



Table of contents

Release Notes

[What is AWStats / Features](#)
[New Features / Changelog](#)
[Upgrade](#)

Reference manual

[Install, Setup and Use AWStats](#)
[Configuration Directives/Options](#)
[Configuration for Extra Sections feature](#)
[Contribs, plugins and resources](#)
[Other provided utilities](#)
[Glossary of terms](#)

Other Topics

[Comparison with other log analyzers](#)
[Benchmarks](#)
[Plugin Development](#)

[FAQ and Troubleshooting](#)
[AWStats License](#)

AWStats modules for third tools

[AWStats module for Webmin](#)

[AWStats module for Dolibarr ERP &CRM](#)



What is AWStats / Features Overview

AWStats is short for Advanced Web Statistics. **AWStats is powerful log analyzer which creates advanced web, ftp, mail and streaming server statistics reports** based on the rich data contained in server logs. Data is graphically presented in **easy to read web pages**.

AWStats development started in 1997 and is still developed today by same author (Laurent Destailleur). However, development is now only "maintenance" or small new features because author spend, since July 2008, most of his time as project leader on another major OpenSource projet called [Dolibarr ERP & CRM](#). But a lot of other developers maintains the software, above all for Linux distributions (fedora, debian, ubuntu...).

Designed with flexibility in mind, AWStats can be run through a web browser **CGI** (common gateway interface) or directly from the operating system **command line**. Through the use of intermediary data base files, AWStats is able to quickly process large log files, as often desired. With support for both standard and custom log format definitions, AWStats can analyze log files from Apache (NCSA combined/XLF/ELF or common/CLF log format), Microsoft's IIS (W3C log format), WebStar and most web, proxy, wap and streaming media servers as well as ftp and mail server logs.

See how the most famous open source statistics tools (AWStats, Analog, Webalizer) stack up feature by feature in this [comparison table](#).

AWStats is free software distributed under the [GNU General Public License](#). The [license chart](#) illustrates what you can and cannot do.

As AWStats works from the command line as well as a CGI, it is compatible with [web hosting providers which allow CGI and log access](#).

Features

AWStats' reports include a wide range of information on your web site usage:

- * Number of Visits, and number of Unique visitors.
- * Visit duration and latest visits.
- * Authenticated Users, and latest authenticated visits.
- * Usage by Months, Days of week and Hours of the day (pages, hits, KB).
- * Domains/countries (and regions, cities and ISP with [Maxmind proprietary geo databases](#)) of visitor's hosts (pages, hits, KB, [269 domains/countries detected](#)).
- * Hosts list, latest visits and unresolved IP addresses list.
- * Most viewed, Entry and Exit pages.
- * Most commonly requested File types.
- * Web Compression statistics (for Apache servers using mod_gzip or mod_deflate modules).
- * Visitor's Browsers (pages, hits, KB for each browser, each version, [123 browsers](#) detected: Web, Wap, Streaming Media browsers..., around [482](#) with the "phone browsers" database).
- * Visitor's Operating Systems (pages, hits, KB for each OS, [45 OS](#) detected).
- * Robots visits, including search engine crawlers ([381 robots](#) detected).
- * Track Downloads such as PDFs, compressed files and others
- * Search engines, Keywords and Phrases used to find your site ([The 122 most famous search engines are detected like Yahoo, Google, Altavista, etc...](#))
- * HTTP Errors (Page Not Found with latest referrer, ...).
- * User defined reports based on url, url parameters, referrer (referer) fields extend AWStats' capabilities to provide even greater technical and marketing information.
- * Number of times your site is added to Bookmarks / Favorites.
- * Screen size (to capture this, some HTML tags must be added to a site's home page).
- * Ratio of integrated Browser Support for: Java, Flash, Real G2 player, Quicktime reader, PDF reader, WMA reader (as above, requires insertion of HTML tags in site's home page).
- * Cluster distribution for load balanced servers.

In addition, AWStats provides the following:

- * Wide range of log formats. AWStats can analyze: Apache NCSA combined (XLF/ELF) or common (CLF) log files,

Microsoft IIS log files (W3C), WebStar native log files and other web, proxy, wap, streaming media, ftp and mail server log files. See [AWStats F.A.Q.](#) for examples.

- * Reports can be run from the operating system command line and from a web browser as a CGI (common gateway interface). In CGI mode, dynamic filter capabilities are available for many charts.
- * Statistics update can be run from a web browser as well as scheduled for automatic processing.
- * Unlimited log file size
- * Load balancing system split log files.
- * Support 'nearly sorted' log files, even for entry and exit pages.
- * Reverse DNS lookup before or during analysis; supports DNS cache files.
- * Country detection from IP location (geoip) or domain name.
- * Plugins for US/Canadian Regions, Cities and major countries regions, ISP and/or Organizations reports (require non free third product [geoipregion](#), [geoipcity](#), [geoipisp](#) and/or [geoiporg](#) database).
- * WhoIS lookup links.
- * Vast array of configurable options/filters and plugins supported.
- * Modular design supports inclusion of addition features via plugins.
- * Multi-named web sites supported (virtual servers, great for web-hosting providers).
- * Cross Site Scripting Attacks protection.
- * Reports available in many international languages. See [AWStats F.A.Q.](#) for full list. Users can provide files for additional languages not yet available.
- * No need for esoteric perl libraries. AWStats works with all basic perl interpreters.
- * Dynamic reports through a CGI interface.
- * Static reports in one or framed HTML or XHTML pages; experimental PDF export through 3rd party "html2pdf" software.
- * Customize look and color scheme to match your site design; with or without CSS (cascading style sheets).
- * Help and HTML tooltips available in reports.
- * Easy to use – all configuration directives are confined to one file for each site.
- * Analysis database can be stored in XML format for easier use by external applications, like XSLT processing (one xslt transform example provided).
- * A Webmin module is supplied.
- * Absolutely free (even for web hosting providers); source code is included ([GNU General Public License](#)).
- * Works on all platforms with Perl support.
- * AWStats has a [XML Portable Application Description](#).

Requirements:

AWStats usage has the following requirements:

- * You must have access to the server logs for the reporting you want to perform (web/ftp/mail).
 - * You must be able to run perl scripts (.pl files) from command line and/or as a CGI. If not, you can solve this by downloading latest Perl version at [ActivePerl](#) (Win32) or [Perl.com](#) (Unix/Linux/Other).
- See [AWStats F.A.Q.](#) for examples of supported OS and Web servers.
-



AWStats Installation, Configuration and Reporting

There are 3 steps to begin using AWStats:

- I. Setup: Installation and configuration
- II. Process logs: Building/updating statistics database
- III. Run Reports: Building and reading reports

I. Setup: Installation and configuration using awstats configure.pl

A) Setup for an Apache or compatible web server (on Unix/Linux, Windows, MacOS...)

* Step 1:

(if you use a package provided with a Linux distribution or Windows installer, step 1 might have already been done; if you don't know, you can run this step again)

After downloading and extracting the AWStats package, you should run the awstats_configure.pl script to do several setup actions. You will find it in the AWStats **tools** directory (If using the Windows installer, the script is automatically launched):

```
perl awstats configure.pl
```

This is what the script does/asks (you can do all these steps manually instead of running awstats configure.pl if you prefer):

A) awstats_configure.pl tries to determine your current log format from your Apache web server configuration file httpd.conf (it asks for the path if not found). If you use a **common** log, awstats_configure.pl will suggest changing it to the **NCSA combined/XLF/ELF** format (you can use your own custom log format but this predefined log format is often the best choice and makes setup easier).

If you answer yes, awstats_configure.pl will modify your **httpd.conf**, changing the following directive:

```
from
CustomLog /yourlogpath/yourlogfile common
to
CustomLog /yourlogpath/yourlogfile combined
```

See the Apache manual for more information on this directive (possibly installed on your server as www.mysite.com/manual).

B) awstats_configure.pl will then add, if not already present, the following directives to your Apache configuration file (note that the "/usr/local/awstats/wwwroot" path might differ according to your distribution or OS:

```
#
# Directives to add to your Apache conf file to allow use of AWStats as a CGI.
# Note that path "/usr/local/awstats/" must reflect your AWStats Installation path.
#
Alias /awstatsclasses "/usr/local/awstats/wwwroot/classes/"
Alias /awstatscss "/usr/local/awstats/wwwroot/css/"
Alias /awstatsicons "/usr/local/awstats/wwwroot/icon/"
ScriptAlias /awstats/ "/usr/local/awstats/wwwroot/cgi-bin/"
#
# This is to permit URL access to scripts/files in AWStats directory.
#
<Directory "/usr/local/awstats/wwwroot">
Options None
AllowOverride None
Order allow,deny
Allow from all
```

</Directory>

C) if changes were made as indicated in parts A and B, awstats_configure.pl restarts Apache to apply the changes. To be sure the log format change is effective, go to your homepage. This is an example of the type of records you should see inserted in your new log file after Apache was restarted:

```
62.161.78.75 -- [dd/mmm/yyyy:hh:mm:ss +0000] "GET / HTTP/1.1" 200 1234 "http://www.from.com/from.html" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)"
```

D) awstats_configure.pl will ask you for a name for the configuration profile file. Enter an appropriate name such as that of your web server or the virtual domain to be analyzed, i.e. **mysite**.

awstats_configure.pl will create a new file called **awstats.mysite.conf** by copying the template file **awstats.model.conf**. The new file location is:

- For Linux/BSD/Unix users: /etc/awstats.
- For Mac OS X, Windows and other operating systems: the same directory as awstats.pl (cgi-bin).

E) awstats_configure.pl ends.

* Step 2:

Once a configuration file has been created (by awstats_configure.pl, by your package installer or just by a manual copy of awstats.model.conf), it's important to verify that the "MAIN PARAMETERS" match your needs. Open awstats.mysite.conf in your favorite text editor (i.e. notepad.exe, vi, gedit, etc) – don't use a word processor – and make changes as required.

Particular attention should be given to these parameters:

- Verify the [LogFile](#) value. It should be the full path of your server log file (You can also use a relative path from your awstats.pl directory, but a full path avoids errors).
- Verify the [LogType](#) value. It should be "W" for analyzing web log files.
- Check if [LogFormat](#) is set to "1" (for "NCSA apache combined/ELF/XLF log format") or use a custom log format if you don't use the combined log format.
- Set the [SiteDomain](#) parameter to the main domain name or the intranet web server name used to reach the web site to analyze (Example: www.mysite.com). If you have several possible names for same site, use the main domain name and add the others to the list in the [HostAlias](#) parameter.
- You can also change other parameters if you want. The full list is described in [Configurations/Directives options](#) page.

Installation and configuration is finished. You can jump to the [Process logs: Building/updating statistics database](#) section.

B) Setup for Microsoft's IIS server

* Step 1:

Configure IIS to create logs in the "**Extended W3C log format**" (You can still use your own custom log format but setup is easier if you use the standard extended format). To do so, start the IIS management console snap-in, select the appropriate web site and open its Properties. Choose "W3C Extended Log Format", then Properties, then the Tab "Extended Properties" and uncheck everything under Extended Properties. Once they are all cleared, check just the following fields:

```
date
time
c-ip
cs-username
cs-method
cs-uri-stem
cs-uri-query
sc-status
sc-bytes
cs-version
cs(User-Agent)
cs(Referer)
```

To be sure the log format change is effective, you must stop IIS, backup it up (if you desire) and remove all of the old log files, restart IIS and go to your homepage. This is an example of the type of records you should find in the new log file:

```
2000-07-19 14:14:14 62.161.78.73 -- GET / 200 1234 HTTP/1.1
Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0) http://www.from.com/from.htm
```

* Step 2:

Copy the contents of the AWStats provided **cgi-bin** folder, from where the AWStats package put it on your local hard drive, to your server's **cgi-bin** directory (this includes **awstats.pl**, **awstats.model.conf**, and the **lang**, **lib** and **plugins** sub-directories).

*** Step 3:**

Move AWStats **icon sub-directories** and its content into a directory readable by your web server, for example `C:\yourwwwroot\icon`.

*** Step 4:**

Create a configuration file by copying **awstats.model.conf** to a new file named **awstats.mysite.conf** where "mysite" is a value of your choice but usually is the domain or virtual host name. This new file must be saved in the same directory as **awstats.pl** (i.e. **cgi-bin**).

*** Step 5:**

Edit your new **awstats.mysite.conf** file to match your specific environment:

- Change the **LogFile** value to the full path of your web server log file (You can also use a relative path from your **awstats.pl** (**cgi-bin**) directory).
- Change the **LogType** value to "W" for analyzing web log files.
- Change the **LogFormat** to 2 if you are using the "**Extended W3C log format**" described in step 1; in the case of a custom format, list the IIS fields being logged, for example:
LogFormat="date time c-ip cs-username cs-method cs-uri-stem cs-uri-query sc-status sc-bytes cs-version cs(User-Agent) cs(Referer)"
- Change the **DirIcons** parameter to reflect relative path of icon directory.
- Set the **SiteDomain** parameter to the main domain name or the intranet web server name used to reach the web site being analyzed (Example: `www.mydomain.com`).
- Set the **AllowToUpdateStatsFromBrowser** parameter to 1 if you don't have command line access and have only **cgi** access.
- Review and change other parameters if appropriate.

Installation and configuration is finished. You can jump to the [Process logs: Building/Updating statistics database](#) section.

C) Setup for other web servers

The setup process is similar to the setup for Apache or IIS.

Use **LogFormat** to value "3" if you have WebStar native log format. Use a personalized **LogFormat** if your log format is other.

D) Setup for other Internet servers, i.e. FTP, Mail, Streaming media

The setup process for other file formats is described in the relevant FAQ topics: [FAQ-COM090: FTP](#) [FAQ-COM100: Mail](#) and [FAQ-COM110: Streaming media](#).

II. Process logs: Building/updating statistics database

*** Update from command line (recommended):**

The first log analysis should be done manually from the command line since the process may be long and it's easier to solve problems when you can see the command output (if you don't have Command Line access, skip to Step 2). The AWStats create (and update) statistics database command is:

```
perl awstats.pl -config=mysite -update
```

where **mysite** must be substituted with the domain/virtual host name you selected earlier during AWStats configuration.

AWStats will read the configuration file **awstats.mysite.conf** (or if not found, **awstats.conf**) and create/update its database with all summary information issued from analyzed log file.

AWStats statistics database files are saved in directory defined by the **DirData** parameter in configuration file. When the create/update is finished, you should see a similar result on your screen:

```
Update for config "/etc/awstats/awstats.mysite.conf"
With data in log file "/pathtoyourlog/yourlog.log"...
```

```
Phase 1 : First bypass old records, searching new record...
Searching new records from beginning of log file...
Phase 2 : Now process new records (Flush history on disk after 20000 hosts)...
Jumped lines in file: 0
Parsed lines in file: 225730
Found 122 dropped records,
Found 87 corrupted records,
Found 0 old records,
Found 225521 new qualified records.
```

Dropped records are records discarded because they were not "user HTTP requests" or were requests matching AWStats filters (See the [SkipHosts](#), [SkipUserAgents](#), [SkipFiles](#), [OnlyHosts](#), [OnlyUserAgents](#) and [OnlyFiles](#) parameters). If you want to see which lines were dropped, you can add the **-showdropped** option on the command line.

Corrupted records are records that do not match the log format defined by the "LogFormat" parameter in the AWStats configuration file. All web servers will typically have a few corrupted records (<5%) even when everything works correctly. This can result for several reasons: 1) Web server internal bugs, 2) bad requests made by buggy browsers, 3) a dirty web server shutdown, such as unplugging the server...

If all of your lines are corrupted and the [LogFormat](#) parameter in AWStats configuration file is correct, then there may be a setup problem with your web server log format. Don't forget that your [LogFormat](#) parameter in the AWStats configuration file **MUST** match the log file format you analyze. If you want to see which lines are corrupted, you can add the **-showcorrupted** option on the command line.

Old records are simply records that were already processed by a previous update session. Although it is not necessary to purge your log file after each update process, it is highly recommended that you do so as often as possible.

New records are records in your log file that were successfully used to build/update the statistics database.

Note: A log analysis process might be slow (one second for each 4500 lines of your logfile with an Athlon 1Ghz, plus DNS resolution time for each different IP address in your logfile if [DNSLookup](#) is set to 1 and not already done in your log file). See the [Benchmarks page](#) for more detailed information.

* Update from a browser:

AWStats statistics can also be updated from a browser, providing real-time statistics, by clicking the "Update now" link that appears when AWStats is used as a CGI (The URL is described in the next section '[Run reports: Building and reading reports](#)').

Warning!!

To enable this link, the configuration file parameter [AllowToUpdateStatsFromBrowser](#) must be set to 1 (The link is not enabled by default).

Using the on-line update does not prevent you from running the update process automatically on a scheduled basis (the command is same as that of the first update process above).

For this, you have two choices:

- Include the update command in your **logrotate** process. See [FAQ-COM120](#) for details.
- Or add instructions in your **crontab** (Unix/Linux) or your **task scheduler** (Windows), to regularly launch the Awstats update process. See [FAQ-COM130](#) for details.

See the AWStats [Benchmarks page](#) for the recommended update/logrotate frequency.

III. Run reports: Building and reading reports

To see the analysis results, you have several options depending on your [security policy](#).

Note: you must have created a statistics data base for the analysis period by processing your log files before you try to create reports. See the previous section.

1. The first option is to build the main reports, in a static HTML page, from the command line, using the following syntax (skip to the second option if you only have CGI access):

```
perl awstats.pl -config=mysite -output -staticlinks > awstats.mysite.html
```


where *mysite* must be substituted with the domain/virtual host name you selected earlier during AWStats configuration.

To create specific individual reports, specify the report name on the command line as follows¹:

```
perl awstats.pl -config=mysite -output=alldomains -staticlinks > awstats.mysite.alldomains.html
perl awstats.pl -config=mysite -output=allhosts -staticlinks > awstats.mysite.allhosts.html
perl awstats.pl -config=mysite -output=lasthosts -staticlinks > awstats.mysite.lasthosts.html
perl awstats.pl -config=mysite -output=unknownip -staticlinks > awstats.mysite.unknownip.html
perl awstats.pl -config=mysite -output=alllogins -staticlinks > awstats.mysite.alllogins.html
perl awstats.pl -config=mysite -output=lastlogins -staticlinks > awstats.mysite.lastlogins.html
perl awstats.pl -config=mysite -output=allrobots -staticlinks > awstats.mysite.allrobots.html
perl awstats.pl -config=mysite -output=lastrobots -staticlinks > awstats.mysite.lastrobots.html
perl awstats.pl -config=mysite -output=urldetail -staticlinks > awstats.mysite.urldetail.html
perl awstats.pl -config=mysite -output=urllentry -staticlinks > awstats.mysite.urllentry.html
perl awstats.pl -config=mysite -output=urlexit -staticlinks > awstats.mysite.urlexit.html
perl awstats.pl -config=mysite -output=browserdetail -staticlinks > awstats.mysite.browserdetail.html
perl awstats.pl -config=mysite -output=osdetail -staticlinks > awstats.mysite.osdetail.html
perl awstats.pl -config=mysite -output=unknownbrowser -staticlinks > awstats.mysite.unknownbrowser.html
perl awstats.pl -config=mysite -output=unknownnos -staticlinks > awstats.mysite.unknownnos.html
perl awstats.pl -config=mysite -output=refererse -staticlinks > awstats.mysite.refererse.html
perl awstats.pl -config=mysite -output=refererpages -staticlinks > awstats.mysite.refererpages.html
perl awstats.pl -config=mysite -output=keyphrases -staticlinks > awstats.mysite.keyphrases.html
perl awstats.pl -config=mysite -output=keywords -staticlinks > awstats.mysite.keywords.html
perl awstats.pl -config=mysite -output=errors404 -staticlinks > awstats.mysite.errors404.html
```

¹If you prefer, you can use the [awstats_buildstaticpages](#) tool to build all these pages in one command, or to generate PDF files.

Notes:

a) You can also add a *filter* on the following reports: **urldetail**, **urllentry**, **urlexit**, **allhosts**, **refererpages**. The *filter* can be a regexp (regular expression) on the full key you want AWStats to report on and is appended to the output parameter separated by a ":".

For example, to output the urldetail report, including only pages which contain /news in their URL, you can use the following command line:

```
perl awstats.pl -config=mysite -output=urldetail:/news -staticlinks > awstats.mysite.urldetailwithfilter.html
```

b) If you want to build a report for a particular month, add the options **-month=MM** **-year=YYYY** where MM is the month expressed as two digits, i.e. 03, and year is the four digit year. To build a report for a full year, add the options **-month=all** **-year=YYYY** (warning: this is often resource intensive and might use a lot of memory and CPU. Unix/Linux like operating systems might benefit from use of the "nice" command.)

2) The second option is to dynamically view your statistics from a browser. To do this, use the URI :

```
http://www.myserver.mydomain/awstats/awstats.pl?config=mysite
```

where *mysite* specifies the configuration file to use (AWStats will use the file *awstats.mysite.conf*).

