Finding top access IPs in Webserver or how to delay connects from Bots (Web Spiders) to your site to prevent connect Denial of Service

Author : admin

If you're a **sysadmin who has to deal with cracker attemps for DoS (Denial of Service)** on **single or multiple servers (clustered CDN or standalone) Apache Webservers**, nomatter whether working for some web hosting company or just **running your private run home brew web server its very useful thing to inspect Web Server log file (in Apache HTTPD case that's access.log)**.

Sometimes Web Server overloads and the follow up **Danial of Service (DoS)** affect is not caused by evil crackers (mistkenly often called **hackers** but by some **data indexing Crawler Search Engine bots** who are badly configured to aggressively crawl websites and hence causing high webserver loads flooding your servers with bad **404** or **400, 500** or other requests, just to give you an example of such obstructive bots.

1. Dealing with bad Search Indexer Bots (Spiders) with robots.txt

Just as I mentioned **hackers** word above I feel obliged to expose the badful lies the press and media spreading for years misconcepting in people's mind the word **cracker (computer intruder)** with a **hacker**, if you're one of those who mistakenly call security intruders hackers I recommend you read Dr. Richard Stallman's article On Hacking to get the proper understanding that **hacker is an cheerful attitude of mind and spirit** and a hacker could be anyone who has this kind of curious and playful mind out there. **Very often hackers are computer professional, though many times they're skillful**

programmers, a hacker is tending to do things in a very undstandard and weird ways to make fun out of life but definitelely follow the rule of do no harm to the neighbor.

Well after the short lirical distraction above, let me continue;

Here is a short list of Search Index Crawler bots with very aggressive behaviour towards websites:

 # mass download bots / mirroring utilities
1. webzip
2. webmirror
3. webcopy
4. netants
5. getright
6. wget
7. webcapture
8. libwww-perl
9. megaindex.ru
10. megaindex.com
11. Teleport / TeleportPro
12. Zeus
****....***

Note that **some of the listed crawler bots are actually a mirroring clients tools (wget) etc.**, they're also included in the list of server hammering bots because often websites are attempted to be mirrored by people who want to mirror content for the sake of good but perhaps these days more often mirror (**duplicate**) your content for the sake of stealing, this is called in Web language **Content Stealing in SEO language.**

I've found a very comprehensive list of Bad Bots to block on Mike's tech blog his website provided example of bad robots.txt file is mirrored as plain text file here

Below is the list of Bad Crawler Spiders taken from his site:

 # robots.txt to prohibit bad internet search engine spiders to crawl your website

```
# Begin block Bad-Robots from robots.txt
User-agent: asterias
Disallow:/
User-agent: BackDoorBot/1.0
Disallow:/
User-agent: Black Hole
Disallow:/
User-agent: BlowFish/1.0
Disallow:/
User-agent: BotALot
Disallow:/
User-agent: BuiltBotTough
Disallow:/
User-agent: Bullseye/1.0
Disallow:/
User-agent: BunnySlippers
Disallow:/
User-agent: Cegbfeieh
Disallow:/
User-agent: CheeseBot
Disallow:/
User-agent: CherryPicker
Disallow:/
User-agent: CherryPickerElite/1.0
Disallow:/
User-agent: CherryPickerSE/1.0
Disallow:/
User-agent: CopyRightCheck
Disallow:/
User-agent: cosmos
Disallow:/
User-agent: Crescent
Disallow:/
User-agent: Crescent Internet ToolPak HTTP OLE Control v.1.0
Disallow:/
User-agent: DittoSpyder
Disallow:/
User-agent: EmailCollector
Disallow:/
User-agent: EmailSiphon
Disallow:/
User-agent: EmailWolf
Disallow:/
User-agent: EroCrawler
Disallow:/
User-agent: ExtractorPro
```

Disallow:/
User-agent: Foobot
Disallow:/
User-agent: Harvest/1.5
Disallow:/
User-agent: hloader
Disallow:/
User-agent: httplib
Disallow:/
User-agent: humanlinks
Disallow:/
User-agent: InfoNaviRobot
Disallow:/
User-agent: JennyBot
Disallow:/
User-agent: Kenjin Spider
Disallow:/
User-agent: Keyword Density/0.9
Disallow:/
User-agent: LexiBot
Disallow:/
User-agent: libWeb/clsHTTP
Disallow:/
User-agent: LinkextractorPro
Disallow:/
User-agent: LinkScan/8.1a Unix
Disallow:/
User-agent: LinkWalker
Disallow:/
User-agent: LNSpiderguy
Disallow:/
User-agent: lwp-trivial
Disallow:/
User-agent: lwp-trivial/1.34
Disallow:/
User-agent: Mata Hari
Disallow:/
User-agent: Microsoft URL Control – 5.01.4511
Disallow:/
User-agent: Microsoft URL Control – 6.00.8169
Disallow:/
User-agent: MIIxpc
Disallow:/
User-agent: MIIxpc/4.2
Disallow:/
User-agent: Mister PiX

