identical information would be filtered out. To counteract this, black-hatters will attempt to randomize through programming certain aspects of the code to reduce or eliminate the possibility of Alexa invalidating the report.

The following PHP code illustrates how a deceptive website operator might use to artificially inflate their Alexa rank:

Located somewhere inside each page where the Alexa trigger should occur: <?php

```

session_start();
include (&ldquo;alexa.inc&rdquo;);
echo '\n';
?>

```

As part of a separate file named &lsquo;alexa.inc&rsquo;:<?php

```

function alexa() {
    $domain = "http://data.alexa.com";
    $keylength = 14;

    if (isset($_SESSION['alexa_keyid'])) {
        $keyid = $_SESSION['alexa_keyid'];
    } else {
        $keyid = substr(str_shuffle("abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890"), 0,
$keylength);
        $_SESSION['alexa_keyid'] = $keyid;
    }

    $cli = "10";
    $dat = "snba";
    $ver = "7.2";
    $cdt = "alx_vw=20";
    $wid = rand(0,32767);
    $act = "000000000000";
    $ssarray = array ("800x800", "1024x768", "1280x768", "1280x800",
        "1280x1024", "1600x1200", "1680x1050", "1920x1200");
    shuffle($ssarray);
    $ss = current($ssarray);
    $bwarray = array (749, 946, 1523);
    shuffle($bwarray);
    $bw = current($bwarray);
    $t = "0";
    $ttl = rand(200,1000);
    $vis = "1";
    $rq = rand(15,80);
    $url = $_SERVER[&lsquo;SERVER_NAME&rsquo;];

    $params="/data/" . $keyid . "?cli=" . $cli . "&dat=" . $dat . "&ver=" . $ver . "&cdt=" . $cdt . "&wid=" . $wid . "&act=" . $act
        . "&ss=" . $ss . "&bw=" . $bw . "&t=" . $t . "&ttl=" . $ttl . "&vis=" . $vis . "&rq=" . $rq . "&url=" . $url;

    return $domain . $params;
}
?>Analysis

```

So why does this work? The rational is simple. Let&rsquo;s suppose that 1% of all web users have the Alexa toolbar installed. An overly simplistic logic would suggest that if Alexa receives 10 reports from unique clients to your website during a given period, your actual traffic is probably much closer to 1,000 visitors, when accounting for all the users without the toolbar installed. But because nearly all visitors will load the image supplied by the above referenced code, the reality is that 100% of your visitors are sending the toolbar report instead of the 1% Alexa is expecting. However, Alexa still applies the same assumption that only 1% of the internet population have the toolbar installed, and still performs the same multiplication to estimate your traffic value.

By using the above code along with a traffic economizer such as PRIVOX, a single website operator can simulate traffic coming from hundreds or thousands of unique sources from all over the world, each sending valid reports to Alexa Internet. The Black-hatters Rationale

Why would anybody want to go thru the trouble of the above? The reason is rather simple; money. While a higher Alexa ranking does not impact search engine results, dishonest web marketers may choose to artificially increase their rankings in preparation for a sale in order to make the property appear more valuable than it really is. On the same token, an unscrupulous search engine marketer might point to inflated Alexa scores as part of a sales pitch to illustrate an increase in web traffic; this goes beyond just filling the log files with illegitimate traffic. By presenting reports from an independent third party, the marketer may attempt to reinforce statistics that in reality aren&rsquo;t true. In Review

It is unclear why Alexa Internet has chosen not to take reasonable steps to prevent &lsquo;ballot-stuffing&rsquo; of their rankings. Doing so would be rather simple; all that would be required would be for the Alexa server to request that each

toolbar &ldquo;verify&rdquo; any data sent to it. The data would be simply ignored by clients without the toolbar installed.

As such, using Alexa to gauge the popularity and/or value of a given website is discouraged as the results are relatively easy for a website operator to manipulate.

Correction: Digg user Hijinks was kind enough to point out a typo in the code above, which has been corrected. Thanks.