All output command line options (except **-staticlinks** and **-logfile**) are also available when using AWStats with a browser. Just use them as URL parameters: change "-option" to "&option", i.e.

http://www.myserver.mydomain/awstats/awstats.pl?month=MM&year=YYYY&output=unknownnos

Reports are generated in real time from the statistics data base. If this is slow, or putting too much load on your server, consider generating static reports instead.

If the [AllowToUpdateStatsFromBrowser](#) parameter is set to 1 in AWStats configuration file, you will also be able to run the update process from your browser. Just click on the link "Update now".



Upgrading AWStats from a previous version

To upgrade AWStats, all you have to do is:

1. Read this page completely before beginning the upgrade.

2. Optionally backup your AWStats configuration and history data files.

Standard AWStats distributions do not change these, but backing them up allows you to recover them if something does go amiss. The AWStats default distribution will only overwrite the configuration file example called *awstats.model.conf*. Your history data files are saved in the directory defined with the *DirData* parameter in your AWStats configuration file (*awstats.mysite.conf*).

3. Replace AWStats runtime files with the new ones.

To do this you can:

- decompress the new AWStats zip or tgz package over the old installation directory (old AWStats runtime files are replaced by new ones. This works for AWStats versions 3.x and higher).
- or just run your OS package tool (i.e. rpm, apt for Linux, .exe for MS Windows).

Migrating to 6.x versions from earlier versions, observe the following additional notes:

- As of AWStats 6.9+, Perl version is 5.007 or higher is required. To see the version number, run the command *perl -v* from your operating system's command line.
- If you use the [ExtraSections](#) feature, you must check that the parameter(s) *ExtraSectionConditionX* uses a full REGEX syntax (with the 5.x series, this parameter could contain simple string values). If you don't update your configuration, the feature will be broken.
- If you use the [MiscTrackerUrl](#) feature, you must check that your [ShowMiscStats](#) parameter is set to "ajdfqrwp" in addition to setting up [MiscTrackerUrl](#) tags and a JavaScript include for your web site. Otherwise the new default value "a" will be used (only the "Add to favorites" will be reported).
- The *MaxLengthOfURL* parameter has been renamed [MaxLengthOfShownURL](#); update your configuration file accordingly.
- To enable the worm detection feature (not enabled by default), you must add the parameter [LevelForWormsDetection=2](#) in your configuration file.
- If you use the *urlalias* or *userinfo* plugins, you must move the *urlalias.*.txt* or *userinfo.*.txt* file from the *Plugins* directory to the *DirData* directory.

4. Convert your AWStats history files when migrating from 3.x or 4.x versions to 5.x or higher.

If you upgrade from the 3.x or 4.x series to 5.x or higher, AWStats will still be able to 'read' your old history files but a warning may appear to ask you to run the 'migrate' process on your old data files. Just run the command that will appear in warning message. This warning will only appear if data migration is necessary after an upgrade from the 3.x or 4.x series to a 5.x or higher version.

5. Review new feature default settings.

Sometimes new feature parameters are introduced in a new AWStats version. AWStats will use default values for them, maintaining backward compatibility with your existing AWStats configuration. However, you can review the "New Features / Change Log" to see how you might change the new feature's behavior.



AWStats configuration directives/options

Each directive available in the AWStats config file (.conf) is listed here (with examples and default values).

Notes To include an environment variable in any parameter (AWStats will replace it with its value when reading it), follow the example:

Parameter="__ENVNAME__"

DIRECTIVES IN MAIN SETUP SECTION (Required to make AWStats work)

- [LogFile](#)
- [LogType](#)
- [LogFormat](#)
- [LogSeparator](#)
- [DNSLookup](#)
- [DirData](#)
- [DirCgi](#)
- [DirIcons](#)
- [SiteDomain](#)
- [HostAliases](#)
- [AllowToUpdateStatsFromBrowser](#)
- [AllowFullYearView](#)

DIRECTIVES IN OPTIONAL SETUP SECTION (Not required but increase AWStats features)

- [EnableLockForUpdate](#)
- [DNSStaticCacheFile](#)
- [DNSLastUpdateCacheFile](#)
- [SkipDNSLookupFor](#)
- [AllowAccessFromWebToAuthenticatedUsersOnly](#)
- [AllowAccessFromWebToFollowingAuthenticatedUsers](#)
- [AllowAccessFromWebToFollowingIPAddresses](#)
- [CreateDirDataIfNotExists](#)
- [BuildHistoryFormat](#)
- [BuildReportFormat](#)
- [SaveDatabaseFilesWithPermissionsForEveryone](#)
- [PurgeLogFile](#)
- [ArchiveLogRecords](#)
- [KeepBackupOfHistoricFiles](#)
- [DefaultFile](#)
- [SkipHosts](#)
- [SkipUserAgents](#)
- [SkipFiles](#)
- [OnlyHosts](#)
- [OnlyUserAgents](#)
- [OnlyUsers](#)
- [OnlyFiles](#)
- [NotPageList](#)
- [ValidHTTPCodes](#)
- [ValidSMTPCodes](#)
- [AuthenticatedUsersNotCaseSensitive](#)
- [URLNotCaseSensitive](#)
- [URLWithAnchor](#)
- [URLQuerySeparators](#)
- [URLWithQuery](#)
- [URLWithQueryWithOnlyFollowingParameters](#)
- [URLWithQueryWithoutFollowingParameters](#)

- URLReferrerWithQuery
- WarningMessages
- ErrorMessage
- DebugMessages
- NbOfLinesForCorruptedLog
- SplitSearchString
- WrapperScript
- DecodeUA
- MiscTrackerUrl

DIRECTIVES IN OPTIONAL ACCURACY SETUP SECTION (Not required but increase AWStats features)

- LevelForBrowsersDetection
- LevelForOSDetection
- LevelForRefererAnalyze
- LevelForRobotsDetection
- LevelForSearchEnginesDetection
- LevelForKeywordsDetection
- LevelForFileTypesDetection
- LevelForWormsDetection

DIRECTIVES IN OPTIONAL APPEARANCE SETUP SECTION (Not required but increase AWStats features)

- UseFramesWhenCGI
- DetailedReportsOnNewWindows
- Expires
- MaxRowsInHTMLOutput
- Lang
- DirLang
- ShowHeader
- ShowMenu
- ShowSummary
- ShowMonthStats
- ShowDaysOfMonthStats
- ShowDaysOfWeekStats
- ShowHoursStats
- ShowDomainsStats
- ShowHostsStats
- ShowAuthenticatedUsers
- ShowRobotsStats
- ShowWormsStats
- ShowSessionsStats
- ShowPagesStats
- ShowCompressionStats
- ShowFileTypesStats
- ShowFileSizesStats
- ShowDownloadsStats
- ShowOSStats
- ShowBrowsersStats
- ShowScreenSizeStats
- ShowOriginStats
- ShowKeyphrasesStats
- ShowKeywordsStats
- ShowMiscStats
- ShowHTTPErrorsStats
- ShowSMTPErrorsStats
- ShowClusterStats
- AddDataArrayMonthStats
- AddDataArrayShowDaysOfMonthStats
- AddDataArrayShowDaysOfWeekStats
- AddDataArrayShowHoursStats
- IncludeInternalLinksInOriginSection
- MaxNbOfDomain
- MinHitDomain
- MaxNbOfHostsShown
- MinHitHost
- MaxNbOfLoginShown

- MinHitLogin
- MaxNbOfRobotShown
- MinHitRobot
- MaxNbOfDownloadsShown
- MinHitDownloads
- MaxNbOfPageShown
- MaxNbOfOsShown
- MinHitOs
- MaxNbOfBrowsersShown
- MinHitBrowser
- MinHitFile
- MaxNbOfScreenSizesShown
- MinHitScreenSize
- MaxNbOfRefererShown
- MinHitRefer
- MaxNbOfKeywordsShown
- MinHitKeyword
- FirstDayOfWeek
- ShowFlagLinks
- ShowLinksOnUrl
- MaxLengthOfURL
- ShowLinksToWhols
- UseHTTPSLinksForUrl
- LinksToWhols
- LinksToIPWhols
- HTMLHeadSection
- HTMLEndSection
- BarWidth
- BarHeight
- Logo
- LogoLink
- StyleSheet
- color_Background
- color_TableBGTitle
- color_TableTitle
- color_TableBG
- color_TableRowTitle
- color_TableBGRowTitle
- color_TableBorder
- color_text
- color_titletext
- color_weekend
- color_link
- color_hover
- color_u
- color_v
- color_p
- color_h
- color_k
- color_s
- color_e
- color_x

DIRECTIVES FOR PLUGINS

- LoadPlugin

DIRECTIVES IN EXTRA SECTIONS

- ExtraSectionNameX
- ExtraSectionCodeFilterX
- ExtraSectionConditionX
- ExtraSectionFirstColumnTitleX
- ExtraSectionFirstColumnValuesX
- ExtraSectionFirstColumnFormatX
- ExtraSectionStatTypesX
- ExtraSectionAddAverageRowX

- [ExtraSectionAddSumRowX](#)
- [MaxNbOfExtraX](#)
- [MinHitExtraX](#)
- [ExtraTrackedRowsLimit](#)

INCLUDES

- [Include](#)

LogFile

Version : 1.0+

3.1+ for tags %YYYY-n,%YY-n,%MM-n,%DD-n,%HH-n

3.2+ for tag %WM-n

4.0+ for tag %DW-n

4.1+ for tag %NS-n

5.0+ for tag %WY-n

5.1+ for tag %Wm-n, %Wy-n, %Dw-n

```
# "LogFile" contains the web, ftp or mail server logfile to analyze.
# You can use a full path or relative path from awstats.pl directory.
# Example: "/var/log/apache/access.log"
# Example: "../logs/mycombinedlog.log"
# You can also use tags in this filename if you need a dynamic file name
# depending on date or time (Replacement is made by AWStats at the beginning
# of its execution). This is available tags :
# %YYYY-n is replaced with 4 digits year we were n hours ago
# %YY-n is replaced with 2 digits year we were n hours ago
# %MM-n is replaced with month we were n hours ago
# %MO-n is replaced with 3 letters month we were n hours ago
# %DD-n is replaced with day we were n hours ago
# %HH-n is replaced with hour we were n hours ago
# %NS-n is replaced with number of seconds at 00:00 since 1970
# %WM-n is replaced with the week number in month (1-5)
# %Wm-n is replaced with the week number in month (0-4)
# %WY-n is replaced with the week number in year (01-52)
# %Wy-n is replaced with the week number in year (00-51)
# %DW-n is replaced with the day number in week (1-7, 1=sunday)
# use n=24 if you need (1-7, 1=monday)
# %Dw-n is replaced with the day number in week (0-6, 0=sunday)
# use n=24 if you need (0-6, 0=monday)
# Use 0 for n if you need current year, month, day, hour...
# Example: "/var/log/access_log.%YYYY-0%MM-0%DD-0.log"
# Example: "C:/WINNT/system32/LogFiles/W3SVC1/ex%YY-24%MM-24%DD-24.log"
# You can also use a pipe if log file come from a pipe.
# Example: "gzip -d </var/log/apache/access.log.gz |"
# If there is several log files from load balancing servers :
# Example: "/pathtotools/logresolvemerge.pl *.log |"
#
LogFile="/var/log/httpd/mylog.log"
```

LogType

Version : 5.7+

```
# Enter the log file type you want to analyze.
# Possible values:
# W - For a web log file
# M - For a mail log file
# F - For a ftp log file
# Example: W
# Default: W
#
LogType=W
```

LogFormat

Version : 2.1+

3.1+ for tags %host,%logname,%time1,%time2,%methodurl,%methodurlnoproto,%method,%url,
%query,%code,%bytesd,%refererquot,%referer,%uaquot,%ua,%other

3.2+ for tags %gzipin,%gzipout

4.0+ for tags %gzipratio,%syslog

4.1+ for tag %virtualname

5.6+ for tag %deflateratio

6.1+ for tag %time4

6.2+ for tag %time3

```
# Enter here your log format (Must agree with your web server. See setup
# instructions in README.txt to know how to configure your web server to have
# the required log format).
# Possible values: 1,2,3,4 or "your_own_personalized_log_format"
# 1 - Apache or Lotus Notes/Domino native combined log format (NCSA combined/XLF/ELF log format)
# 2 - IIS or ISA format (IIS W3C log format). See FAQ-COM115 For ISA.
# 3 - Webstar native log format.
# 4 - Apache or Squid native common log format (NCSA common/CLF log format)
# With LogFormat=4, some features (browsers, os, keywords...) can't work.
# "your_own_personalized_log_format" = If your log is ftp, mail or other format,
# you must use following keys to define the log format string (See FAQ for
# ftp, mail or exotic web log format examples):
# %host Client hostname or IP address (or Sender host for mail log)
# %host_r Receiver hostname or IP address (for mail log)
# %lognamequot Authenticated login/user with format: "alex"
# %logname Authenticated login/user with format: alex
# %time1 Date and time with format: [dd/mon/yyyy:hh:mm:ss +0000] or [dd/mon/yyyy:hh:mm:ss]
# %time2 Date and time with format: yyyy-mm-dd hh-mm-ss
# %time3 Date and time with format: Mon dd hh:mm:ss or Mon dd hh:mm:ss yyyy
# %time4 Date and time with unix timestamp format: dddddddddd
# %methodurl Method and URL with format: "GET /index.html HTTP/x.x"
# %methodurlnoproto Method and URL with format: "GET /index.html"
# %method Method with format: GET
# %url URL only with format: /index.html
# %query Query string (used by URLWithQuery option)
# %code Return code status (with format for web log: 999)
# %bytesd Size of document in bytes
# %refererquot Referer page with format: "http://from.com/from.htm"
# %referer Referer page with format: http://from.com/from.htm
# %uaquot User agent with format: "Mozilla/4.0 (compatible, ...)"
# %ua User agent with format: Mozilla/4.0_(compatible...)
# %gzipin mod_gzip compression input bytes: In:XXX
# %gzipout mod_gzip compression output bytes & ratio: Out:YYY:ZZpct.
# %gzipratio mod_gzip compression ratio: ZZpct.
# %deflateratio mod_deflate compression ratio with format: (ZZ)
# %email EMail sender (for mail log)
# %email_r EMail receiver (for mail log)
# %virtualname Web sever virtual hostname. Use this tag when same log
# contains data of several virtual web servers. AWStats
# will discard records not in SiteDomain nor HostAliases
# %cluster If log file is provided from several computers (merged by
# logresolvemerge.pl), this tag define field of cluster id.
# %extraX Another field that you plan to use for building a
# personalized report with ExtraSection feature (See later).
# If your log format has some fields not included in this list, use
# %other Means another field
# %otherquot Means another not used double quoted field
#
# Examples for Apache combined logs (following two examples are equivalent):
# LogFormat = 1
# LogFormat = "%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot"
#
# Example for IIS:
# LogFormat = 2
#
LogFormat=1
```

LogSeparator

Version : 5.0+

If your log field's separator is not a space, you can change this parameter.
This parameter is not used if LogFormat is a predefined value (1,2,3,4)
Example: " "
Example: "\t"
Example: "|"
Default: " "

LogSeparator=" "

DNSLookup

Version : 1.0+ (5.0+ for value 2)

If you want to have hosts reported by name instead of ip address, AWStats
need to make reverse DNS lookups (if not already done in your log file).
With DNSLookup to 0, all hosts will be reported by their IP addresses and
not by the full hostname of visitors (except if names are already available
in log file).
If you want/need to set DNSLookup to 1, don't forget that this will reduce
dramatically AWStats update process speed. Do not use on large web sites.
Note: Reverse DNS lookup is done on IPv4 only (Enable ipv6 plugin for IPv6).
Note: Result of DNS Lookup can be used to build the Country report. However
it is highly recommended to enable the plugin 'geoipfree' or 'geoip' to
have an accurate Country report with no need of DNS Lookup.
Possible values:
0 – No DNS Lookup
1 – DNS Lookup is fully enabled
2 – DNS Lookup is made only from static DNS cache file (if it exists)
Default: 2

DNSLookup=2

DirData

Version : 1.0+

When AWStats updates its statistics, it stores results of its analysis in
files (AWStats database). All those files are written in the directory
defined by the "DirData" parameter. Set this value to the directory where
you want AWStats to save its database and working files into.
Warning: If you want to be able to use the "AllowToUpdateStatsFromBrowser"
feature (see later), you need write permissions by webserver user on this
directory.
Example: "/var/lib/awstats"
Example: "./data"
Example: "C:/awstats_data_dir"
Default: "." (means same directory as awstats.pl)

DirData="."

DirCgi

Version : 1.0+

Relative or absolute web URL of your awstats cgi-bin directory.
This parameter is used only when AWStats is ran from command line
with -output option (to generate links in HTML reported page).
Example: "/awstats"
Default: "/cgi-bin" (means awstats.pl is in "/yourwwwroot/cgi-bin")

DirCgi="/cgi-bin"

DirIcons

Version : 1.0+

```
# Relative or absolute web URL of your awstats icon directory.
# If you build static reports ("... -output > outputpath/output.html"), enter
# path of icon directory relative to the output directory 'outputpath'.
# Example: "/awstatsicons"
# Example: "../icon"
# Default: "/icon" (means you must copy icon directories in "/mywwwroot/icon")
#
DirIcons="/icon"
```

SiteDomain

Version : 3.2+

```
# "SiteDomain" must contain the main domain name or the main intranet web
# server name used to reach the web site.
# If you share the same log file for several virtual web servers, this
# parameter is used to tell AWStats to filter record that contains records for
# this virtual host name only (So check that this virtual hostname can be
# found in your log file and use a personalized log format that include the
# %virtualname tag).
# But for multi hosting a better solution is to have one log file for each
# virtual web server. In this case, this parameter is only used to generate
# full URL's links when ShowLinksOnUrl option is set to 1.
# If analysing mail log, enter here the domain name of mail server.
# Example: "myintranetserver"
# Example: "www.domain.com"
# Example: "ftp.domain.com"
# Example: "domain.com"
#
SiteDomain=""
```

HostAliases

Version : 1.0+ (5.6+ for REGEX syntax)

```
# Enter here all other possible domain names, addresses or virtual host
# aliases someone can use to access your site. Try to keep only the minimum
# number of possible names/addresses to have the best performances.
# You can repeat the "SiteDomain" value in this list.
# This parameter is used to analyze referer field in log file and to help
# AWStats to know if a referer URL is a local URL of same site or an URL of
# another site.
# Note: Use space between each value.
# Note: You can use regular expression values writing value with REGEX[value].
# Example: "www.myserver.com localhost 127.0.0.1 REGEX[\\.mydomain\\. (net|org)$]"
#
HostAliases="localhost 127.0.0.1 REGEX[^\\.myserver\\.com$]"
```

AllowToUpdateStatsFromBrowser

Version : 3.0+

```
# When this parameter is set to 1, AWStats add a button on report page to
# allow to "update" statistics from a web browser. Warning, when "update" is
# made from a browser, AWStats is ran as a CGI by the web server user
# defined in your web server (user "nobody" by default with Apache, "IUSR_XXX"
# with IIS), so the "DirData" directory and all already existing history files
# (awstatsMMYYYY[.xxx].txt) must be writable by this user. Change permissions
# if required.
# Warning: Update process can be long so you might experience "time out"
# browser errors if you don't launch AWStats enough frequently.
# When set to 0, update is only made when AWStats is ran from the command
# line interface (or a task scheduler).
# Possible values: 0 or 1
# Default: 0
```

```
#
AllowToUpdateStatsFromBrowser=0
```

AllowFullYearView

Version : 5.9+

```
# AWStats save and sort its database on a month basis, this allows to build
# build a report quickly. However, if you choose the --month=all from command
# line or value '-Year-' from CGI combo form to have a report for all year,
# AWStats needs to reload all data for full year, and resort them completely,
# requiring a large amount of time, memory and CPU. This might be a problem
# for web hosting providers that offer AWStats for large sites on shared
# servers to non CPU cautious customers.
# For this reason, the 'full year' is only enable on Command Line by default.
# You can change this by setting this parameter to 0, 1, 2 or 3.
# Possible values:
# 0 - Never allowed
# 1 - Allowed on CLI only, --Year- value in combo is not visible
# 2 - Allowed on CLI only, --Year- value in combo is visible but not allowed
# 3 - Possible on CLI and CGI
# Default: 2
#
AllowFullYearView=2
```

EnableLockForUpdate

Version : 5.0+

```
# When the update process run, AWStats can set a lock file in TEMP or TMP
# directory. This lock is to avoid to have 2 update processes running at the
# same time to prevent unknown conflicts problems and avoid DoS attacks when
# AllowToUpdateStatsFromBrowser is set to 1.
# Because, when you use lock file, you can experience sometimes problems in
# lock file not correctly removed (killed process for example requires that
# you remove the file manually), this option is not enabled by default (Do
# not enable this option with no console server access).
# Possible values: 0 or 1
# Default: 0
#
EnableLockForUpdate=0
```

DNSStaticCacheFile

Version : 5.0+

```
# AWStats can do reverse DNS lookups through a static DNS cache file that was
# previously created manually. If no path is given in static DNS cache file
# name, AWStats will search DirData directory. This file is never changed.
# This option is not used if DNSLookup=0.
# Note: DNS cache file format is 'minsince1970 ipaddress resolved_hostname'
# or just 'ipaddress resolved_hostname'
# Example: "/mydnscachedir/dnscache"
# Default: "dnscache.txt"
#
DNSStaticCacheFile="dnscache.txt"
```