Disallow:/
User-agent: moget
Disallow:/
User-agent: moget/2.1
Disallow:/
User-agent: mozilla/4
Disallow:/
User-agent: Mozilla/4.0 (compatible; BullsEye; Windows 95)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows 95)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows 98)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows NT)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows XP)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows 2000)
Disallow:/
User-agent: Mozilla/4.0 (compatible; MSIE 4.0; Windows ME)
Disallow:/
User-agent: mozilla/5
Disallow:/
User-agent: NetAnts
Disallow:/
User-agent: NICErsPRO
Disallow:/
User-agent: Offline Explorer
Disallow:/
User-agent: Openfind
Disallow:/
User-agent: Openfind data gathere
Disallow:/
User-agent: ProPowerBot/2.14
Disallow:/
User-agent: ProWebWalker
Disallow:/
User-agent: QueryN Metasearch
Disallow:/
User-agent: RepoMonkey
Disallow:/
User-agent: RepoMonkey Bait & Tackle/v1.01
Disallow:/
User-agent: RMA
Disallow:/
User-agent: SiteSnagger

Disallow:/
User-agent: SpankBot
Disallow:/
User-agent: spanner
Disallow:/
User-agent: suzuran
Disallow:/
User-agent: Szukacz/1.4
Disallow:/
User-agent: Teleport
Disallow:/
User-agent: TeleportPro
Disallow:/
User-agent: Telesoft
Disallow:/
User-agent: The Intraformant
Disallow:/
User-agent: TheNomad
Disallow:/
User-agent: TightTwatBot
Disallow:/
User-agent: Titan
Disallow:/
User-agent: toCrawl/UrlDispatcher
Disallow:/
User-agent: True_Robot
Disallow:/
User-agent: True_Robot/1.0
Disallow:/
User-agent: turingos
Disallow:/
User-agent: URLy Warning
Disallow:/
User-agent: VCI
Disallow:/
User-agent: VCI WebViewer VCI WebViewer Win32
Disallow:/
User-agent: Web Image Collector
Disallow:/
User-agent: WebAuto
Disallow:/
User-agent: WebBandit
Disallow:/
User-agent: WebBandit/3.50
Disallow:/
User-agent: WebCopier

Disallow:/
User-agent: WebEnhancer
Disallow:/
User-agent: WebmasterWorldForumBot
Disallow:/
User-agent: WebSauger
Disallow:/
User-agent: Website Quester
Disallow:/
User-agent: Webster Pro
Disallow:/
User-agent: WebStripper
Disallow:/
User-agent: WebZip
Disallow:/
User-agent: WebZip/4.0
Disallow:/
User-agent: Wget
Disallow:/
User-agent: Wget/1.5.3
Disallow:/
User-agent: Wget/1.6
Disallow:/
User-agent: WWW-Collector-E
Disallow:/
User-agent: Xenu's
Disallow:/
User-agent: Xenu's Link Sleuth 1.1c
Disallow:/
User-agent: Zeus
Disallow:/
User-agent: Zeus 32297 Webster Pro V2.9 Win32
Disallow:/
Crawl-delay: 20
 # Begin Exclusion From Directories from robots.txt
Disallow: /cgi-bin/

Veryimportant variable among the ones passed by above robots.txt is

Crawl-Delay*: 20*

You might want to tune that variable a ***Crawl-Delay of 20*** *instructs all IP connects from any Web Spiders that are respecting* ***robots.txt*** *variables* ***to delay crawling with 20 seconds between each and every connect client request****, that is really useful for the Webserver* ***as less connects means less CPU and Memory usage and less degraded performance put by aggressive bots crawling your site like crazy, requesting resources 10 times per second or so ...***

As you can conclude by the naming of some of the bots having them disabled would prevent your domain/s clients from Email harvesting Spiders and other not desired activities.

2. Listing IP addresses Hits / How many connects per IPs used to determine problematic server overloading a huge number of IPs connects

After saying few words about SE bots and I think it it is fair to also **mention here a number of commands, that helps the sysadmin to inspect Apache's access.log files.**
Inspecting the log files regularly is really useful as the number of malicious Spider Bots and the Cracker users tends to be raising with time, so having a good way to track the IPs that are stoning at your webserver and later prohibiting them softly to crawl either via **robots.txt (not all of the Bots would respect that)** or **.htaccess file or as a last resort directly form firewall is really useful to know.**

- Below command Generate a list of IPs showing how many times of the IPs connected the webserver (bear in mind that commands are designed log fields order as given by most GNU / Linux distribution + Apache default logging configuration;

```
webhosting-server:~# cd /var/log/apache2 webhosting-server:/var/log/apache2# cat access.log| awk '{print $1}' | sort | uniq -c |sort -n
```

Below command provides statistics info based on whole access.log file records, **sometimes you will need to have analyzed just a chunk of the webserver log, lets say last 12000 IP connects, here is how**:

```
webhosting-server:~# cd /var/log/apache2 webhosting-server:/var/log/apache2# tail -n 12000 access.log| awk '{print $1}' | sort | uniq -c |sort -n
```