DNSLastUpdateCacheFile

Version : 5.0+

```
# AWStats can do reverse DNS lookups through a DNS cache file that was created
# by a previous run of AWStats. This file is erased and recreated after each
# statistics update process. You don't need to create and/or edit it.
# AWStats will read and save this file in DirData directory.
# This option is used only if DNSLookup=1.
# Note: If a DNSStaticCacheFile is available, AWStats will check for DNS
# lookup in DNSLastUpdateCacheFile after checking into DNSStaticCacheFile.
```

```
# Example: "/mydnscachedir/dnscachelastupdate"
# Default: "dnscachelastupdate.txt"
#
DNSLastUpdateCacheFile="dnscachelastupdate.txt"
```

SkipDNSLookupFor

Version : 3.0+ (5.6+ for REGEX syntax)

```
# You can specify specific IP addresses that should NOT be looked up in DNS.
# This option is used only if DNSLookup=1.
# Note: Use space between each value.
# Note: You can use regular expression values writing value with REGEX[value].
# Change : Effective for new updates only
# Example: "123.123.123.123 REGEX[^192\.168\.]"
# Default: ""
#
SkipDNSLookupFor=""
```

AllowAccessFromWebToAuthenticatedUsersOnly

Version : 4.0+

```
# The following two parameters allow you to protect a config file from being
# read by AWStats when called from a browser if web user has not been
# authenticated. Your AWStats program must be in a web protected "realm" (With
# Apache, you can use .htaccess files to do so. With other web servers, see
# your server setup manual).
# Change : Effective immediatly
# Possible values: 0 or 1
# Default: 0
#
AllowAccessFromWebToAuthenticatedUsersOnly=0
```

AllowAccessFromWebToFollowingAuthenticatedUsers

Version : 4.0+

```
# This parameter gives the list of all authorized authenticated users to view
# statistics for this domain/config file. This parameter is used only if
# AllowAccessToAuthenticatedUsersOnly is set to 1.
# Change : Effective immediatly
# Example: "user1 user2"
# Default: ""
#
AllowAccessFromWebToFollowingAuthenticatedUsers=""
```

AllowAccessFromWebToFollowingIPAddresses

Version : 5.0+

```
# When this parameter is defined to something, the IP address of the user that
# read its statistics from a browser (when AWStats is used as a CGI) is
# checked and must match one of the IP address values or ranges.
# Change : Effective immediatly
# Example: "127.0.0.1 123.123.123.1-123.123.123.255"
# Default: ""
#
AllowAccessFromWebToFollowingIPAddresses=""
```

CreateDirDataIfNotExists

Version : 4.0+

```
# If the "DirData" directory (see above) does not exists, AWStats return an
# error. However, you can ask AWStats to create it.
# This option can be used by some Web Hosting Providers that has defined a
# dynamic value for DirData (for example DirData="/home/___REMOTE_USER___") and
```

```
# don't want to have to create a new directory each time they add a new user.
# Change : Effective immediatly
# Possible values: 0 or 1
# Default: 0
#
CreateDirDataIfNotExists=1
```

BuildHistoryFormat

Version : 6.0+

```
# You can choose in which format the Awstats history database is saved.
# Note: Using "xml" format make AWStats building database files three times
# larger than using "text" format.
# Change : Database format is switched after next update
# Possible values: text or xml
# Default: text
#
BuildHistoryFormat=text
```

BuildReportFormat

Version : 6.0+

```
# If you prefer having the report output pages be built as XML compliant pages
# instead of simple HTML pages, you can set this to 'xhtml' (May not works
# properly with old browsers).
# Possible values: html or xhtml
# Default: html
#
BuildReportFormat=html
```

SaveDatabaseFilesWithPermissionsForEveryone

Version : 4.0+

```
# AWStats databases can be updated from command line or from a browser (when
# used as a cgi program). So AWStats database files need write permission
# for both command line user and default web server user (nobody for Unix,
# IUSR_xxx for IIS/Windows,...).
# To avoid permission's problems between update process (run by an admin user)
# and CGI process (ran by a low level user), AWStats can save its database
# files with read and write permissions for everyone.
# By default, AWStats keep default user permissions on updated files. If you
# set AllowToUpdateStatsFromBrowser to 1, you can change this parameter to 1.
# Change : Effective for new updates only
# Possible values: 0 or 1
# Default: 0
#
SaveDatabaseFilesWithPermissionsForEveryone=0
```

PurgeLogFile

Version : 2.23+

```
# AWStats can purge log file, after analyzing it. Note that AWStats is able
# to detect new lines in a log file, to process only them, so you can launch
# AWStats as often as you want, even with this parameter to 0.
# With 0, no purge is made, so you must use a scheduled task or a web server
# that make this purge frequently.
# With 1, the purge of the log file is made each time AWStats is ran.
# This parameter doesn't work with IIS (This web server doesn't let its log
# file to be purged).
# Possible values: 0 or 1
# Default: 0
#
PurgeLogFile=0
```

ArchiveLogRecords

Version : 2.1+ (6.4+ to use tags for suffix)

```
# When PurgeLogFile is setup to 1, AWStats will clean your log file after
# processing it. You can however keep an archive file of all processed log
# records by setting this parameter (For example if you want to use another
# log analyzer). The archived log file is saved in "DirData" with name
# awstats_archive.configname[suffix].log
# This parameter is not used if PurgeLogFile=0
# Change : Effective for new updates only
# Possible values: 0, 1, or tags (See LogFile parameter) for suffix
# Example: 1
# Example: %YYYY%MM%DD
# Default: 0
#
ArchiveLogRecords=0
```

KeepBackupOfHistoricFiles

Version : 3.2+

```
# Each time you run the update process, AWStats overwrite the 'historic file'
# for the month (awstatsMMYYYY[.*.txt) with the updated one.
# When write errors occurs (IO, disk full,...), this historic file can be
# corrupted and must be deleted. Because this file contains information of all
# past processed log files, you will loose old stats if removed. So you can
# ask AWStats to save last non corrupted file in a .bak file. This file is
# stored in "DirData" directory with other 'historic files'.
# Possible values: 0 or 1
# Default: 0
#
KeepBackupOfHistoricFiles=0
```

DefaultFile

Version : 1.0+ (5.0+ can accept several values)

```
# Default index page name for your web server.
# Change : Effective for new updates only
# Example: "index.php index.html default.html"
# Default: "index.html"
#
DefaultFile="index.html"
```

SkipHosts

Version : 1.0+ (5.6+ for REGEX syntax)

```
# Do not include access from clients that match following criteria.
# If your log file contains IP addresses in host field, you must enter here
# matching IP addresses criteria.
# If DNS lookup is already done in your log file, you must enter here hostname
# criteria, else enter ip address criteria.
# The opposite parameter of "SkipHosts" is "OnlyHosts".
# Note: Use space between each value.
# Note: You can use regular expression values writing value with REGEX[value].
# Change : Effective for new updates only
# Example: "127.0.0.1 REGEX[^192\168\.] REGEX[^10\0\0\0]"
# Example: "localhost REGEX[^\.\localdomain$]"
# Default: ""
#
SkipHosts=""
```

SkipUserAgents

Version : 5.1+ (5.6+ for REGEX syntax)


```
# Do not include access from clients with a user agent that match following
# criteria. If you want to exclude a robot, you should update the robots.pm
# file instead of this parameter.
# The opposite parameter of "SkipUserAgents" is "OnlyUserAgents".
# Note: Use space between each value. This parameter is not case sensitive.
# Note: You can use regular expression values writing value with REGEX[value].
# Change : Effective for new updates only
# Example: "konqueror REGEX[ua_test_v\d\\.d]"
# Default: ""
#
SkipUserAgents=""
```

SkipFiles

Version : 1.0+ (5.6+ for REGEX syntax)

```
# Use SkipFiles to ignore access to URLs that match one of following entries.
# You can enter a list of not important URLs (like framed menus, hidden pages,
# etc...) to exclude them from statistics. You must enter here exact relative
# URL as found in log file, or a matching REGEX value. Check apply on URL with
# all its query paramaters.
# For example, to ignore /badpage.php, just add "/badpage.php". To ignore all
# pages in a particular directory, add "REGEX[^/directorytoexclude]".
# The opposite parameter of "SkipFiles" is "OnlyFiles".
# Note: Use space between each value. This parameter is or not case sensitive
# depending on URLNotCaseSensitive parameter.
# Note: You can use regular expression values writing value with REGEX[value].
# Change : Effective for new updates only
# Example: "/badpage.php /page.php?param=x REGEX[^/excludedirectory]"
# Default: ""
#
SkipFiles=""
```

SkipReferrersBlackList

Version : 6.5+

```
# Use SkipReferrersBlackList if you want to exclude records coming from a SPAM
# referrer. Parameter must receive a local file name containing rules applied
# on referrer field. If parameter is empty, no filter is applied.
# An example of such a file is available in lib/blacklist.txt
# You can download updated version at http://www.jayallen.org/comment\_spam/
# Change : Effective for new updates only
# Example: "/mylibpath/blacklist.txt"
# Default: ""

# WARNING!! Using this feature make AWStats running very slower (5 times slower
# with black list file provided with AWStats !
#
SkipReferrersBlackList=""
```

OnlyHosts

Version : 5.2+ (5.6+ for REGEX syntax)

```
# Include in stats, only accesses from hosts that match one of following
# entries. For example, if you want AWStats to filter access to keep only
# stats for visits from particular hosts, you can add those hosts names in
# this parameter.
# If DNS lookup is already done in your log file, you must enter here hostname
# criteria, else enter ip address criteria.
# The opposite parameter of "OnlyHosts" is "SkipHosts".
# Note: Use space between each value. This parameter is not case sensitive.
# Note: You can use regular expression values writing value with REGEX[value].
# Change : Effective for new updates only
# Example: "127.0.0.1 REGEX[^192\\.168\\.] REGEX[^10\\.0\\.0\\.]"
# Default: ""
#
```

OnlyHosts=""

OnlyUserAgents

Version : 5.8+

Include in stats, only accesses from user agent that match one of following
entries. For example, if you want AWStats to filter access to keep only
stats for visits from particular browsers, you can add their user agents
string in this parameter.
The opposite parameter of "OnlyUserAgents" is "SkipUserAgents".
Note: Use space between each value. This parameter is not case sensitive.
Note: You can use regular expression values writing value with REGEX[value].
Change : Effective for new updates only
Example: "msie"
Default: ""

OnlyUserAgents=""

OnlyUsers

Version : 6.8+

Include in stats, only accesses from authenticated users that match one of
following entries. For example, if you want AWStats to filter access to keep
only stats for authenticated users, you can add those users names in
this parameter. Useful for statistics for per user ftp logs.
Note: Use space between each value. This parameter is not case sensitive.
Note: You can use regular expression values writing value with REGEX[value].
Change : Effective for new updates only
Example: "john bob REGEX[^testusers]"
Default: ""

OnlyUsers=""

OnlyFiles

Version : 3.0+ (5.6+ for REGEX syntax)

Include in stats, only accesses to URLs that match one of following entries.
For example, if you want AWStats to filter access to keep only stats that
match a particular string, like a particular directory, you can add this
directory name in this parameter.
The opposite parameter of "OnlyFiles" is "SkipFiles".
Note: Use space between each value. This parameter is or not case sensitive
depending on URLNotCaseSensitive parameter.
Note: You can use regular expression values writing value with REGEX[value].
Change : Effective for new updates only
Example: "REGEX[marketing_directory] REGEX[officeV.*\.(csv|sxw)\$]"
Default: ""

OnlyFiles=""

NotPageList

Version : 3.2+

Add here a list of kind of url (file extension) that must be counted as
"Hit only" and not as a "Hit" and "Page/Download". You can set here all
images extensions as they are hit downloaded that must be counted but they
are not viewed pages. URLs with such extensions are not included in the TOP
Pages/URL report.
Note: If you want to exclude particular URLs from stats (No Pages and no
Hits reported), you must use SkipFiles parameter.
Example: "css js class gif jpg jpeg png bmp rss xml swf zip arj gz z wav mp3 wma mpg"
Example: ""
Default: "css js class gif jpg jpeg png bmp rss xml swf"
#

ValidHTTPCodes

Version : 4.0+

```
# By default, AWStats considers that records found in log file are successful
# hits if HTTP code returned by server is a valid HTTP code (200 and 304).
# Any other code are reported in HTTP error chart.
# However in some specific environment, with web server HTTP redirection,
# you can choose to also accept other codes.
# Example: "200 304 302 305"
# Default: "200 304"
#
ValidHTTPCodes="200 304"
```

This is examples of current HTTP codes

```
#[Miscellaneous successes]
"2xx", "[Miscellaneous successes]",
"200", "OK", # HTTP request OK
"201", "Created",
"202", "Request recorded, will be executed later",
"203", "Non-authoritative information",
"204", "Request executed",
"205", "Reset document",
"206", "Partial Content",
#[Miscellaneous redirections]
"3xx", "[Miscellaneous redirections]",
"300", "Multiple documents available",
"301", "Moved Permanently",
"302", "Found",
"303", "See other document",
"304", "Not Modified since last retrieval", # HTTP request OK
"305", "Use proxy",
"306", "Switch proxy",
"307", "Document moved temporarily",
#[Miscellaneous client/user errors]
"4xx", "[Miscellaneous client/user errors]",
"400", "Bad Request",
"401", "Unauthorized",
"402", "Payment required",
"403", "Forbidden",
"404", "Document Not Found",
"405", "Method not allowed",
"406", "Document not acceptable to client",
"407", "Proxy authentication required",
"408", "Request Timeout",
"409", "Request conflicts with state of resource",
"410", "Document gone permanently",
"411", "Length required",
"412", "Precondition failed",
"413", "Request too long",
"414", "Requested filename too long",
"415", "Unsupported media type",
"416", "Requested range not valid",
"417", "Failed",
#[Miscellaneous server errors]
"5xx", "[Miscellaneous server errors]",
"500", "Internal server Error",
"501", "Not implemented",
"502", "Received bad response from real server",
"503", "Server busy",
"504", "Gateway timeout",
"505", "HTTP version not supported",
"506", "Redirection failed",
#[Unknown]
"xxx", "[Unknown]"
```

ValidSMTPCodes

Version : 5.0+

By default, AWStats considers that records found in mail log file are
successful mail transfers if field that represent return code in analyzed
log file match values defined by this parameter.
Change : Effective for new updates only
Example: "1 250 200"
Default: "1 250"

ValidSMTPCodes="1 250"

AuthenticatedUsersNotCaseSensitive

Version : 5.3+

Some web servers on some Operating systems (IIS–Windows) considers that a
login with same value but different case are the same login. To tell AWStats
to also considers them as one, set this parameter to 1.
Possible values: 0 or 1
Default: 0

AuthenticatedUsersNotCaseSensitive=0

URLNotCaseSensitive

Version : 5.1+

Some web servers on some Operating systems (IIS–Windows) considers that two
URLs with same value but different case are the same URL. To tell AWStats to
also considers them as one, set this parameter to 1.
Possible values: 0 or 1
Default: 0

URLNotCaseSensitive=0

URLWithAnchor

Version : 5.4+

Keep or remove the anchor string you can find in some URLs.
Possible values: 0 or 1
Default: 0

URLWithAnchor=0

URLQuerySeparators

Version : 5.2+

In URL links, "?" char is used to add parameter's list in URLs. Syntax is:
/mypage.html?param1=value1
However, some servers/sites have also others chars to isolate dynamic part of
their URLs. You can complete this list with all such characters.
Example: "?;,"
Default: "?;"

URLQuerySeparators="?;"

URLWithQuery

Version : 3.2+

Keep or remove the query string to the URL in the statistics for individual
pages. This is primarily used to differentiate between the URLs of dynamic

```
# pages. If set to 1, mypage.html?id=x and mypage.html?id=y are counted as two
# different pages.
# Warning, when set to 1, memory required to run AWStats is dramatically
# increased if you have a lot of changing URLs (for example URLs with a random
# id inside). Such web sites should not set this option to 1 or use seriously
# the next parameter URLWithQueryWithoutFollowingParameters.
# Possible values:
# 0 – URLs are cleaned from the query string (ie: "/mypage.html")
# 1 – Full URL with query string is used (ie: "/mypage.html?p=x&q=y")
# Default: 0
#
URLWithQuery=0
```

URLWithQueryWithOnlyFollowingParameters

Version : 6.0+

```
# When URLWithQuery is on, you will get the full URL with all parameters in
# URL reports. But among those parameters, sometimes you don't need a
# particular parameter because it does not identify the page or because it's
# a random ID changing for each access even if URL points to same page. In
# such cases, it is highly recommended to ask AWStats to keep only parameters
# you need (if you know them) before counting, manipulating and storing URL.
# Enter here list of wanted parameters. For example, with "param", one hit on
# /mypage.cgi?param=abc&id=Yo4UomP9d and /mypage.cgi?param=abc&id=MU8fdx13r
# will be reported as 2 hits on /mypage.cgi?param=abc
# This parameter is not used when URLWithQuery is 0 and can't be used with
# URLWithQueryWithoutFollowingParameters.
# Change : Effective for new updates only
# Example: "param"
# Default: ""
#
URLWithQueryWithOnlyFollowingParameters=""
```

URLWithQueryWithoutFollowingParameters

Version : 5.1+

```
# When URLWithQuery is on, you will get the full URL with all parameters in
# URL reports. But among those parameters, sometimes you don't need a
# particular parameter because it does not identify the page or because it's
# a random ID changing for each access even if URL points to same page. In
# such cases, it is highly recommended to ask AWStats to remove such parameters
# from the URL before counting, manipulating and storing URL. Enter here list
# of all non wanted parameters. For example, if you enter "id", one hit on
# /mypage.cgi?param=abc&id=Yo4UomP9d and /mypage.cgi?param=abc&id=MU8fdx13r
# will be reported as 2 hits on /mypage.cgi?param=abc
# This parameter is not used when URLWithQuery is 0 and can't be used with
# URLWithQueryWithOnlyFollowingParameters.
# Change : Effective for new updates only
# Example: "PHPSESSID jsessionid"
# Default: ""
#
URLWithQueryWithoutFollowingParameters=""
```

URLReferrerWithQuery

Version : 5.1+

```
# Keep or remove the query string to the referrer URL in the statistics for
# external referrer pages. This is used to differentiate between the URLs of
# dynamic referrer pages. If set to 1, mypage.html?id=x and mypage.html?id=y
# are counted as two different referrer pages.
# Possible values:
# 0 – Referrer URLs are cleaned from the query string (ie: "/mypage.html")
# 1 – Full URL with query string is used (ie: "/mypage.html?p=x&q=y")
# Default: 0
#
```

WarningMessages

Version : 1.0+

AWStats can detect setup problems or show you important informations to have
a better use. Keep this to 1, except if AWStats says you can change it.
Possible values: 0 or 1
Default: 1

WarningMessages=1

ErrorMessages

Version : 5.2+

When an error occurs, AWStats output a message related to errors. If you
want (in most cases for security reasons) to have no error messages, you
can set this parameter to your personalized generic message.
Example: "An error occured. Contact your Administrator"
Default: ""

ErrorMessages=""

DebugMessages

Version : 5.2+

AWStat can be run with debug=x parameter to output various informations
to help in debugging or solving troubles. If you want to allow this (not
enabled by default for security reasons), set this parameter to 0.
Change : Effective immediatly
Possible values: 0 or 1
Default: 0

DebugMessages=0

NbOfLinesForCorruptedLog

Version : 3.2+

To help you to detect if your log format is good, AWStats report an error
if all the first NbOfLinesForCorruptedLog lines have a format that does not
match the LogFormat parameter.
However, some worm virus attack on your web server can result in a very high
number of corrupted lines in your log. So if you experience awstats stop
because of bad virus records at the beginning of your log file, you can
increase this parameter (very rare).
Default: 50

NbOfLinesForCorruptedLog=50

SplitSearchString

Version : 2.24 – 4.0 (deprecated since 4.1)

This parameter has been removed since 4.1.

AWStats 4.1+ supports both keywords AND keyphrases by default with no need of any parameter.

WrapperScript

Version : 4.0+

For some particular integration needs, you may want to have CGI links to
point to another script than awstats.pl.
Use the name of this script in WrapperScript parameter.


```
# Example: "awstatslauncher.pl"
# Default: ""
#
WrapperScript=""
```

DecodeUA

Version : 5.0+

```
# DecodeUA must be set to 1 if you use Roxen web server. This server converts
# all spaces in user agent field into %20. This make the AWStats robots, os
# and browsers detection fail in some cases. Just change it to 1 if and only
# if your web server is Roxen.
# Possible values: 0 or 1
# Default: 0
#
DecodeUA=0
```

MiscTrackerUrl

Version : 5.6+

```
# MiscTrackerUrl can be used to make AWStats able to detect some miscellaneous
# things, that can not be tracked on other way like:
# - Screen size
# - Screen color depth
# - Java enabled
# - Macromedia Director plugin
# - Macromedia Shockwave plugin
# - Realplayer G2 plugin
# - QuickTime plugin
# - Mediaplayer plugin
# - Acrobat PDF plugin
# To enable all this features, you must copy the awstats_misc_tracker.js file
# into a /js/ directory stored in your web document root and add the following
# HTML code at the end of your index page (before </BODY>) :
# <script language=javascript src="/js/awstats_misc_tracker.js"></script>
# If code is not added in index page, all this detection capabilities will be
# disabled. You must also check that ShowScreenSizeStats and ShowMiscStats
# parameters are set to 1 to make results appear in report page.
# If you want to use another directory than /js/, you must also change the
# awstatsmisctrackerurl variable into the awstats_misc_tracker.js file.
# Change : Effective for new updates only.
# Possible value: Full URL of javascript tracker file added in HTML code
# Default: "/js/awstats_misc_tracker.js"
#
MiscTrackerUrl="/js/awstats_misc_tracker.js"
```

LevelFor

Version : 4.0+

6.0+ for LevelForFileTypesDetection, LevelForSearchEnginesDetection, LevelForKeywordsDetection, LevelForWormsDetection

```
# Following values allows you to define accuracy of AWStats entities (robots,
# browsers, os, referers, file types) detection.
# It might be a good idea for large web sites or ISP that provides AWStats to
# high number of customers, to set this parameter to 1 (or 0), instead of 2.
# Possible values:
# 0 = No detection,
# 1 = Medium/Standard detection
# 2 = Full detection
# Change : Effective for new updates only
# Default: 2 (0 for LevelForWormsDetection)
#
LevelForBrowsersDetection=2 # 0 disables Browsers detection.
# 2 reduces AWStats speed by 2%
LevelForOSDetection=2 # 0 disables OS detection.
```

2 reduces AWStats speed by 3%
LevelForRefererAnalyze=2 # 0 disables Origin detection.
2 reduces AWStats speed by 14%
LevelForRobotsDetection=2 # 0 disables Robots detection.
2 reduces AWStats speed by 2.5%
LevelForSearchEnginesDetection=2 # 0 disables Search engines detection.
2 reduces AWStats speed by 9%
LevelForKeywordsDetection=2 # 0 disables Keyphrases/Keywords detection.
2 reduces AWStats speed by 1%
LevelForFileTypesDetection=2 # 0 disables File types detection.
2 reduces AWStats speed by 1%
LevelForWormsDetection=0 # 0 disables Worms detection.
2 reduces AWStats speed by 15%

UseFramesWhenCGI

Version : 5.0+

When you use AWStats as a CGI, you can have the reports shown in HTML views.
Frames are only available for report viewed dynamically. When you build
pages from command line, this option is not used and no frames are built.
Possible values: 0 or 1
Default: 1

UseFramesWhenCGI=1

DetailedReportsOnNewWindows

Version : 4.1+ (5.0+ for value 2)

This parameter ask your browser to open detailed reports into a different
window than the main page.
Possible values:
0 – Open all in same browser window
1 – Open detailed reports in another window except if using frames
2 – Open always in a different window even if reports are framed
Default: 1

DetailedReportsOnNewWindows=1

Expires

Version : 3.1+

You can add, in the HTML report page, a cache lifetime (in seconds) that
will be returned to browser in HTTP header answer by server.
This parameter is not used when report are built with –staticlinks option.
Example: 3600
Default: 0

Expires=0

MaxRowsInHTMLOutput

Version : 4.0+

To avoid too large web pages, you can ask AWStats to limit number of rows of
all reported charts to this number when no other limit apply.
Default: 1000

MaxRowsInHTMLOutput=1000

Lang

Version : 2.1+

Set your primary language.
Possible value:

```
# Albanian=al, Bosnian=ba, Bulgarian=bg,
# Chinese (Taiwan)=tw, Chinese (Simplified)=cn, Czech=cz,
# Danish=dk, Dutch=nl, English=en, Estonian=et, Finnish=fi, French=fr,
# German=de, Greek=gr, Hebrew=he, Hungarian=hu, Indonesian=id, Italian=it,
# Japanese=jp, Korean=kr, Latvian=lv, Norwegian (Nynorsk)=nn,
# Norwegian (Bokmal)=nb, Polish=pl, Portuguese=pt, Portuguese (Brazilian)=br,
# Romanian=ro, Russian=ru, Serbian=sr, Slovak=sk, Spanish=es,
# Spanish (Catalan)=es_cat, Swedish=se, Turkish=tr, Ukrainian=ua, Welsh=wlk.
# First available language accepted by browser=auto
# Default: "auto"
#
Lang="auto"
```

DirLang

Version : 2.1+

```
# Set the location of language files.
# Example: "/usr/share/awstats/lang"
# Default: "./lang" (means lang directory is in same location than awstats.pl)
#
DirLang="./lang"
```

Show...

Version :

3.2 – 5.0 for ShowCompressionStats (deprecated since 5.1, use code C with ShowFileTypesStats instead)

3.2 – 5.3 for ShowHeader (deprecated since 5.4)

3.2+ for ShowMenu, ShowMonthStats, ShowDaysOfWeekStats, ShowHoursStats,
ShowDomainsStats, ShowHostsStats, ShowAuthenticatedUsers, ShowRobotsStats,
ShowPagesStats, ShowFileTypesStats, ShowFileSizesStats, ShowBrowsersStats,
ShowOSStats, ShowOriginStats, ShowKeyphrasesStats, ShowKeywordsStats, ShowHTTPErrorsStats
4.1+ for ShowSessionsStats, ShowKeywordsStats

5.1+ for all letters codes

5.5+ for ShowDaysOfMonthStats

5.6+ for ShowMiscStats, ShowSTMPErrorsStats

5.8+ for ShowClusterStats

6.0+ for ShowWormsStats

6.4+ for ShowSummary

7.0+ for ShowDownloadsStats

```
# You choose here which reports you want to see in the main page and what you
# want to see in those reports.
```

```
# Possible values:
```

```
# 0 – Report is not shown at all
```

```
# 1 – Report is shown in main page with an entry in menu and default columns
```

```
# XYZ – Report shows column informations defined by code X,Y,Z...
```

```
# X,Y,Z... are code letters among the following:
```

```
# U = Unique visitors
```

```
# V = Visits
```

```
# P = Number of pages
```

```
# H = Number of hits (or mails)
```

```
# B = Bandwith (or total mail size for mail logs)
```

```
# L = Last access date
```

```
# E = Entry pages
```

```
# X = Exit pages
```

```
# C = Web compression (mod_gzip,mod_deflate)
```

```
# M = Average mail size (mail logs)
```

```
#
```

```
# Show menu header with reports' links
```

```
# Possible values: 0 or 1
```

```
# Default: 1
```

```
#
```

```
ShowMenu=1
```

```
# Show monthly summary
```

```
# Context: Web, Streaming, Mail, Ftp
```

```

# Default: UVPHB, Possible column codes: UVPHB
ShowSummary=UVPHB

# Show monthly chart
# Context: Web, Streaming, Mail, Ftp
# Default: UVPHB, Possible column codes: UVPHB
ShowMonthStats=UVPHB

# Show days of month chart
# Context: Web, Streaming, Mail, Ftp
# Default: VPHB, Possible column codes: VPHB
ShowDaysOfMonthStats=VPHB

# Show days of week chart
# Context: Web, Streaming, Mail, Ftp
# Default: PHB, Possible column codes: PHB
ShowDaysOfWeekStats=PHB

# Show hourly chart
# Context: Web, Streaming, Mail, Ftp
# Default: PHB, Possible column codes: PHB
ShowHoursStats=PHB

# Show domains/country chart
# Context: Web, Streaming, Mail, Ftp
# Default: PHB, Possible column codes: PHB
ShowDomainsStats=PHB

# Show hosts chart
# Context: Web, Streaming, Mail, Ftp
# Default: PHBL, Possible column codes: PHBL
ShowHostsStats=PHBL

# Show authenticated users chart
# Context: Web, Streaming, Ftp
# Default: 0, Possible column codes: PHBL
ShowAuthenticatedUsers=0

# Show robots chart
# Context: Web, Streaming
# Default: HBL, Possible column codes: HBL
ShowRobotsStats=HBL

# Show worms chart
# Context: Web, Streaming
# Default: 0 (If set to other than 0, see also LevelForWormsDetection), Possible column codes: HBL
ShowWormsStats=0

# Show email senders chart (For use when analyzing mail log files)
# Context: Mail
# Default: 0, Possible column codes: HBML
ShowEMailSenders=0

# Show email receivers chart (For use when analyzing mail log files)
# Context: Mail
# Default: 0, Possible column codes: HBML
ShowEMailReceivers=0

# Show session chart
# Context: Web, Streaming, Ftp
# Default: 1, Possible column codes: None
ShowSessionsStats=1

# Show pages-url chart.
# Context: Web, Streaming, Ftp
# Default: PBEX, Possible column codes: PBEX
ShowPagesStats=PBEX

# Show file types chart.

```

Context: Web, Streaming, Ftp
Default: HB, Possible column codes: HBC
ShowFileTypesStats=HB

Show file size chart (Not yet available)
Context: Web, Streaming, Mail, Ftp
Default: 1, Possible column codes: None
ShowFileSizesStats=0

Show downloads chart.
Context: Web, Streaming, Ftp
Default: HB, Possible column codes: HB
ShowDownloadsStats=HB

Show operating systems chart
Context: Web, Streaming, Ftp
Default: 1, Possible column codes: None
ShowOSStats=1

Show browsers chart
Context: Web, Streaming
Default: 1, Possible column codes: None
ShowBrowsersStats=1

Show screen size chart
Context: Web, Streaming
Default: 0 (If set to 1, see also MiscTrackerUrl), Possible column codes: None
ShowScreenSizeStats=0

Show origin chart
Context: Web, Streaming
Default: PH, Possible column codes: PH
ShowOriginStats=PH

Show keyphrases chart
Context: Web, Streaming
Default: 1, Possible column codes: None
ShowKeyphrasesStats=1

Show keywords chart
Context: Web, Streaming
Default: 1, Possible column codes: None
ShowKeywordsStats=1

Show misc chart
Context: Web, Streaming
Default: a (See also MiscTrackerUrl parameter), Possible column codes: anjdfqrwp
ShowMiscStats=a

Show http errors chart
Context: Web, Streaming
Default: 1, Possible column codes: None
ShowHTTPErrorsStats=1

Show smtp errors chart (For use when analyzing mail log files)
Context: Mail
Default: 0, Possible column codes: None
ShowSMTPErrorsStats=0

Show the cluster report (Your LogFormat must contains the %cluster tag)
Context: Web, Streaming, Ftp
Default: 0, Possible column codes: PHB
ShowClusterStats=0

AddDataArray...

Version :

5.4+ for AddDataArrayMonthStats,AddDataArrayShowDaysOfWeekStats,AddDataArrayShowHoursStats

5.5+ for AddDataArrayShowDaysOfMonthStats

```
# Some graphical reports are followed by the data array of values.
# If you don't want this array (to reduce report size for example), you can
# set those options to 0.
# Possible values: 0 or 1
# Default: 1
#
# Data array values for the ShowMonthStats report
AddDataArrayMonthStats=1
# Data array values for the ShowDaysOfMonthStats report
AddDataArrayShowDaysOfMonthStats=1
# Data array values for the ShowDaysOfWeekStats report
AddDataArrayShowDaysOfWeekStats=1
# Data array values for the ShowHoursStats report
AddDataArrayShowHoursStats=1
```

IncludeInternalLinksInOriginSection

Version : 6.1+

```
# In the Origin chart, you have stats on where your hits came from. You can
# includes hits on pages that comes from pages of same sites in this chart.
# Possible values: 0 or 1
# Default: 0
#
IncludeInternalLinksInOriginSection=0
```

Max...

Version : 1.0+

```
# This value can be used to choose maximum number of lines shown for each
# particular reporting.
#
# Stats by domains
MaxNbOfDomain = 10
MinHitDomain = 1
# Stats by hosts
MaxNbOfHostsShown = 10
MinHitHost = 1
# Stats by authenticated users
MaxNbOfLoginShown = 10
MinHitLogin = 1
# Stats by robots
MaxNbOfRobotShown = 10
MinHitRobot = 1
# Stats for Downloads
MaxNbOfDownloadsShown = 10
MinHitDownloads = 1
# Stats by pages
MaxNbOfPageShown = 10
MinHitFile = 1
# Stats by OS
MaxNbOfOsShown = 10
MinHitOs = 1
# Stats by browsers
MaxNbOfBrowsersShown = 10
MinHitBrowser = 1
# Stats by screen size
MaxNbOfScreenSizesShown = 5
MinHitScreenSize = 1
# Stats by referers
MaxNbOfRefererShown = 10
MinHitRefer = 1
# Stats for keywords
MaxNbOfKeywordsShown = 10
MinHitKeyword = 1
```


Stats for sender or receiver emails
MaxNbOfEMailsShown = 20
MinHitEMail = 1

FirstDayOfWeek

Version : 3.2+

Choose if you want week to start on sunday or monday
Possible values:
0 – Week start on sunday
1 – Week start on monday
Default: 1

FirstDayOfWeek=1

ShowFlagLinks

Version : 3.2+

List of visible flags with link to other language translations.
See Lang parameter for list of allowed flag/language codes.
If you don't want any flag link, set ShowFlagLinks to "".
This parameter is used only if ShowMenu parameter is set to 1.
Possible values: "" or "language_codes_separated_by_space"
Default: "en es fr it nl es"

ShowFlagLinks="en fr de it nl es"

ShowLinksOnUrl

Version : 3.1+

Each URL shown in stats report views are links you can click.
Possible values: 0 or 1
Default: 1

ShowLinksOnUrl=1

UseHTTPSLinkForUrl

Version : 4.0+

When AWStats build HTML links in its report pages, it starts those link
with "http://". However some links might be HTTPS links, so you can enter
here the root of all your HTTPS links. If all your site is a SSL web site,
just enter "/".
This parameter is not used if ShowLinksOnUrl is 0.
Example: "/shopping"
Example: "/"
Default: ""

UseHTTPSLinkForUrl=""

MaxLengthOfShownURL

Version : 1.0+

Maximum length of URL part shown on stats page (number of characters).
This affects only URL visible text, larger links still work.
Default: 64

MaxLengthOfShownURL=64

ShowLinksToWhols

Version : 4.0 – 5.6 (deprecated since 5.7, replaced by plugin 'hostinfo')

This parameter has been removed since 5.7.
You must enable the plugin 'hostinfo' to get the same result if you were using this parameter.

LinksToWhols

Version : 4.0 – 5.9 (deprecated since 6.0, replaced by plugin 'hostinfo')
This parameter has been removed since 6.0.
This parameter is no more required.

LinksToIPWhols

Version : 5.0 – 5.9 (deprecated since 6.0, replaced by plugin 'hostinfo')
This parameter has been removed since 6.0.
This parameter is no more required.

HTMLHeadSection

Version : 3.2+

You can enter HTML code that will be added at the top of AWStats reports.
Default: ""

HTMLHeadSection=""

HTMLEndSection

Version : 3.2+

You can enter HTML code that will be added at the end of AWStats reports.
Great to add advert ban.
Default: ""

HTMLEndSection=""

Bar...

Version : 1.0+

Value of maximum bar width/height for horizontal/vertical HTML graphics bar.
Default: 260/90

BarWidth = 260
BarHeight = 90

Logo...

Version : 3.1+

You can set Logo and LogoLink to use your own logo.
Logo must be the name of image file (must be in \$DirIcons/other directory).
LogoLink is the expected URL when clicking on Logo.
Default: "awstats_logo1.png"

Logo="awstats_logo1.png"
LogoLink="http://www.awstats.org"

StyleSheet

Version : 5.6+

You can ask AWStats to use a particular CSS (Cascading Style Sheet) to
change its look. To create a style sheet, you can use samples provided with
AWStats in wwwroot/css directory.
Example: "/awstatscss/awstats_bw.css"
Example: "/css/awstats_bw.css"
Default: ""
#

StyleSheet=""

color_...

Version :

3.1 for color_Background,color_TableBGTitle,color_TableTitle,color_TableBG,
color_TableRowTitle,color_TableBGRowTitle,color_TableBorder,color_text,
color_textpercent,color_titletext,color_weekend,color_link,color_hover, color_u,color_v,color_p,color_h,color_k,color_s
4.1 for color_e,color_x
5.0 for color_other

Those colors parameters can be used (if StyleSheet parameter is not used)

to change AWStats look.

Example: color_name="RRGGBB" # RRGGBB is Red Green Blue components in Hex

#

color_Background="FFFFFF" # Background color for main page (Default = "FFFFFF")
color_TableBGTitle="CCCCDD" # Background color for table title (Default = "CCCCDD")
color_TableTitle="000000" # Table title font color (Default = "000000")
color_TableBG="CCCCDD" # Background color for table (Default = "CCCCDD")
color_TableRowTitle="FFFFFF" # Table row title font color (Default = "FFFFFF")
color_TableBGRowTitle="ECECEC" # Background color for row title (Default = "ECECEC")
color_TableBorder="ECECEC" # Table border color (Default = "ECECEC")
color_text="000000" # Color of text (Default = "000000")
color_textpercent="606060" # Color of text for percent values (Default = "606060")
color_titletext="000000" # Color of text title within colored Title Rows (Default = "000000")
color_weekend="EAEAEA" # Color for week-end days (Default = "EAEAEA")
color_link="0011BB" # Color of HTML links (Default = "0011BB")
color_hover="605040" # Color of HTML on-mouseover links (Default = "605040")
color_other="666688" # Color of text for 'other' record in charts (Default = "666688")
color_u="FFB055" # Background color for number of unique visitors (Default = "FFB055")
color_v="F8E880" # Background color for number of visites (Default = "F8E880")
color_p="4477DD" # Background color for number of pages (Default = "4477DD")
color_h="66F0FF" # Background color for number of hits (Default = "66F0FF")
color_k="2EA495" # Background color for number of bytes (Default = "2EA495")
color_s="8888DD" # Background color for number of search (Default = "8888DD")
color_e="CEC2E8" # Background color for number of entry pages (Default = "CEC2E8")
color_x="C1B2E2" # Background color for number of exit pages (Default = "C1B2E2")

LoadPlugin

Version : 5.0+

Add here all plugins file you want to load.

Plugin files must be .pm files stored in 'plugins' directory.

Uncomment LoadPlugin lines to enable a plugin after checking that plugin

required perl modules are installed.

Plugin: PluginName

PluginName description

Perl modules required: ...

#

LoadPlugin="pluginname"

Extra...

Version :

5.2+

5.8 for ExtraSectionFirstColumnFormatX, ExtraSectionAddAverageRowX, ExtraSectionAddSumRowX

You can see the [following page](#) for explanation of all ExtraSection...X directives and how to use them.

ExtraTrackedRowsLimit

Version : 6.1

There is also a global parameter ExtraTrackedRowsLimit that limit the

number of possible rows an ExtraSection can report. This parameter is

here to protect too much memory use when you make a bad setup in your

```
# ExtraSection. It applies to all ExtraSection independently meaning that
# none ExtraSection can report more rows than value defined by ExtraTrackedRowsLimit.
# If you know an ExtraSection will report more rows than its value, you should
# increase this parameter or AWStats will stop with an error.
# Example: 2000
# Default: 500
#
ExtraTrackedRowsLimit=500
```

Include

Version : 5.4+

```
# You can include other config files using the directive with the name of the
# config file.
# This is particularly usefull for users who have a lot of virtual servers, so
# a lot of config files and want to maintain common values in only one file.
# Note that when a variable is defined both in a config file and in an
# included file, AWStats will use the last value read for parameters that
# contains one value and AWStats will concat all values from both files for
# parameters that are lists of value.
#
Include ""
```



Adding extra reports using the ExtraSection feature

The AWStats ExtraSection features are powerful setup options to allow you to add your own report not provided by default with AWStats. You can use it to build special reports, like number of sales for a particular product, marketing reports, counting for a particular user or agent, etc...

Explanation on how to add/edit an Extra report in your config file

Take a look inside the AWStats config file to find the following part:

```
#-----  
# EXTRA SECTIONS  
#-----
```

Read all explanation in config file after this point, they will explain you how to add an Extra report by adding an ExtraSection configuration in your config file,
or just click [here](#) to jump to a copy of this explanation.

Following examples are precious tutorials...

Some examples of ExtraSection setup you can follow to build your own personalized reports:

- [Example 1: Tracking Product orders](#)
- [Example 2: Tracking Bugzilla most frequently viewed bugs](#)
- [Example 3: Tracking Exit clicks](#)
- [Example 4: Tracking aborted download](#)
- [Example 5: Tracking most requested domain aliases](#)
- [Example 6: List of top level 2 path under a directory /mydir](#)

And for more usage examples (like more precise Search Engine Optimization, referrals by domain, ...), you can have a look at the very good page [Antezeta AWStats Enhancements and Extensions page](#).