You can combine above basic bash shell parser commands with the **watch** *command to have a top like refresh statistics every few updated refreshing IP statistics of most active customers on your websites.*

Here is an example:

```
webhosting-server:~# watch "cat access.log| awk '{print $1}' | sort | uniq -c |sort -n";
```

Once you have the top connect IPs if **you have a some IP connecting with lets say 8000-10000 thousand times in a really short interval of time 20-30 minues or so**. Hence it is a good idea to **investigate further where is this IP originating from and if it is some malicious Denial of Service**, **filter it out either in Firewall (with iptables rules) or ask your ISP or webhosting to do you a favour and drop all the incoming traffic from that IP.**

Here is how to investigate a bit more about a server stoner IP;
Lets assume that you found IP: **176.9.50.244 to be having too many connects to your webserver:**

```
webhosting-server:~# grep -i 176.9.50.244 /var/log/apache2/access.log|tail -n 1
176.9.50.244 - - [12/Sep/2017:07:42:13 +0300] "GET / HTTP/1.1" 403 371 "-" "Mozilla/5.0 (compatible; MegaIndex.ru/2.0; +http://megaindex.com/crawler)"
```

```
webhosting-server:~# host 176.9.50.244
244.50.9.176.in-addr.arpa domain name pointer static.244.50.9.176.clients.your-server.de.
```

```
webhosting-server:~# whois 176.9.50.244|less
```

The outout you will get would be something like:

```
% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
%       To receive output for a database update, use the "-B" flag.

% Information related to '176.9.50.224 - 176.9.50.255'

% Abuse contact for '176.9.50.224 - 176.9.50.255' is 'abuse@hetzner.de'

inetnum:        176.9.50.224 - 176.9.50.255
netname:        HETZNER-RZ15
descr:          Hetzner Online GmbH
descr:          Datacenter 15
country:        DE
admin-c:        HOAC1-RIPE
tech-c:         HOAC1-RIPE
status:         ASSIGNED PA
mnt-by:         HOS-GUN
mnt-lower:      HOS-GUN
mnt-routes:     HOS-GUN
created:        2012-03-12T09:45:54Z
last-modified:  2015-08-10T09:29:53Z
source:         RIPE

role:           Hetzner Online GmbH - Contact Role
address:        Hetzner Online GmbH
address:        Industriestrasse 25
address:        D-91710 Gunzenhausen
address:        Germany
phone:          +49 9831 505-0
fax-no:         +49 9831 505-3
```

abuse-mailbox: abuse@hetzner.de
*remarks: **********
remarks: * For spam/abuse/security issues please contact *
remarks: * abuse@hetzner.de, not this address. *
remarks: * The contents of your abuse email will be *
remarks: * forwarded directly on to our client for *
....*

3. Generate list of directories and files that are most called by clients

```
webhosting-server:~# cd /var/log/apache2; webhosting-server:/var/log/apache2# awk '{print $7}' access.log|cut -d? -f1|sort|uniq -c|sort -nk1|tail -n10
```

(take in consideration that this info is provided only on current records from **/var/log/apache2/** and is short term for long term statistics you have to merge all existing **gzipped /var/log/apache2/access.log.*.gz**)

To merge all the old gzipped files into one single file and later use above shown command to analyize run:

```
    cd /var/log/apache2/
cp -rpf *access.log*.gz apache-gzipped/
cd apache-gzipped
for i in $(ls -1 *access*.log.*.gz); do gzip -d $i; done
rm -f *.log.gz;

for i in $(ls -1 *|grep -v access_log_complete); do cat $i >> access_log_complete; done
```

Though the accent of above article is Apache Webserver log analyzing, the given command examples can easily be recrafted to work properly on other Web Servers LigHTTPD, Nginx etc.

Above **commands are about to put a higher load to your server during execution*, so on busy servers it is a better idea, to first go and synchronize the access.log files to another less loaded servers in most small and midsized companies this is being done by a periodic synchronization of the logs to the log server used usually only to store log various files*** and later used to do various analysis our run analyse software such as ***Awstats, Webalizer,*** Piwik**,** Go Access **etc.**

Worthy to mention **one great text console must have Apache tool** that should be mentioned to analyze in real time for the lazy ones to type so much is *Apache-top* **but those script will be not installed on most webhosting servers and VPS-es, so if you don't happen to own a self-hosted dedicated server / have webhosting company etc. - (have root admin access on server)**, but have an ordinary server account you can use above commands to **get an overall picture of abusive webserver IPs.**

If you have a Linux with a desktop GUI environment and have somehow mounted remotely the weblog server partition another really awesome way to visualize in real time the connect requests to web server Apache / Nginx etc. is with Logstalgia

Well that's all folks, I hope that article learned you something new. Enjoy

Thanks for article neo-tux picture to segarkshtri.com.np)