Example 1: Tracking Product orders

Imagine your web site is an e-store that sells 80 different products. Each of them has an id. Imagine each time, someone make an order for product 49, the order.cgi script or order2.cgi script is called with, in URL query parameter, the id of the product, meanings that you get in your log file a hit that looks like this:

```
GET /cgi-bin/order.cgi?productid=49
```

So this is how you need to setup your ExtraSection to track your product orders:

```
ExtraSectionName1="Product orders"  
ExtraSectionCodeFilter1="200 304"  
ExtraSectionCondition1="URL,\\cgi\\-bin\\order\\.cgi|URL,\\cgi\\-bin\\order2\\.cgi"  
ExtraSectionFirstColumnName1="Product ID"  
ExtraSectionFirstColumnValues1="QUERY_STRING,productid=(^[^  
ExtraSectionFirstColumnFormat1="%s"  
ExtraSectionStatTypes1=PL  
ExtraSectionAddAverageRow1=0  
ExtraSectionAddSumRow1=1  
MaxNbOfExtra1=100  
MinHitExtra1=1
```

Example 2: Tracking Bugzilla most frequently viewed bugs

This is an example on how to setup your ExtraSection:

```
ExtraSectionName1="Bugzilla: Most frequently viewed bugs"
ExtraSectionCodeFilter1="200 304"
ExtraSectionCondition1="URL,\bugzilla\show_bug.cgi"
ExtraSectionFirstColumnTitle1="Bug ID"
ExtraSectionFirstColumnValues1="QUERY_STRING,id=(^
ExtraSectionFirstColumnFormat1="<a href='/bugzilla/show_bug.cgi?id=%s' target=new>%s</a>"
ExtraSectionStatTypes1=PL
ExtraSectionAddAverageRow1=0
ExtraSectionAddSumRow1=1
MaxNbOfExtra1=500
MinHitExtra1=1
```

Example 3: Tracking Exit clicks

AWStats shows you naturally the exit pages. However, you don't know where your visitor goes after exiting your site since clicking on a link that points to an external link will log the viewed page on the external server and not on yours. If you want to track this, you can, using the ExtraSection and the **awredir.pl** tool (provided with AWStats). This tool must be used as a CGI wrapper. When called on a 'A HREF' link, it returns to the browser a redirector to tell it to show the required page. So, to use this script, you must replace HTML href tags that point to external web sites onto your HTML pages from

```
<a href="http://externalsite/pagelinked">Link</a>
```

to

```
<a href="http://yoursite/cgi-bin/awredir.pl?url=http://externalsite/pagelinked">Link</a>
```

For your web visitor, there is no difference. However, this allows you to track clicks done on links onto your web pages that point to external web sites, because an entry will be seen in your own server log like this record:

```
80.1.2.3 - - [01/Jan/2001:16:00:00 -0300] "GET /cgi-bin/awredir.pl?url=http://externalsite/pagelinked HTTP/1.1"
302 70476 "http://yoursite/pagewithlink.html" "FireBird/0.7"
```

Then, you can add in AWStats a chart to track all calls to *awredir.pl* with key values taken from the "url=" parameter. You will get an independent chart, counting all external pages viewed by your visitor after exiting your site.

To have this chart, this is how you must setup your ExtraSection:

```
ExtraSectionName1="Redirected Hit"
ExtraSectionCodeFilter1="302"
ExtraSectionCondition1="URL,\cgi-bin\awredir.pl"
ExtraSectionFirstColumnTitle1="Url"
ExtraSectionFirstColumnValues1="QUERY_STRING,url=(^
ExtraSectionStatTypes1=HL
MaxNbOfExtra1=500
MinHitExtra1=1
ExtraSectionAddSumRow1=1
```

Example 4: Tracking aborted download

Aborted downloads are reported in a log file by a 206 error, so this is how you need to setup your ExtraSection to add a chart for such tracking:

```
ExtraSectionName1="List of aborted download"
ExtraSectionCodeFilter1="206"
ExtraSectionCondition1=""
ExtraSectionFirstColumnTitle1="URL"
ExtraSectionFirstColumnValues1="URL,(.*)"
ExtraSectionStatTypes1=PHB
MaxNbOfExtra1=100
```

Example 5: Tracking most requested domain aliases

You have one website, but this web site has several domains named (for example the same site domain.com can be reached with urls domain.com,www.domain.com,www.otherdomainname.com,www.againadomainname.org,...). You want to know which domain alias is the most used.

The first thing to do is to be sure the domain alias is recorded inside your log file. If you use Apache, you must use a personalized Apache log file that contains the %V tag. For example you can add in your Apache httpd.conf file a new Apache log format (This is the Apache directive, not AWStats, to define an Apache log format that contains the virtual domain):

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %V" combinedv
```

Then check that your Apache CustomLog directives are defined like this:

```
CustomLog pathtoyourlog/yourlog.log combinedv
```

After restarting Apache, your log format should look like this:

```
66.130.77.181 - - [09/Aug/2004:03:01:05 +0200] "GET /index.php HTTP/1.1" 200 1473 "-" "Firefox 1.0"
www.otherdomainname.com
```

When your web server log file contains the domain alias, you can now setup AWStats to use it. For this use a personalised AWStats log format and use the %extra1 tag at the same position where the domain alias is. For example, for your combinedv Apache log format, we will use:

```
LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot %extra1"
```

Every tag defined by name *extraZ* (Z is a number, you can use as many tags as you need) can be used in any ExtraSection to extract the parameter. You can use the name *extraZ* as a criteria in the *ExtraSectionFirstColumnValuesX* parameter (X is number of the extra report, you can add as many report as you need) to tell AWStats to use value in log file at the place of the tag, as the key for your report. For example:

```
ExtraSectionName1="Domains aliases"
ExtraSectionCodeFilter1="200 304"
ExtraSectionCondition1=""
ExtraSectionFirstColumnTitle1="Domain alias"
ExtraSectionFirstColumnValues1="extra1,([^\n\r])"
ExtraSectionFirstColumnFormat1="%s"
ExtraSectionStatTypes1=HL
ExtraSectionAddAverageRow1=0
ExtraSectionAddSumRow1=1
MaxNbOfExtra1=20
MinHitExtra1=1
```

And result will be:

Domains aliases

Domain alias	Hits	Last visit
www.domain.com	1757131	08 August 2004 – 13:01
www.otherdomainname.com	98518	08 August 2004 – 12:54
domain.com	19107	08 August 2004 – 12:42
www.againadomainname.org	7609	08 August 2004 – 11:56
Total	1883986	

Example 6: List of top level 2 path under a directory /mydir

So this is how you need to setup your ExtraSection to add a chart for such a tracking:

```
ExtraSectionName1="List of top level 2 path under /mydir"
ExtraSectionCodeFilter1="200 304"
ExtraSectionCondition1="URL,^VmydirV.*"
ExtraSectionFirstColumnTitle1="Directory name"
ExtraSectionFirstColumnValues1="URL,^VmydirV([\\w]+)V"
ExtraSectionStatTypes1=PHK
MaxNbOfExtra1=50
MinHitExtra1=1
```

There is a lot of other possible use for Extra Sections ...

The following explanation is same than the one found in AWStats config file:

```
#-----
# EXTRA SECTIONS
#-----
# You can define your own charts, you choose here what are rows and columns
# keys. This feature is particularly usefull for marketing purpose, tracking
# products orders for example.
# For this, edit all parameters of Extra section. Each set of parameter is a
# different chart. For several charts, duplicate section changing the number.
# Note: Each Extra section reduces AWStats speed by 8%.
#
# WARNING: A wrong setup of Extra section might result in too large arrays
# that will consume all your memory, making AWStats unusable after several
# updates, so be sure to setup it correctly.
# In most cases, you don't need this feature.
#
# ExtraSectionNameX is title of your personalized chart.
# ExtraSectionCodeFilterX is list of codes the record code field must match.
# Put an empty string for no test on code.
# ExtraSectionConditionX are conditions you can use to count or not the hit,
# Use one of the field condition
# (URL,URLWITHQUERY,QUERY_STRING,REFERER,UA,HOSTINLOG,HOST,VHOST,extraX)
# and a regex to match, after a coma. Use "||" for "OR".
# ExtraSectionFirstColumnTitleX is the first column title of the chart.
# ExtraSectionFirstColumnValuesX is a string to tell AWStats which field to
# extract value from
# (URL,URLWITHQUERY,QUERY_STRING,REFERER,UA,HOSTINLOG,HOST,VHOST,extraX)
# and how to extract the value (using regex syntax). Each different value
# found will appear in first column of report on a different row. Be sure
# that list of different possible values will not grow indefinitely.
# ExtraSectionFirstColumnFormatX is the string used to write value.
# ExtraSectionStatTypesX are things you want to count. You can use standard
# code letters (P for pages,H for hits,B for bandwidth,L for last access).
# ExtraSectionAddAverageRowX add a row at bottom of chart with average values.
# ExtraSectionAddSumRowX add a row at bottom of chart with sum values.
# MaxNbOfExtraX is maximum number of rows shown in chart.
# MinHitExtraX is minimum number of hits required to be shown in chart.
```

Warning: the ExtraSectionConditionX MUST use regex values since AWStats 6.0.
ExtraSectionFirstColumnValuesX also need REGEX value for all AWStats versions.
Return to [examples](#) for examples on syntax use.



Other utilities

This is a list of other tools provided with AWStats.
All those tools are available in **tools** directory of AWStats distribution.

awstats_updateall.pl

awstats_updateall launches update process for all AWStats config files (except awstats.model.conf) found in a particular directory, so you can easily setup a cron/scheduler job. The scanned directory is by default /etc/awstats.

Usage: awstats_updateall.pl now [options]

Where options are:

- awstatsprog=pathtoawstatspl
- configdir=confdirtoscan

awstats_buildstaticpages.pl

awstats_buildstaticpages allows you to launch AWStats with -staticlinks option to build all possible pages allowed by AWStats -output option.

Usage:

awstats_buildstaticpages.pl (awstats_options) [awstatsbuildstaticpages_options]

where awstats_options are any option known by AWStats

- config=configvalue is value for -config parameter (REQUIRED)
- update option used to update statistics before to generate pages
- lang=LL to output a HTML report in language LL (en,de,es,fr,...)
- month=MM to output a HTML report for an old month=MM
- year=YYYY to output a HTML report for an old year=YYYY

and awstatsbuildstaticpages_options can be

- awstatsprog=pathtoawstatspl gives AWStats software (awstats.pl) path
- dir=outputdir to set output directory for generated pages
- builddate=%YY%MM%DD Used to add build date in built pages filenames
- staticlinksext=xxx For pages with .xxx extension instead of .html
- buildpdf[=pathtohtmldoc] Build a PDF file after building HTML pages.

Output directory must contains icon directory
when this option is used (need 'htmldoc').

New versions and FAQ at <http://www.awstats.org>

logresolvemerge.pl

logresolvemerge allows you to get one unique output log file, sorted on date, built from particular sources:

- It can read several input log files,
- It can read .gz/.bz2 log files,
- It can also makes a fast reverse DNS lookup to replace all IP addresses into host names in resulting log file.

logresolvemerge comes with ABSOLUTELY NO WARRANTY. It's a free software distributed with a GNU General Public License (See COPYING.txt file).

logresolvemerge is part of AWStats but can be used alone as a log merger or resolver before using any other log analyzer.

Usage:

```
logresolvemerge.pl [options] file
logresolvemerge.pl [options] file1 ... fileN
logresolvemerge.pl [options] *.*
perl logresolvemerge.pl [options] *.* > newfile
```

Options:

- dnslookup make a reverse DNS lookup on IP addresses
- dnslookup=n same with a n parallel threads instead of serial requests
- dnscache=file make DNS lookup from cache file first before network lookup
- showsteps print on stderr benchmark information every 8192 lines
- addfilenum if used with several files, file number can be added in first
- addfilename if used with several files, file name can be added in first field of output file. This can be used to add a cluster id when log files come from several load balanced computers.
- stoponfirsteof Stop processing when any logfile reaches end-of-file.
- printfields For IIS or W3C logs, prints the latest field header for the currentlog file when switching between log file entries so that the parsercan automatically determine which fields are available.
- ignoremissing will not fail if a log file is missing

This runs logresolvemerge in command line to open one or several server log files to merge them (sorted on date) and/or to make a reverse DNS lookup (if asked). The result log file is sent on standard output.

Note: logresolvemerge is not a 'sort' tool to sort one file. It's a software able to output sorted log records (with a reverse DNS lookup included or not) even if log records are dispatched in several files.

Each of those files must be already independently sorted itself (but that is the case in all web server log files). So you can use it for load balanced log files or to group several old log files.

Don't forget that the main goal of logresolvemerge is to send log records to a log analyzer in a sorted order without merging files on disk (NO NEED OF DISK SPACE AT ALL) and without loading files into memory (NO NEED OF MORE MEMORY). Choose of output records is done on the fly.

So logresolvemerge is particularly usefull when you want to output several and/or large log files in a fast process, with no use of disk or more memory, and in a chronological order through a pipe (to be used by a log analyzer).

Note: If input records are not 'exactly' sorted but 'nearly' sorted (this occurs with heavy servers), this is not a problem, the output will also be 'nearly' sorted but a few log analyzers (like AWStats) knows how to deal with such logs.

WARNING: If log files are old MAC text files (lines ended with CR char), you can't run this tool on Win or Unix platforms.

WARNING: Because of memory holes in ActiveState Perl version, use another Perl interpreter if you need to process large log files.

Now supports/detects:

- Automatic detection of log format
- Files can be .gz/.bz2 files if zcat/bzcat tools are available in PATH.
- Multithreaded reverse DNS lookup (several parallel requests) with Perl 5.8+.

New versions and FAQ at <http://www.awstats.org>

maillogconvert.pl

maillogconvert is mail log preprocessor that convert a mail log file (from postfix, sendmail or qmail servers) into a human readable format. The output format is also ready to be used by a log analyzer, like AWStats.

Usage:

```
perl maillogconvert.pl [standard|vadmin] [year] < logfile > output
```

The first parameter specifies what format the mail log file is :

standard – logfile is standard postfix,sendmail,qmail or mdaemon log format

vadmin – logfile is qmail log format with vadmin multi-host support

The second parameter specifies what year to timestamp logfile with, if current year is not the correct one (ie. 2002). Always use 4 digits. If not specified, current year is used.

If no output is specified, it goes to the console (stdout).

For example, the following sample from postfix mail log server:

```
# 1 Mail from user@aol.com -> touser@toserver.com, forward touser@toserver.com -> touser@mainserver.com
Jan 01 07:27:31 apollon postfix/smtpd[1684]: connect from remt30.cluster1.charter.net[209.225.8.40]
Jan 01 07:27:32 apollon postfix/smtpd[1684]: 2BC793B8A4: client=remt30.cluster1.charter.net[209.225.8.40]
Jan 01 07:27:32 apollon postfix/cleanup[1687]: 2BC793B8A4: message-id=<36027278@vneka>
Jan 01 07:27:32 apollon postfix/qmgr[13860]: 2BC793B8A4: from=, size=2130, nrcpt=1 (queue active)
Jan 01 07:27:32 apollon postfix/smtpd[1684]: disconnect from remt30.cluster1.charter.net[209.225.8.40]
Jan 01 07:27:38 apollon postfix/local[1689]: 2BC793B8A4: to=, orig_to=, relay=local, delay=6, status=sent
("/usr/bin/procmail")
# 2 Reject: 450
Jan 01 14:05:44 apollon postfix/smtpd[2114]: connect from baby.mainframe.nl[81.29.4.2]
Jan 01 14:05:44 apollon postfix/smtpd[2114]: E0C9D3BD9A: client=baby.mainframe.nl[81.29.4.2]
Jan 01 14:05:44 apollon postfix/smtpd[2114]: E0C9D3BD9A: reject: RCPT from baby.mainframe.nl[81.29.4.2]: 450 : User
unknown in local recipient table; from=<> to= proto=ESMTP helo=
Jan 01 14:10:16 juni postfix/smtpd[2568]: C34ED1432B: reject: RCPT from relay2.tp2rc.edu.tw[163.28.32.177]: 450 : User
unknown in local recipient table; from=<> proto=ESMTP helo=
# 1 From unknown
Jan 01 15:17:05 apollon postfix/smtpd[29866]: connect from tom12.bellnexxia.net[209.226.175.56]
Jan 01 15:17:05 apollon postfix/smtpd[29866]: 578093B8B5: client=tom12.bellnexxia.net[209.226.175.56]
Jan 01 15:17:05 apollon postfix/cleanup[28931]: 578093B8B5:
message-id=<20030905131913.EOVH11393.tom12-srv.bellnexxia.net@tom12-srv>
Jan 01 15:17:06 apollon postfix/qmgr[965]: 578093B8B5: from=<>, size=109367, nrcpt=1 (queue active)
Jan 01 15:17:06 apollon postfix/local[32432]: 578093B8B5: to=, orig_to=, relay=local, delay=1, status=sent
("/usr/bin/procmail")
Jan 01 15:17:06 apollon postfix/smtpd[29866]: disconnect from tom12.bellnexxia.net[209.226.175.56]
```

will give a file that looks like this:

```
2004-01-01 07:27:38 fromuser@aol.com touser@toserver.com remt30.cluster1.charter.net localhost SMTP - 1 2130
2004-01-01 14:05:44 <> touser2@toserver.com baby.mainframe.nl - SMTP - 450 0
2004-01-01 14:10:16 <> unknownuser@unknownserver.com relay2.tp2rc.edu.tw - SMTP - 450 0
2004-01-01 15:17:06 <> touser@toserver.com tom12.bellnexxia.net localhost SMTP - 1 109367
```

See [FAQ-COM100](#) to see how to use maillogconvert.pl with AWStats to analyze mail log files.

urlaliasbuilder.pl

Urlaliasbuilder generates an 'urlalias' file from an input file (an urlalias file is a file with two columns: url and clear title of url).

The input file must contain a list of URLs (It can be an AWStats history file).

For each of those URLs, the script get the corresponding HTML page and catch the header information (title), then it writes an output file that contains one line for each URLs and several fields:

- The first field is the URL,
- The second is title caught from web page.

This resulting file can be used by AWStats urlalias plugin.

Usage: urlaliasbuilder.pl -site=www.myserver.com [options]

The site parameter contains the web server to get the page from.

Where options are:

-urllistfile=Input urllist file

If this file is an AWStats history file then urlaliasbuilder will use the SIDER section of this file as its input URL's list.

-urlaliasfile=Output urlalias file to build

-overwrite Overwrite output file if exists (by default appends to file).

-secure Use https protocol

Example: urlaliasbuilder.pl -site=www.someotherhost.com

New versions and FAQ at <http://www.awstats.org>

This script was written from Simon Waight original works title-grabber.pl.



Glossary

Unique Visitor:

A unique visitor is a person or computer (host) that has made at least 1 hit on 1 page of your web site during the current period shown by the report. If this user makes several visits during this period, it is counted only once. Visitors are tracked by IP address, so if multiple users are accessing your site from the same IP (such as a home or office network), they will be counted as a single unique visitor.

The period shown by AWStats reports is by default the current month.

However if you use AWStats as a CGI you can click on the "year" link to have a report for all the year. In such a report, period is a full year, so Unique Visitors are number of hosts that have made at least 1 hit on 1 page of your web site during the year.

Visits:

Number of visits made by all visitors.

Think "session" here, say a unique IP accesses a page, and then requests three other pages within an hour. All of the "pages" are included in the visit, therefore you should expect multiple pages per visit and multiple visits per unique visitor (assuming that some of the unique IPs are logged with more than an hour between requests)

Pages:

The number of "pages" viewed by visitors. Pages are usually HTML, PHP or ASP files, not images or other files requested as a result of loading a "Page" (like js,css... files). Files listed in the NotPageList config parameter (and match an entry of OnlyFiles config parameter if used) are not counted as "Pages".

Hits:

Any files requested from the server (including files that are "Pages") except those that match the SkipFiles config parameter.

Bandwidth:

Total number of bytes for pages, images and files downloaded by web browsing.

Note 1: Of course, this number includes only traffic for web only (or mail only, or ftp only depending on value of LogType).

Note 2: This number does not include technical header data size used inside the HTTP or HTTPS protocol or by protocols at a lower level (TCP, IP...).

Because of two previous notes, this number is often lower than bandwidth reported by your provider (your provider counts in most cases bandwidth at a lower level and includes all IP and UDP traffic).

Entry Page:

First page viewed by a visitor during its visit.

Note: When a visit started at end of month to end at beginning of next month, you might have an Entry page for the month report and no Exit pages.

That's why Entry pages can be different than Exit pages.

Exit Page:

Last page viewed by a visitor during its visit.

Note: When a visit started at end of month to end at beginning of next month, you might have an Entry page for the month report and no Exit pages.

That's why Entry pages can be different than Exit pages.

Session Duration:

The time a visitor spent on your site for each visit.

Some Visits durations are 'unknown' because they can't always be calculated. This is the major reason for this:

- Visit was not finished when 'update' occurred.
 - Visit started the last hour (after 23:00) of the last day of a month (A technical reason prevents AWStats from calculating duration of such sessions).
-

Grabber:

A browser that is used primarily for copying locally an entire site. These include for example "teleport", "webcapture", "webcopier"...

Direct access / Bookmark:

This number represent the number of hits or ratio of hits when a visit to your site comes from a direct access. This means the first page of your web site was called:

- By typing your URL on the web browser address bar
- By clicking on your URL stored by a visitor inside its favorites
- By clicking on your URL found everywhere but not another internet web pages (a link in a document, an application, etc...)
- Clicking an URL of your site inside a mail is often counted here.

Add To Favourites:

This value, available in the "miscellaneous chart", reports an estimated indicator that can be used to have an idea of the number of times a visitor has added your web site into its favourite bookmarks.

The technical rules for that is the following formula:

$Number\ of\ Add\ to\ Favourites = round((x+y) / r)$

where

x = Number of hits made by IE browsers for "/anydir/favicon.ico", with a referer field not defined, and with no 404 error code

y = Number of hits made by IE browsers for "/favicon.ico", with a referer field not defined, with or without 404 error code

r = Ratio of hits made by IE browsers compared to hits made by all browsers ($r \leq 1$)

As you can see in formula, only IE is used to count reliable "add", the "Add to favourites" for other browsers are estimated using ratio of other browsers usage compared to ratio of IE usage. The reason is that only IE do a hit on favicon.ico nearly ONLY when a user add the page to its favourites. The other browsers make often hits on this file also for other reasons so we can't count one "hit" as one "add" since it might be a hit for another reason.

AWStats differentiate also hits with error and not to avoid counting multiple hits made recursively in upper path when favicon.ico file is not found in deeper directory of path.

Note that this number is just an indicator that is in most case higher than true value. The reason is that even IE browser sometimes make hit on favicon without an "Add to favourites" action by a user.

HTTP Status Codes:

HTTP status codes are returned by web servers to indicate the status of a request. Codes **200** and **304** are used to tell the browser the page can be viewed. 206 codes indicate partial downloading of content and is reported in the Downloads section. All other codes generates hits and traffic 'not seen' by the visitor. For example a return code 301 or 302 will tell the browser to ask another page. The browser will do another hit and should finally receive the page with a return code **200** and **304**. All codes that are 'unseen' traffic are isolated by AWStats in the HTTP Status report chart, enabled by the directives [ShowHTTPErrorsStats](#), in config file. You can also change value for 'not error' hits (set by default to **200** and **304** with the [ValidHTTPcodes](#) directive. The following table outlines all status codes defined for the HTTP/1.1 draft specification outlined in [IETF rfc 2068](#).

They are 3-digit codes where the first digit of this code identifies the class of the status code and the remaining 2 digits correspond to the specific condition within the response class. They are classified in 5 categories:

- [1xx](#) – informational
- [2xx](#) – successful
- [3xx](#) – redirection
- [4xx](#) – client error
- [5xx](#) – server error

1xx class – Informational

Informational status codes are provisional responses from the web server... they give the client a heads-up on what the server is doing. Informational codes do not indicate an error condition.

100	100 Continue The continue status code tells the browser to continue sending a request to the server.
101	101 Switching Protocols The server sends this response when the client asks to switch from HTTP/1.0 to HTTP/1.1

2xx class – Successful

This class of status code indicates that the client's request was received, understood, and successful.

200	200 Successful
201	201 Created
202	202 Accepted
203	203 Non-Authorative Information
204	204 No Content
205	205 Reset Content
206	206 Partial Content The partial content success code is issued when the server fulfills a partial GET request. This happens when the client

is downloading a multi-part document or part of a larger file.

3xx class – Redirection

This code tells the client that the browser should be redirected to another URL in order to complete the request. This is not an error condition.

300	300 Multiple Choices
301	301 Moved Permanently
302	302 Moved Temporarily
303	303 See Other
304	304 Not Modified
305	305 Use Proxy

4xx class – Client Error

This status code indicates that the client has sent bad data or a malformed request to the server. Client errors are generally issued by the webserver when a client tries to gain access to a protected area using a bad username and password.

400	400 Bad Request
401	401 Unauthorized
402	402 Payment Required
403	403 Forbidden
404	404 Not Found
405	405 Method Not Allowed
406	406 Not Acceptable
407	407 Proxy Authentication Required
408	408 Request Timeout
409	409 Conflict
410	410 Gone
411	411 Length Required
412	412 Precondition Failed
413	413 Request Entity Too Long
414	414 Request-URI Too Long
415	415 Unsupported Media Type

5xx class – Server Error

This status code indicates that the client's request couldn't be successfully processed due to some internal error in the web server. These error codes may indicate something is seriously wrong with the web server.

500	500 Internal Server Error An internal server error has caused the server to abort your request. This is an error condition that may also indicate a misconfiguration with the web server. However, the most common reason for 500 server errors is when you try to execute a script that has syntax errors.
501	501 Not Implemented This code is generated by a webserver when the client requests a service that is not implemented on the server. Typically, not implemented codes are returned when a client attempts to POST data to a non-CGI (ie, the form action tag refers to a non-executable file).
502	502 Bad Gateway The server, when acting as a proxy, issues this response when it receives a bad response from an upstream or support server.
503	503 Service Unavailable The web server is too busy processing current requests to listen to a new client. This error represents a serious problem with the webserver (normally solved with a reboot).
504	504 Gateway Timeout Gateway timeouts are normally issued by proxy servers when an upstream or support server doesn't respond to a request in a timely fashion.
505	505 HTTP Version Not Supported The server issues this status code when a client tries to talk using an HTTP protocol that the server doesn't support or is configured to ignore.

SMTP Status Codes:

SMTP status codes are returned by mail servers to indicate the status of a sending/receiving mail. The status code depends on mail server and preprocessor used to analyze log file.

All codes that are failure codes are isolated by AWStats in the SMTP Status report chart, enabled by the directives [ShowSMTPErrorsStats](#) in AWStats config file. You can decide which codes are successful mail transfer that should not appear in this chart with the [ValidSMTPCodes](#) directive.

Here are values reported for most mail servers (This should also be values when mail log file is preprocessing with maillogconvert.pl).

SMTP Errors are classified in 3 categories:

- [2xx/3xx](#) – successful
- [4xx](#) – failure, asking sender to try later
- [5xx](#) – permanent failure

2xx/3xx class – Success

They are SMTP protocols successful answers

200	200 Non standard success response Non standard success response
211	211 System status, or system help reply System status, or system help reply
214	214 Help message Help message
220	220 Service ready Service ready
221	221 Service closing transmission channel Service closing transmission channel
250	250 Requested mail action taken and completed Your ISP mail server have successfully executes a command and the DNS is reporting a positive delivery.
251	251 User not local: will forward to Your message to a specified email address is not local to the mail server, but it will accept and forward the message to a different recipient email address.
252	252 Recipient cannot be verified Recipient cannot be verified but mail server accepts the message and attempts delivery
354	354 Start mail input and end with . Indicates mail server is ready to accept the message or instruct your mail client to send the message body after the mail server have received the message headers.

4xx class – Temporary Errors

Those codes are temporary error message. They are used to tell client sender that an error occurred but he can try to solve it but trying again, so in most cases, clients that receive such codes will keep the mail in their queue and will try again later.

421	421 Service not available, closing transmission channel This may be a reply to any command if the service knows it must shut down.
450	450 Requested mail action not taken: mailbox busy or access denied Your ISP mail server indicates that an email address does not exist or the mailbox is busy. It could be the network connection went down while sending, or it could also happen if the remote mail server does not want to accept mail from you for some reason i.e. (IP address, From address, Recipient, etc.)
451	451 Requested mail action aborted: error in processing Your ISP mail server indicates that the mailing has been interrupted, usually due to overloading from too many messages or transient failure is one in which the message sent is valid, but some temporary event prevents the successful sending of the message. Sending in the future may be successful.
452	452 Requested mail action not taken: insufficient system storage Your ISP mail server indicates, probable overloading from too many messages and sending in the future may be successful.
453	453 Too many messages Some mail servers have the option to reduce the number of concurrent connection and also the number of messages sent per connection. If you have a lot of messages queued up it could go over the max number of messages per connection. To see if this is the case you can try submitting only a few messages to that domain at a time and then keep increasing the number until you find the maximum number accepted by the server.

5xx class – Permanent Errors

This are permanent error codes. Mail transfer is definitely a failure. No other try will be done.

500	500 Syntax error, command unrecognized or command line too long
501	501 Syntax error in parameters or arguments
502	502 Command not implemented
503	503 Server encountered bad sequence of commands

504	504 Command parameter not implemented
521	521 does not accept mail or closing transmission channel You must be pop-authenticated before you can use this SMTP server and you must use your mail address for the Sender/From field.
530	530 Access denied A sendmailism ?
550	550 Requested mail action not taken (Relaying not allowed, Unknown recipient user, ...) Sending an email to recipients outside of your domain are not allowed or your mail server does not know that you have access to use it for relaying messages and authentication is required. Or to prevent the sending of SPAM some mail servers will not allow (relay) send mail to any e-mail using another company's network and computer resources.
551	551 User not local: please try or Invalid Address: Relay request denied
552	552 Requested mail action aborted: exceeded storage allocation ISP mail server indicates, probable overloading from too many messages.
553	553 Requested mail action not taken: mailbox name not allowed Some mail servers have the option to reduce the number of concurrent connection and also the number of messages sent per connection. If you have a lot of messages queued up (being sent) for a domain, it could go over the maximum number of messages per connection and/or some change to the message and/or destination must be made for successful delivery.
554	554 Requested mail action rejected: access denied
557	557 Too many duplicate messages Resource temporarily unavailable Indicates (probable) that there is some kind of anti-spam system on the mail server.



Log analyzers Comparisons

Comparison between AWStats and other famous statistics tools

Features/Softwares	AWStats	Analog	Webalizer	Sawmill Analytics
Version – Date	6.95 – October 2009	6.0 – December 2004	2.01–10 – April 2002	8.5 – July 2011
Language	Perl	C	C	C/Salang
Available on all platforms	Yes	Yes	Yes	Yes
Readable sources available	Yes	Yes	Yes	No (obfuscated sources for compilation only)
Price/Licence	Free/GPL	Free/GPL	Free/GPL	From \$99 Per Profile Lite/Pro/Ent
Works with Apache combined (XLF/ELF)	Yes	Yes	Yes	Yes
Works with Apache common (CLF) log format	All features available with log format (b)	All features available with log format (b)	All features available with log format (b)	All features available with log format (b)
Works with IIS (W3C) log format	Yes	Yes	Need a patch	Yes
Works with personalized log format	Yes	Yes	No	Yes
Analyze Web/Ftp/Mail log files	Yes/Yes/Yes	Yes/No/No	Yes/No/No	Yes/Yes/Yes (850+)
Report and update of statistics from	Command line (CLI) and/or a browser (CGI)	Command line (CLI) and/or a browser (CGI)	Command line	Command line (CLI) and/or a browser (CGI) and/or HTTP API
Scheduler	External (crontab, windows task manager)	External (crontab, windows task manager)	External (crontab, windows task manager)	Built-in
Internal reverse DNS lookup	Yes	Yes	Yes	Yes
DNS cache file	Static and dynamic	Static or dynamic	Static or dynamic	Yes (per update, or custom)
Process logs spitted by load balancing systems	Yes	Yes	No	Yes

Report number of "human" visits	Yes	No	Yes	Yes (Sessions)
Report unique "human" visitors	Yes	No	No	Yes (Visitors)
Report session duration	Yes	No	No	Yes
Not ordered records tolerance and reorder for visits	Yes	Visits not supported	No	Yes
Statistics for visits are based on	Pages *****	Not supported	Pages *****	Pages *****
Statistics for unique visitors are based on	Pages *****	Not supported	Not supported	Client IP / Cookie Custom *****
Report countries	From IP location or domain name	Domain name	Domain name	From IP location or domain name
Report regions (US and Canada states)	Need Maxmind Regions database	No	No	Yes GeoLite City included
Report cities and major countries regions	Need Maxmind Cities database	No	No	Yes GeoLite City included
Report ISP	Need Maxmind ISP database	No	No	Need Maxmind ISP database
Report Organizations name	Need Maxmind Org database	No	No	Need Maxmind Org database
Report hosts	Yes	Yes	Yes	Yes
Report WhoIs informations on hosts	Yes	No	No	No
Report authenticated users	Yes	Yes	No	Yes
Report/Filter robots (nb detected)	Yes/Yes (642**)	Yes / Yes (8**)	No/No	Yes/Yes (250**)
Report/Filter worms (nb of families detected)	Yes/Yes (5)	No / No	No/No	Yes/Yes (4)
Report rush hours	Yes	Yes	Yes	Yes
Report days of week	Yes	Yes	Yes	Yes
Report most often viewed pages	Yes	Yes	Yes	Yes
Report entry pages	Yes	No	Yes	Yes
Report exit pages	Yes	No	Yes	Yes
Not ordered records tolerance and reorder for entry/exit pages	Yes	Entry/Exit not supported	No	Yes
Detection of CGI pages as pages (and not just hits)	Yes	Only if prog ends by a defined value	Only if prog ends by a defined value	Yes
Report pages by directory	No	Yes	No	Yes
Report pages with last access time/average size	Yes/Yes	Yes/No	No/No	Yes/No
	Yes/Yes/Yes	No/No/No	No/No/No	Yes/Yes/Yes

Dynamic filter on hosts/pages/referers report				
Report web compression statistics (mod_gzip,mod_deflate)	Yes	No	No	No
Report file types	Yes	Yes	No	Yes
Report by file size	No	Yes	No	Yes
Report OS (nb detected)	Yes (71)	Yes (29)	No (0)	Yes
Report browsers (nb detected)	Yes (208*)	Yes (9*)	Yes (4*)	Yes (~20*)
Report details of browsers versions	Major and minor versions	Major versions by default, minor with SUBBROW option	Major an minor versions	Major and minor versions
Report screen sizes	Yes	No	No	Yes & Depths
Report tech supported by browser for Java/Flash/PDF	Yes/Yes/Yes	No/No/No	No/No/No	No/No/No
Report audio format supported by browser for Real/QuickTime/Mediaplayer	Yes/Yes/Yes	No/No/No	No/No/No	No/No/No
Report search engines used (nb detected)	Yes (228***)	Yes (24)	No (0)	Yes (67***)
Report keywords/keyphrases used on search engines (nb detected)	Yes/Yes (118***)	Yes/No (29***)	No/Yes (14***)	Yes/Yes (67***)
Report external referring web page with/without query	Yes/Yes	No/No	No/Yes	Yes/Yes
Report HTTP Errors	Yes	Yes	Yes	Yes
Report 404 Errors	Nb + List referer	Nb only	Nb only	Nb + List last date/referer
Other personalized reports for miscellaneous/marketing purpose	Yes	No	No	Yes
Daily statistics	Yes	Yes	Yes	Yes
Weekly statistics	No	No	No	Yes
Monthly statistics	Yes	Yes	Yes	Yes
Yearly statistics	Yes	Yes	Yes	Yes
Custom date range	No	No	No	Yes
Benchmark with no DNS lookup in lines/seconds (full features enabled, with XLF format, cygwin Perl 5.8, Athlon 1Ghz)	5200****	39000****	12000****	Not calculated
Benchmark with DNS lookup in lines/seconds (full features enabled, with XLF format, cygwin Perl 5.8, Athlon 1Ghz)	80****	80****	80****	Not calculated
Analyzed data save format (to use with third tools)	Structured text file or XML	Text files with OUTPUT option	Flat text file	Flat text file/MySQL/MS SQL/Oracle
Export statistics to PDF	Experimental	No	No	

Graphical statistics in one page /
several / or frames

Yes/Yes/Yes

Yes/No/No

Yes/Yes/No

Yes/Yes/Yes

* This number is not really the number of browsers detected. All browsers (known and unknown) can be detected by products that support user agent listing (AWStats, Analog, Webalizer, Sawmill). The 'browser detection feature' and number is the number of known browsers for which different versions/ids of same browser are grouped by default in one browser name.

** AWStats can detect robots visits: All robots among the most common are detected, list is in [robotslist.txt](#) (250Kb). Products that are not able to do this give you false information, above all if your site has few visitors. For example, if you're site was submitted to all famous search engines, robots can make 500 visits a month, to find updates or to see if your site is still online. So, if you have only 2000 visits a month, products with no robot detection capabilities will report 2500 visits (A 25% error !). AWStats will report 500 visits from robots and 2000 visits from human visitors. Sawmill Analytics uses a "currently active" list of robots based on the [robotstxt.org](#) database.

*** AWStats has url syntax rules for the most popular search engines (that's the 'number detected'). Those rules are updated with AWStats updates. But AWStats has also an algorithm to detect keywords of unknown search engines with unknown url syntax rules. Sawmill uses unique syntax to detect 67 search engines, and you can add any number of custom SE's.

**** Most log analyzers have poor (or not at all) robots, search engines, os or browsers detection capabilities and less features (no or poor visits count, no filter rules, etc...).

It is not possible to add all AWStats features to other log analyzers, so don't forget that benchmarks results are for 'different features'. For this benchmark, I did just complete Webalizer and Analog robots or search engines databases with part of AWStats database. So Webalizer config file was completed with this [file](#), Analog config file was completed with this [file](#). Note that without this very light add (using default conf file), Webalizer speed is 3 times faster, Analog is 15% faster). Benchmark was made on a combined (XLF/CLF) log record on an Athlon 1GHz.

You must keep in mind that all this times are without reverse DNS lookup. DNS lookup speed depends on your system, network and Internet but not on the log analyzer you use. For this reason, DNS lookup is disabled in all log analyzer benchmarks. Don't forget that DNS lookup is 95% (even with a lookup cache) of the time used by a log analyzer, so if your host is not already resolved in log file and DNS lookup is enable, the total time of the process will be nearly the same whatever is the speed of the log analyzer.

***** Some visitors use a lot of proxy servers to surf (ie: AOL users), this means it's possible that several hosts (with several IP addresses) are used to reach your site for only one visitor (ie: one proxy server download the page and 2 other servers download all images). Because of this, if stats of unique visitors are made on "Hits", 3 users are reported but it's wrong. So AWStats, considers only HTML pages to count unique visitors. This decrease the error (not totally, because it's always possible that a proxy server download one HTML frame and another one download another frame). Sawmill Analytics allows you to choose what you define as a visitor – by default the client IP is used, but you can use a cookie (persistant or session) or any custom string, or combination of string from teh log data.

(a) Data were provided by Sawmill company (Graham Smith).

(b) With such log format, there is no user agent information in log file, so some reports are broken. For example, it's not possible to make reports on browser or os for (information is not stored in log file). To solve this, use another log format (like the combined format).



Benchmarks

AWStats update process must be ran frequently, so it's important to know what is AWStats speed to choose an optimum delay between each update process according to AWStats speed and the refresh rate you need to have. AWStats speed depends on AWStats version and options/setup you use in configuration file.

This is benchmark results with AWStats version 6.0 and a common configuration:

HARDWARE: Athlon 1 GHz / 256MB

SOFTWARE: Windows 2000 / Perl 5.8 (Cygwin Perl)

CONFIG OPTIONS: Default values were used: [LogFormat=1](#), [DNSLookup=0](#), [URLWithQuery=0](#), [URLReferrerWithQuery=0](#), [URLWithAnchor=0](#), No plugins

AVERAGE SPEED: **5200** lines by seconds

Other times for different kind of web sites sizes are shown later in this page...

This is other important information to know:

- A log file size is about **150** (NCSA common/CLF log files) to **320 times** (NCSA extended/XLF/ELF log files) its number of lines,
 - **1,000 visits = 8,000 pages** (with 8 pages/visits) = **64,000 lines** (with 8 hits/page) = **20 MB file => 15 seconds** (Athlon 1GHz, Standard Perl 5.8)
- History files (AWStats database, resuming the log analysis) has the following size (one file a month) : **15000+90*x+100*y bytes** (where x is number of unique visitors a month and y is number of different pages on web sites). If you use option [BuildHistoryFormat=xml](#), you must multiplie this value by 3.

WARNING ! All those data are average values for a common public site with default configuration. Calculation rule can be seriously changed according to web server or AWStats configuration and web site content.

Don't forget that benchmarks of log analyzers are made without reverse DNS lookup because DNS lookup is so slow (depending on Internet network and your system), that if enabled in AWStats configuration file, it would take more than **99%** of the time of a log analysis ! Take a look at the following chart to:

- Get more real ideas on benchmarks results
- Get more information and advice on a good setup for your site.

This is examples of frequency/parameters you should use to have a good use of AWStats:

Your Web site traffic	Perl distrib	Values for parameters			Recommended update frequency (Rotate log delay)	Memory required**	Update process duration***
		DNSLookup*	URLWithQuery	URLReferrerWithQuery			
0 – 1,000 visits/month	Your choice	0 (or 2)	0 or 1	0 or 1	Once a day Log files are 0–1 MB 2000 lines to process	4 MB free	1s
		1	0 or 1	0 or 1	Once a day Log files are 0–1 MB 2000 lines to process	4 MB free	2mn
1,000 – 10,000 visits/month	Your choice	0 (or 2)	0 or 1	0 or 1	Once a day Log files are 1–10 MB 2000–20000 lines	4–8 MB free	1s–10s

		1	0 or 1	0 or 1	Once a day Log files are 1–10 MB 2000–20000 lines	4–8 MB free	2mn–10mn
10,000 – 100,000 visits/month	Your choice	0 (or 2)	0 or 1	0 or 1	One a day Log Files are 10–100 MB 20000–200000 lines	8–32 MB free	10s–120s
		1	0 or 1	0 or 1	Once a day Log files are 10–100 MB 20000–200000 lines	8–32 MB free	10mn–50mn
100,000 – 500,000 visits/month	Your choice	0 (or 2)	0	0 or 1	Every 6 hours Log Files are 24–96 MB 50000–300000 lines	16–92 MB free	30s–3mn
		1	0	0 or 1	Every 6 hours Log Files are 24–96 MB 50000–300000 lines	16–92 MB free	15mn–60mn
500,000 – 2,000,000 visits/month	See next section on pb with ActiveState	0 (or 2)	0	0	Every 6 hours Log Files are 96–384 MB 300000–1200000 lines	64–256 MB free	3mn–12mn
2,000,000 – 4,000,000 visits/month	See next section on pb with ActiveState	0 (or 2)	0	0	Every 6 hours Log Files are 384–768 MB 1200000–2400000 lines	256–512 MB free	12mn–24mn
+4,000,000 visits/month	See next section on pb with ActiveState	AWStats is a good choice for such web sites only if you use a dedicated server with a large amount of memory. Try a tool with less features but faster like "row counter log analyzers" like Analog if not.					

* You should set [DNSLookup](#) parameter to 0 (or 2) if
– reverse DNS lookup is already done in your log file,
– or if your web site has more than 250,000 visits a month.

Note: Country report can works without reverse DNS lookup if plugin 'geoip' is enabled (faster and more accurate than reverse DNS lookup).

** This is free memory required for update process (in MB), this is not hardware memory installed !

Warning: If you use the [URLWithQuery](#) or [URLReferrerWithQuery](#) option, or forget to complete correctly [URLQuerySeparators](#) for some sites, this value can be dramatically increased.

*** Duration with DNSLookup set to 1 is very long because of DNS lookup whatever is speed of your computer. Duration with DNSLookup set to 0 (or 2) is with Athlon 1GHz/256MB, Cygwin Perl 5.8 and LogFormat=1.

SOME IMPORTANT ADVICES FOR A GOOD USE OF AWSTATS:

★★★★★ – Check that [DNSLookup](#) is disabled in AWStats (DNSLookup should not be set to 1). If you need a 'Country' report, you should prefer **using the 'geoip' plugin using [Maxmind database](#) instead of a DNS lookup**. Those plugins allow you to have more accurate results, faster with no network queries (With DNSLookup enabled, log analyze speed is decreased by 40 to 100 times, so use it only if required). Note that without DNS lookup and without the 'geoip' plugins, 'Country' report might work but results will be less accurate than the 'geoip' usage and it works only if hosts addresses in your log file are already resolved (need to setup your web server to do so, your web server will be slowed).

★★★★★ – Use carefully parameters [URLWithQuery](#), [URLReferrerWithQuery](#) and [URLWithAnchor](#) (Let them set to 0 if you don't know what they means) and check your web site URLs' syntax to know if you don't need to complete the parameter [URLQuerySeparators](#). If you really need to use [URLWithQuery](#)=1, check that [URLWithQueryWithOnlyFollowingParameters](#) or [URLWithQueryWithoutFollowingParameters](#) are set properly.

★★★★★ – Use **last Perl version** (For example Perl 5.8 is 5% faster than 5.6) and, more important, **for large log files, use standard Perl distribution instead of ActiveState**. This is because ActiveState 5.006 (and may be also other

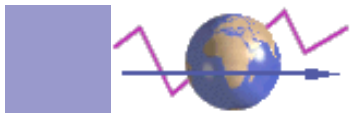
versions), has very important memory hole problem making speed of analysis slower and slower reaching 0 lines/seconds and using all your memory. You can see the decrease by adding the `--showsteps` option on command line. The speed should be constant to value given in top of this page, even for several Gigabytes log files !

☆☆☆ – **Rotate your log** (See [FAQ-SET500](#)) and launch AWStats more often (from crontab or a scheduler, See [FAQ-SET550](#)). The more often you launch AWStats, the less AWStats has new lines in log to process. This can also solve the ActiveState memory problem (see next advice).

☆☆☆ – Be sure that your [HostAliases](#) parameter list is complete.

☆☆☆ – Use **last AWStats version** (For example AWStats 6.0 is 15% faster than 5.9).

☆☆☆ – For geeks users, you can also recompile your Perl with different options. For example, it seems that the `"use64bitint=define usemymalloc=y"` can increase speed by 10%.



Little Tips about Security

A lot of AWStats users have several web site to manage. This is particularly true for web hosting providers. The most common things you would like to do is to prevent user xxx (having a site www.xxx.com) to see statistics of user yyy (having a site www.yyy.com).

This is example of possible way of working:

1) HIGHLY SECURED POLICY

Policy:

You have several different config/domains owned by different users and you want to build statistics for each of them. You don't need that your customer have "real-time" statistics.

This is a very good choice for web hosting providers with few but very large web sites of important customers.

Advantage:

Very highly secured.

Disadvantage:

Statistics are static, no dynamic update/view.

How:

All statistics pages for a config/domain file are built in static html files using **-output -staticlinks** option.

There is no CGI use of AWStats and static built pages are stored in a web protected **realm** to be securely viewed by correct allowed users only (or sent by mails).

If users have a command line access (telnet) on statistics server, you must set correct permissions on AWStats database files. Set all AWStats database files (built by the update process) for config/domain1 to have read/write for *user1* (or an admin user) and NO read and NO write permissions for any other users.

Then, check that the [SaveDatabaseFilesWithPermissionsForEveryone](#) parameter is set 0 in your config/domain files.

If AWStats database files/directory for config/domain1 are read protected, only allowed users can see statistics for config/domain1.

If AWStats database files/directory for config/domain1 are write protected, only allowed users can update statistics for config/domain1.

2) MEDIUM SECURED POLICY

Policy:

You have several config/domain and several users. You want to specify which user can see or update dynamically statistics for each config/domain.

This is one of the most popular way of working.

Advantage:

Statistics are dynamic. High level of manageability.

Disadvantage:

AWStats database files must still be readable by anonymous web server user, so if an experienced user can have an access to the server (telnet) where AWStats database files are stored, he can succeed in installing and running a "hacked" version of AWStats that ignores value of parameter `AllowAccessFromWebToAuthenticatedUsersOnly`.

How:

awstats.pl must be saved in a web protected **realm** to force a visitor to enter its username/password to access AWStats CGI program.

Example of directives you can add into Apache to have awstats.pl in a web protected realm:

```
<Files "awstats.pl">
```

```
AuthUserFile /path/to/.passwd
```

```
AuthGroupFile /path/to/.group
AuthName "Restricted Area For Customers"
AuthType Basic
require valid-user
</Files>
```

If you add such directives into a .htaccess file, you must also check that the *AllowOverride* directive is set to *All* in Apache config file to allow the use of .htaccess files.

To know how to create a protected realm for servers other than Apache, see your web server manual.

Then edit each config/domain file you want to be protected to set [AllowAccessFromWebToAuthenticatedUsersOnly](#) to 1.

You can also edit list of authorized users in the [AllowAccessFromWebToFollowingAuthenticatedUsers](#) parameter.

You can also specify a range of allowed browsers IP Addresses with the [AllowAccessFromWebToFollowingIPAddresses](#) parameter.

You can also set [SaveDatabaseFilesWithPermissionsForEveryone](#) parameter to 0 in all config/domain files, except if you want to allow update from web with option [AllowToUpdateStatsFromBrowser](#)=1. But this is not recommended as you need to give read/write permission for Web server user on all history files (Except if you setuid AWStats script for each authorized user, but this make setup much harder).

The following parameters [ErrorMessage](#)s and [DebugMessages](#) are also parameters related to security.

Other tip: If the **AWSTATS_FORCE_CONFIG** environment variable is defined, AWStats will always use the config file *awstats.VALUE_OF_AWSTATS_FORCE_CONFIG.conf* as the config/domain file. So if you add this environment variable into your web server environment, for example by adding the line

SetEnv AWSTATS_FORCE_CONFIG configvalueforthisdomain

in your Apache *<VirtualHost>* directive group in httpd.conf (with other directives), AWStats will use the config file called *awstats.configvalueforthisdomain.conf* to choose which statistics used, even if a visitor try to force the config/domain file with the URL '*http://mydomain/cgi-bin/awstats.pl?config=otherdomain*'. This might be usefull for those who edit their config/domain file with [AllowAccessFromWebToFollowingAuthenticatedUsers](#)="__REMOTE_USER__" instead of maintaining the list of authorized users into each AWStats config file.

3) NO SECURITY POLICY

Policy:

You have only one hosts or several hosts or users but you don't need to manage particular permissions for your different config/domain statistics.

Advantage:

Setup is very easy (No need of particular setup). Statistics are dynamic.

Disadvantage:

No way to prevent stats for config/domain to be seen by a user that known the config/domain name and the url syntax to see stats of a particular config/domain.

How:

No particular things to do (You can however easily use [AllowAccessFromWebToFollowingIPAddresses](#) parameter to have a minimum of security).

There is a lot of possible use for AWStats combining all its options/parameters with all web servers options/parameters and operating systems security features. Just use the one you need...



Frequently Asked Questions + Troubleshooting

ABOUT QUESTIONS:

- FAQ-ABO100 [Which server log files or operating systems are supported ?](#)
- FAQ-ABO150 [Which log format can AWStats analyze ?](#)
- FAQ-ABO200 [Which languages are available ? How to add my own language ?](#)
- FAQ-ABO250 [Can AWStats be integrated with PHP Nuke ?](#)
- FAQ-ABO300 [About AWStats history](#)

COMMON SETUP/USAGE QUESTIONS:

Here, you can find the most common questions and answers about AWStats setup/usage process.

- FAQ-COM025 [How to use AWStats with no server log](#)
- FAQ-COM050 [What is the log size limit AWStats can analyze ?](#)
- FAQ-COM090 [Setup for FTP server log files \(proftpd, vsftpd, ...\).](#)
- FAQ-COM100 [Setup for MAIL log files \(Postfix, Sendmail, QMail, MDAemon, Exchange\).](#)
- FAQ-COM110 [Setup for MEDIA SERVER log files \(Realmedia, Windows media, Darwin streaming server\).](#)
- FAQ-COM115 [Setup/Examples for LogFormat parameter.](#)
- FAQ-COM120 [How to rotate my logs without losing data.](#)
- FAQ-COM130 [How to run AWStats frequently ?](#)
- FAQ-COM140 [How to exclude my IP address \(or whole subnet mask\) from stats ?](#)
- FAQ-COM142 [How to get the screen size and browser capabilities report working ?](#)
- FAQ-COM145 [How to use the Extra Sections features ?](#)
- FAQ-COM150 [Benchmark question.](#)
- FAQ-COM200 [How reverse DNS Lookup works, unresolved IP Addresses ?](#)
- FAQ-COM250 [Different results than other log analyzers \(Analog, Webalizer, WUUsage, wwwStats...\).](#)
- FAQ-COM300 [Difference between local hour and AWStats reported hour.](#)
- FAQ-COM320 [What does mean "eu \(European country\)" in GeoIP country reports.](#)
- FAQ-COM350 [How can I process old log file ?](#)
- FAQ-COM360 [How can I process several log files in one run ?](#)
- FAQ-COM400 [How can I update my statistics when I use a load balancing system that splits my logs ?](#)
- FAQ-COM500 [How can I reset all my statistics ?](#)
- FAQ-COM600 [How can I compile and build statistics on a daily basis only ?](#)
- FAQ-COM700 [Can I safely remove a line in AWStats history files \(awstatsMMYYYY*.txt\) ?](#)

ERRORS/TROUBLESHOOTING QUESTIONS:

Here, you can find the most common questions and answers about errors or problems using AWStats.

- FAQ-SET050 [Error "Missing \\$ on loop variable ..."](#)
- FAQ-SET100 [I see Perl script's source instead of its execution in my browser.](#)
- FAQ-SET150 [Error "...couldn't create/spawn child process..." with Apache for windows.](#)
- FAQ-SET200 ["Internal Error" or "Error 500" in a browser connecting to Apache.](#)
- FAQ-SET210 ["Internal Error" after a long time in my browser \(See FAQ-COM100 "AWStats speed/timeout problems"\).](#)
- FAQ-SET220 [Crash while running awstats.pl or page content only partially loaded](#)
- FAQ-SET270 [Only corrupted/dropped records](#)
- FAQ-SET280 [Error "Not same number of records of..."](#)
- FAQ-SET300 [Error "Couldn't open file ..."](#)
- FAQ-SET320 [Error "Malformed UTF-8 character \(unexpected..."](#)
- FAQ-SET350 [Empty or null statistics reported.](#)
- FAQ-SET360 [Statistics reported except for os, browsers, robots and keywords/keyphrases.](#)
- FAQ-SET400 [Pipe redirection to a file give me an empty file.](#)
- FAQ-SET450 [No pictures/graphics shown.](#)
- FAQ-SET700 [My visits are doubled for old month I migrated from 3.2 to 5.x](#)
- FAQ-SET750 [AWStats run out of memory during update process with cygwin Perl.](#)
- FAQ-SET800 [AWStats speed/timeout problems.](#)

SECURITY QUESTIONS:

Here, you can find the common questions about security problems when setting or using AWStats.

FAQ-ABO100 : WHICH SERVER LOG FILES OR OS ARE SUPPORTED ?

AWStats can works with :

- All web server able to write log file with a combined log format (XLF/ELF) like Apache, a common log format (CLF) like Apache or Squid, a W3C log format like IIS 5.0 or higher, or any other log format that contains all information AWStats expect to find.
 - Most of all others Web/Wap/Proxy/Streaming servers.
 - Some FTP, Syslog or Mail log files.
- Because AWStats is in Perl, it can works on all Operating Systems.

Examples of used platforms (bold means 'tested by author', others were reported by AWStats users to work correctly) :

OS:

Windows 2000, Windows NT 4.0, Windows Me, **Linux (RedHat, Mandrake, Debian, Suse...)**, Macintosh, **Solaris**, Aix, BeOS, FreeBSD, ...

Web/Wap/Proxy/Streaming servers

Apache 1.3.x and 2.x, IIS 5.0 or higher, ISA, WebStar, WebLogic, WebSite, **Windows Media Server**, Tomcat, **Squid**, Sambar, Roxen, Resin, RealMedia server, Oracle9iAS, **Lotus Notes/Domino**, Darwin, IPlanet, IceCast, ZeroBrand, Zeus, Zope, Abyss...

FTP servers

ProFTPD, vsFTPD...

Mails servers

Postfix, Sendmail, QMail, **Mdaemon**, www4mail, ...

Perl interpreters (all Perl >= 5.005):

ActivePerl 5.6, ActivePerl 5.8, Perl 5.8, Perl 5.6, Perl 5.005, mod_perl and **mod_perl2** for Apache, ...

FAQ-ABO150 : WHICH LOG FORMATS CAN AWSTATS ANALYZE ?

AWStats setup knows predefined log formats you can use to make AWStats config easier. However, you can define your own log format, that's the reason why AWStats can analyze nearly all web, wap and proxy server log files. Some FTP servers log files, Syslog or mail logs can also be analyzed.

The only requirement is "Your log file must contain required information".

This is very short examples of possible log format:

Apache common log format (see Note*),
Apache combined log format (known as NCSA combined log format or XLF or ELF format),
Any other personalized Apache log format,
Any IIS log format (known as W3C format),
Webstar native log format,
Realmedia server, Windows Media Server, Darwin streaming server,
ProFTPD server, vsFTPD server,
Postfix, Sendmail, QMail, Mdaemon
A lot of web/wap/proxy/streaming servers log format

*Note: Apache common log format (AWStats can now analyze such log files but such log files does not contain all information AWStats is looking for. The problem is in the content, not in the format). I think analyzing common log files is not interesting because there is a lot of missing information: no way to filter robots, find search engines, keywords, os, browser. But a lot of users asked me for it, so AWStats support it. However, a lot of interesting advanced features can't work: browsers, os's, keywords, robot detection...).

See also [F.A.Q.: LOG FORMAT SETUP OR ERRORS](#) .

FAQ-ABO200 : WHICH LANGUAGES ARE AVAILABLE ?

AWStats can make reports in 43 languages. This is a list of all of them, for last version, in alphabetical order (The code you can use for [Lang](#) and [ShowFlagLinks](#) parameter are the ISO-639-1 language codes):

Albanian=al, Bosnian=ba, Brezhoneg=bzg, Bulgarian=bg, Catalan=ca, Chinese (Taiwan)=tw, Chinese (Simplified)=cn, Croatian=hr, Czech=cz, Danish=dk, Dutch=nl, English=en, Estonian=et, Euskara=eu, Finnish=fi, French=fr, Galician=gl, German=de, Greek=gr, Hebrew=he, Hungarian=hu, Icelandic=is, Indonesian=id, Italian=it, Japanese=jp, Korean=ko, Latvian=lv, Norwegian (Nynorsk)=nn, Norwegian (Bokmal)=nb, Polish=pl, Portuguese=pt, Portuguese (Brazilian)=br, Romanian=ro, Russian=ru, Serbian=sr, Slovak=sk, Slovenian=si, Spanish=es, Swedish=se, Thai=th, Turkish=tr, Ukrainian=ua, Welsh=cy.

However, AWStats documentation is only provided in English.

But, you may find small documentation for other languages made by contributors on [Documentation Contrib page](#).

If your language is not in this list, you can translate it yourself. For this, find what is your 2 letter language code: [here](#). Once, you get it, for example "gl" for Galician, copy the file awstats-en.txt into awstats-gl.txt, in langs directory and translate every sentence inside. You can do same for files inside tooltips_f, tooltips_m and tooltips_w sub-directories. Then send your translated file(s) to eldy@users.sourceforge.net.

FAQ-ABO250 : CAN AWSTATS BE INTEGRATED WITH PHP NUKE ?

The only plugin I know to integrate AWStats inside PHPNuke is here: [PhpNuke addon for AWStats](#)

FAQ-ABO300 : ABOUT AWSTATS HISTORY

AWStats was initially designed for my own use to track visitors on my personal web sites or other projects i worked on ([www.chiensderace.com](#), [www.chatsderace.com](#), [www.lesbonnesannonces.com](#), [www.pourmaplanete.com](#), [www.dolibarr.org](#), [www.nitechno.com](#), and [www.destailleur.fr](#))

Then I decided to put it on sourceforge in year 2000. Then a lot of new versions were developed to add enhancements until today. See changelog for full history of changes.

FAQ-COM025 : HOW TO USE AWSTATS WITH NO SERVER LOG FILE

PROBLEM:

I want to have AWStats statistics but i have no access to my server log file.

SOLUTION:

Because AWStats is a log analyzer, if you don't have any way to read your server log file, you have nothing to analyze and you should not be able to use AWStats. However, this is a trick that you can use to have a log file be built. You must add a tag to call a CGI script like pslogger into each of your web pages. This will allow you to have an artificial log file that can be analyzed by AWStats.

You can find a Perl version of CGI pslogger enhanced by AWStats author [here](#) or a php version of CGI pslogger made by Florent CHANTRET [here](#).

FAQ-COM050 : WHAT IS THE LOG SIZE LIMIT AWSTATS CAN ANALYZE

PROBLEM:

I know I must run AWStats update process frequently on new log files, this means those files have a regular size, but for my first update, I want/need to run update process on old log files that are very large. Is there a limit on log file size AWStats can analyze ?

SOLUTION:

No. There is no limit in AWStats. This means you can use it on large log files (test were made on 10GB log files). However your system (Operating System or Perl version) might have a limit. For example, you can experience size limit errors on files larger than 2 or 4 GB. If limit is Perl only, try to use a Perl version compiled with "large file" option.

~~If you can't find it nor build it, you can try to use a LogFile parameter that looks like this~~

`LogFile="cat /yourlogfilepath/yourlogfile |"`

instead of

`LogFile="/yourlogfilepath/yourlogfile"`

FAQ-COM090 : SETUP FOR FTP SERVER LOG FILES (proftpd, vsftpd, ...)

PROBLEM:

What do I have to do to use AWStats to analyze some FTP server log files ?

SOLUTION:

AWStats can be used with some FTP server log files.

With ProFTPD:

1- Setup your server log file format:

Modify the proftpd.conf file to add the following two lines :

`LogFormat awstats "%t %h %u %m %f %s %b" # WARNING: You must use a tab char between % tags and not a space !
ExtendedLog /var/log/xferlog read,write awstats # WARNING: ExtendedLog directive might need to be placed inside a virtual host context if you use them.`

Then turn off old format Transfer log:

TransferLog none # WARNING: TransferLog directive might need to be placed inside a virtual host context if you use them.

To have the change effective, stop your server, remove old log file /var/log/xferlog and restart the server.
Download a file by FTP and check that your new log file looks like this:

[01/Jan/2001:21:49:57 +0200] ftp.server.com user RETR /home/fileiget.txt 226 1499

2- Then setup AWStats to analyze the FTP log file:

Copy config file "awstats.model.conf" to "awstats.ftp.conf".

Modify this new config file:

```
LogFile="/var/log/xferlog"
LogType=F
LogFormat="%time1 %host %logname %method %url %code %bytesd"
LogSeparator="\t"
NotPageList=""
LevelForBrowsersDetection=0
LevelForOSDetection=0
LevelForRefererAnalyze=0
LevelForRobotsDetection=0
LevelForWormsDetection=0
LevelForSearchEnginesDetection=0
ShowLinksOnUrl=0
ShowMenu=1
ShowSummary=UVHB
ShowMonthStats=UVHB
ShowDaysOfMonthStats=HB
ShowDaysOfWeekStats=HB
ShowHoursStats=HB
ShowDomainsStats=HB
ShowHostsStats=HBL
ShowAuthenticatedUsers=HBL
ShowRobotsStats=0
ShowEMailSenders=0
ShowEMailReceivers=0
ShowSessionsStats=1
ShowPagesStats=PBEX
ShowFileTypesStats=HB
ShowFileSizesStats=0
ShowBrowsersStats=0
ShowOSStats=0
ShowOriginStats=0
ShowKeyphrasesStats=0
ShowKeywordsStats=0
ShowMiscStats=0
ShowHTTPErrorsStats=0
ShowSMTPErrorsStats=0
```

Now you can use AWStats as usual (run the update process and read statistics).

With vsFTPd, or any FTP server that log with xferlog format:

1- Check your server log file format:

Take a look at your FTP server log file. You must have a format that match the following example to use this FAQ :

Wed Jan 01 19:29:35 2001 1 192.168.1.1 102 /home/file1.txt b_o r username ftp 0 * c

2- Then setup AWStats to analyze the FTP log file:

If your FTP log file format looks good, copy config file "awstats.model.conf" to "awstats.ftp.conf".

Modify this new config file:

```
LogFile="/var/log/xferlog"
LogType=F
LogFormat="%time3 %other %host %bytesd %url %other %other %method %other %logname %other %code
%other %other"
LogSeparator="\s"
NotPageList=""
```

```

LevelForBrowsersDetection=0
LevelForOSDetection=0
LevelForRefererAnalyze=0
LevelForRobotsDetection=0
LevelForWormsDetection=0
LevelForSearchEnginesDetection=0
ShowLinksOnUrl=0
ShowMenu=1
ShowSummary=UVHB
ShowMonthStats=UVHB
ShowDaysOfMonthStats=HB
ShowDaysOfWeekStats=HB
ShowHoursStats=HB
ShowDomainsStats=HB
ShowHostsStats=HBL
ShowAuthenticatedUsers=HBL
ShowRobotsStats=0
ShowEMailSenders=0
ShowEMailReceivers=0
ShowSessionsStats=1
ShowPagesStats=PBEX
ShowFileTypesStats=HB
ShowFileSizesStats=0
ShowBrowsersStats=0
ShowOSStats=0
ShowOriginStats=0
ShowKeyphrasesStats=0
ShowKeywordsStats=0
ShowMiscStats=0
ShowHTTPErrorsStats=0
ShowSMTPErrorsStats=0

```

Now you can use AWStats as usual (run the update process and read statistics).

FAQ-COM100 : SETUP FOR MAIL LOG FILES (Postfix, Sendmail, Qmail, MDAemon, Exchange...)

PROBLEM:

What do I have to do to use AWStats to analyze my mail log files ?

SOLUTION:

This tip works with AWStats 5.5 or higher.

For Postfix, Sendmail, QMail or MDAemon log files

You must setup AWStats to use a mail log file preprocessor (*maillogconvert.pl* is provided into AWStats *tools* directory, but you can use the one of your choice):

For this, copy config "awstats.model.conf" file to "awstats.mail.conf".

Modify this new config file: For standard Postfix, Sendmail, MDAemon and standard QMail logfiles, set

```
LogFile="perl /path/to/maillogconvert.pl standard < /path/to/maillog/maillog |"
```

If the logfiles are compressed, they can be processed this way

```
LogFile="gzip -cd /var/log/maillog.0.gz | /path/to/maillogconvert.pl standard |"
```

And for VAdmin QMail logfiles (multi-host/virtualhost mail servers running vadmin software), set

```
LogFile="perl /path/to/maillogconvert.pl vadmin < /path/to/maillog/maillog |"
```

Then, whatever is you mail server, you must also change:

```

LogType=M
LogFormat="%time2 %email %email_r %host %host_r %method %url %code %bytesd"
LevelForBrowsersDetection=0
LevelForOSDetection=0
LevelForRefererAnalyze=0
LevelForRobotsDetection=0
LevelForWormsDetection=0
LevelForSearchEnginesDetection=0
LevelForFileTypesDetection=0
ShowMenu=1
ShowSummary=HB
ShowMonthStats=HB
ShowDaysOfMonthStats=HB

```



```
ShowDaysOfWeekStats=HB
ShowHoursStats=HB
ShowDomainsStats=0
ShowHostsStats=HBL
ShowAuthenticatedUsers=0
ShowRobotsStats=0
ShowEMailSenders=HBML
ShowEMailReceivers=HBML
ShowSessionsStats=0
ShowPagesStats=0
ShowFileTypesStats=0
ShowFileSizesStats=0
ShowBrowsersStats=0
ShowOSStats=0
ShowOriginStats=0
ShowKeyphrasesStats=0
ShowKeywordsStats=0
ShowMiscStats=0
ShowHTTPErrorsStats=0
ShowSMTPErrorsStats=1
```

Warning: For MDAemon mail server, you must use the new MDAemon log file that ends with "-Statistics.log".

Now you can use AWStats as usual (run the update process and read statistics).

For Exchange log files

Despite the high number of possible log format provided with Exchange, none of them is built enough seriously to offer an interesting analyze (missing informations, messy data, no id to join multiple records for same mail, etc...). For this reason, an "exact" log analysis is a joke with Exchange log files. However a little support is provided. In order to analyze Exchange traffic, you have to enable "Message Tracking" (see article <http://support.microsoft.com/default.aspx?scid=kb;EN-US;246856>). Then copy config awstats.model.conf file to "awstats.mail.conf".

Modify this new config file:

```
LogType=M
LogFormat="%time2 %email %email_r %host %host_r %method %url %code %bytesd"
LevelForBrowsersDetection=0
LevelForOSDetection=0
LevelForRefererAnalyze=0
LevelForRobotsDetection=0
LevelForWormsDetection=0
LevelForSearchEnginesDetection=0
LevelForFileTypesDetection=0
ShowMenu=1
ShowSummary=HB
ShowMonthStats=HB
ShowDaysOfMonthStats=HB
ShowDaysOfWeekStats=HB
ShowHoursStats=HB
ShowDomainsStats=0
ShowHostsStats=HBL
ShowAuthenticatedUsers=0
ShowRobotsStats=0
ShowEMailSenders=HBML
ShowEMailReceivers=HBML
ShowSessionsStats=0
ShowPagesStats=0
ShowFileTypesStats=0
ShowFileSizesStats=0
ShowBrowsersStats=0
ShowOSStats=0
ShowOriginStats=0
ShowKeyphrasesStats=0
ShowKeywordsStats=0
ShowMiscStats=0
ShowHTTPErrorsStats=0
ShowSMTPErrorsStats=1
```

Also don't forget that with Exchange, informations in a log analyses can't be exact. Do not send any questions or requests

for using AWStats with Exchange, this is not a problem in AWStats and we have no time to support non opened products. If you want to have complete and accurate information with Exchange, forget using AWStats or use a more serious mail serveur (Postfix, Sendmail, QMail...)

FAQ-COM110 : SETUP FOR A MEDIA SERVER (REALMEDIA, WINDOWS MEDIA SERVER, DARWIN STREAMING SERVER)

PROBLEM:

What do I have to do to use AWStats to analyze my Media Server log files.

SOLUTION:

For Realmedia

Your log file will probably looks like this:

```
216.125.146.50 -- [16/Sep/2002:14:57:21 -0500] "GET cme/rhythmcity/rcitycaddy.rm?cloakport=8080,554,7070
RTSP/1.0" 200 6672 [Win95_4.0_6.0.9.374_play32_NS80_en-US_586]
[80d280e1-c9ae-11d6-fa53-d52aaed98681] [UNKNOWN] 281712 141 3 0 0 494
```

Copy config awstats model conf file to "awstats mediaserver conf" Modify this new config file:

```
LogFile="/pathtomediaseverlog/mediaseverlog"
LogType=S
LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %uabracet %other %other %other
%other %other %other %other %other"
LogSeparator="\s+"
ShowMenu=1
ShowSummary=UHB
ShowMonthStats=UHB
ShowDaysOfMonthStats=HB
ShowDaysOfWeekStats=HB
ShowHoursStats=HB
ShowDomainsStats=HB
ShowHostsStats=HBL
ShowAuthenticatedUsers=0
ShowRobotsStats=0
ShowEMailSenders=0
ShowEMailReceivers=0
ShowSessionsStats=0
ShowPagesStats=PB
ShowFileTypesStats=HB
ShowFileSizesStats=0
ShowBrowsersStats=1
ShowOSStats=1
ShowOriginStats=H
ShowKeyphrasesStats=0
ShowKeywordsStats=0
ShowMiscStats=0
ShowHTTPErrorsStats=1
ShowSMTPErrorsStats=0
```

Now you can use AWStats as usual (run the update process and read statistics).

For Windows Media Server / Darwin Streaming Server

1- If your Windows Media / Darwin streaming Server version allows it, setup your log format to write the following fields:

```
c-ip
date
time
cs-uri-stem
c-starttime
x-duration
c-rate
c-status
c-playerid
c-playerversion
c-playerlanguage
cs(User-Agent)
cs(Referer)
```

```
c-hostexe  
c-hostexever  
c-os  
c-osversion  
c-cpu  
filelength  
filesize  
avgbandwidth  
protocol  
transport  
audiocodec  
videocodec  
channelURL  
sc-bytes
```

To make the change effective, stop your server, remove old log files and restart the server.

Listen to streaming files and check that your new log file looks like this:

```
80.223.91.37 2002-10-08 14:18:58 mmst://mydomain.com/mystream 0 106 1 200  
{F4A826EE-FA46-480F-A49B-76786320FC6B} 8.0.0.4477 fi-FI -- wmpplayer.exe 8.0.0.4477 Windows_2000  
5.1.0.2600 Pentium 0 0 20702 mms TCP Windows_Media_Audio_9 -- 277721
```

If your Windows Media/Darwin Streaming Server version does not allow to define your log format:

Just follow instructions in step 2 directly but use the log format string found in first lines of your log files (Just after the "#Fields" string) as value for AWStats LogFormat parameter. For example, you could have a LogFormat defined like this:

```
LogFormat="c-ip date time c-dns cs-uri-stem c-starttime x-duration c-rate c-status c-playerid c-playerversion  
c-playerlanguage cs(User-Agent) cs(Referer) c-hostexe c-hostexever c-os c-osversion c-cpu filelength filesize  
avgbandwidth protocol transport audiocodec videocodec channelURL sc-bytes c-bytes s-pkts-sent  
c-pkts-received c-pkts-lost-client c-pkts-lost-net c-pkts-lost-cont-net c-resendreqs c-pkts-recovered-ECC  
c-pkts-recovered-resent c-buffercount c-totalbuffertime c-quality s-ip s-dns s-totalclients s-cpu-util"
```

This means you don't use the AWStats tags but AWStats can often also understand all the IIS and/or Windows Media Server tags.

2- Then setup AWStats to analyze your Media Server log:

Copy config awstats.model.conf file to "awstats.mediaserver.conf".

Modify this new config file:

```
LogFile="/pathtomediaserver/mediaserverlog"  
LogType=S  
LogFormat="c-ip date time cs-uri-stem c-starttime x-duration c-rate c-status c-playerid c-playerversion  
c-playerlanguage cs(User-Agent) cs(Referer) c-hostexe c-hostexever c-os c-osversion c-cpu filelength filesize  
avgbandwidth protocol transport audiocodec videocodec channelURL sc-bytes"  
DecodeUA=1  
ShowMenu=1  
ShowSummary=UHB  
ShowMonthStats=UHB  
ShowDaysOfMonthStats=HB  
ShowDaysOfWeekStats=HB  
ShowHoursStats=HB  
ShowDomainsStats=HB  
ShowHostsStats=HBL  
ShowAuthenticatedUsers=0  
ShowRobotsStats=0  
ShowEMailSenders=0  
ShowEMailReceivers=0  
ShowSessionsStats=0  
ShowPagesStats=PB  
ShowFileTypesStats=HB  
ShowFileSizesStats=0  
ShowBrowsersStats=1  
ShowOSStats=1  
ShowOriginStats=H  
ShowKeyphrasesStats=0  
ShowKeywordsStats=0  
ShowMiscStats=0  
ShowHTTPErrorsStats=1  
ShowSMTPErrorsStats=0
```

Now you can use AWStats as usual (run the update process and read statistics).

FAQ-COM115 : SETUP/EXAMPLES FOR LOGFORMAT PARAMETER

PROBLEM:

Which value do I have to put in the LogFormat parameter to make AWStats working with my log file format ?

SOLUTION:

The AWStats config file gives you all the possible values for LogFormat parameter. To help you, this is some common cases of log file format, and the corresponding value of LogFormat you must use in your AWStats config file:

If your log records are EXACTLY like this (**NCSA combined/XLF/ELF log format**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 1234 "http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)"

You must use : LogFormat=1

This is same than: LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot"

If your log records are EXACTLY like this (**NCSA combined with several virtualhostname sharing same log file**):

virtualserver1 62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 1234

"http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)"

You must use : LogFormat="%virtualname %host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot"

If your log records are EXACTLY like this (**NCSA combined and mod_gzip format 1 with Apache 1.x**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 3904 "http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)" mod_gzip: 66pct.

You must use : LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot %other %gzipratio"

If your log records are EXACTLY like this (**NCSA combined and mod_gzip format 2 with Apache 1.x**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 3904 "http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)" mod_gzip: DECHUNK:OK In:11393 Out:3904:66pct.

You must use : LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot %other %other %gzipin %gzipout"

If your log records are EXACTLY like this (**NCSA combined and mod_deflate with Apache 2**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 3904 "http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)" (45)

You must use : LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot %deflateratio"

If your log records are EXACTLY like this (**NCSA combined with 2 spaces between some fields with Zope**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 3904 "http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)" (45)

You must use :

LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %refererquot %uaquot"

LogSeparator="**"

If your log records are EXACTLY like this (**NCSA common CLF log format**):

62.161.78.73 - - [dd/mm/yy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 1234

You must use : LogFormat=4

Note: Browsers, OS's, Keywords and Referers features are not available with a such format.

If your log records are EXACTLY like this (With some **Squid** versions, after setting *emulate http log to on*):

200.135.30.181 - - [dd/mm/yy:hh:mm:ss +0x00] "GET http://www.mydomain.com/page.html HTTP/1.0" 200 456 TCP_CLIENT_REFRESH_MISS:DIRECT

You must use : LogFormat="%host %other %logname %time1 %methodurl %code %bytesd %other"

If your log records are EXACTLY like this (Some old **IIS W3C log format**):

yy:yy-mm-dd hh:mm:ss 62.161.78.73 - GET /page.html 200 1234 HTTP/1.1

Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0) http://www.from.com/from.html

You must use : LogFormat=2

If your log records are EXACTLY like this (Some **IIS W3C log format with some .net servers**):

yy:yy-mm-dd hh:mm:ss GET /page.html - 62.161.78.73 - Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0) http://www.from.com/from.html 200 1234 HTTP/1.1

You must use : LogFormat=2 (or LogFormat="%time2 %method %url %logname %host %other %ua %referer %code

%bytesd %other")

If your log records are EXACTLY like this (Some **IIS 6+** W3C log format):

yyyy-mm-dd hh:mm:ss GET /page.html - 62.161.78.73 - Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+NT+5.0)
http://www.from.com/from.html 200 1234

You must use : LogFormat=2 (or LogFormat="date time cs-method cs-uri-stem cs-username c-ip cs-version
cs(User-Agent) cs(Referer) sc-status sc-bytes")

If your log records are EXACTLY like this (Some **ISA** W3C log format):

62.161.78.73, anonymous, Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1), N, 1/1/2001, 0:00:16, W3ReverseProxy,
HCSERV2, -, www.host.be, 192.168.141.101, 80, 266, 406, 10042, http, TCP, GET, http://192.168.141.101/, text/html,
Inet, 200, 0x42330010, -, -

You must use :

LogFile="sed -e 's/, /&g' "/yourlogpath/yourlogfile.log" |"

LogFormat=2

LogSeparator=" " "

If your log records are EXACTLY like this (With some **WebSite** versions):

yyyy-mm-dd hh:mm:ss 62.161.78.73 - 192.168.1.1 80 GET /page.html - 200 11205 0 0 HTTP/1.1 mydomain.com
Mozilla/4.0+(compatible;+MSIE+5.5;+Windows+98) - http://www.from.com/from.html

You must use : LogFormat="%time2 %host %logname %other %other %method %url %other %code %bytesd %other
%other %other %other %ua %other %referer"

If your log records are EXACTLY like this (**Webstar** native log format):

05/21/00 00:17:31 OK 200 212.242.30.6 Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt) http://www.cover.dk/
"www.cover.dk" :Documentation:graphics:starninelogo.white.gif 1133

You must use : LogFormat=3

If your log records are EXACTLY like this (With some **Lotus Notes/Domino** versions):

62.161.78.73 - Name Surname Service [dd/mmm/yyyy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 1234
"http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)"

You must use : LogFormat=6

If your log records are EXACTLY like this (**Lotus Notes/Domino 6.x** log format):

62.161.78.73 - "Name Surname" Service [dd/mmm/yyyy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" 200 1234
"http://www.from.com/from.htm" "Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)"

You must use : LogFormat="%host %other %lognamequot %time1 %methodurl %code %bytesd %refererquot %uaquot"

If your log records are EXACTLY like this (With **Oracle9iAS**):

62.161.78.73 - [dd/mmm/yyyy:hh:mm:ss +0x00] GET /page.html HTTP/1.1 200 1234 - "Mozilla/4.0 (compatible; MSIE
5.01; Windows NT 5.0)"

Where separators are "tab" characters or several "spaces", You must use : LogFormat="%host %logname %time1
%method %url %other %code %bytesd %referer %uaquot" and LogSeparator="\s+"

If you use a FTP server like **ProFTPD**:

See [FAQ-COM090](#).

If you want to analyze a mail log file (**Postfix**, **Sendmail**, **QMail**, **MDaemon**, **Exchange**):

See [FAQ-COM100](#).

If you use a Media Server (**Realmedia**, **Windows Media Server**):

See [FAQ-COM110](#).

If your log records are EXACTLY like this (With some providers):

62.161.78.73 - - [dd/Month/yyyy:hh:mm:ss +0x00] "GET /page.html HTTP/1.1" "-" 200 1234

You must use : LogFormat="%host %other %logname %time1 %methodurl %other %code %bytesd"

Note: Browsers, OS's, Keywords and Referers features are not available with a such format.

There are a lot of other possible log formats.

You must use a personalized log format LogFormat = "..." as described in config file to support other various log formats.

FAQ-COM120 : HOW TO ROTATE MY LOGS WITHOUT LOSING DATA

PROBLEM:

I want to archive/rotate my logs using my server system (for example logrotate) options or a third software (rotatelog,

cronolog) but I don't want to lose any visits information during the rotate process.

SOLUTION:

- If your config file is setup with a [LogFile](#) parameter that point to your current running log file (required if you want to use the [AllowToUpdateStatsFromBrowser](#) option to have "real-time" statistics), to avoid losing too much records during the rotate process, you must run the AWStats update JUST BEFORE the rotate process is done.

The best way to do that on 'Linux like' OS is to use the linux built-in logrotate feature. You must edit the logrotate config file used for your web server log file (usually stored in /etc/logrotate.d directory) by adding the AWStats update process as a preprocessor command, like this example (bold lines are lines to add for having a prerotate process):

```
/usr/local/apache/logs/*log
{
notifempty
daily
rotate 7
compress
sharedscripts
prerotate
/usr/local/awstats/wwwroot/cgi-bin/awstats.pl -update -config=mydomainconfig
endscript
postrotate
/usr/bin/killall -HUP httpd
endscript
}
```

If using a such solution, this is sequential steps that happens:

Step	Description	Step name	Date/Time example
A	logrotate is started (by cron)	Start of logrotate	04:02:00
B	awstats -update is launched by logrotate	Start of awstats	04:02:01
C	awstats start to read the log file <i>file.log</i>		04:02:02
D	awstats has reached the end of log file so now it starts to save its database on disk.		04:05:00
E	awstats has finished to save its new database. so it stops	End of awstats	04:06:00
F	logrotate moves old log file <i>file.log</i> to a new name <i>file.log.sav</i> . Apache now logs in this file <i>file.log.sav</i> since log file handle has not been changed (only log file name has been renamed).	Log move	04:06:01
G	logrotate sends the -HUP or -USR1 signal to Apache. With -HUP, Apache immediatly kills all its child process/thread, close log file <i>file.log.sav</i> , and reopen file <i>file.log</i> . So now, ALL hits are written to new file. With -USR1, Apache only ask its child process/thread to stop only when HTTP request will be completely served. However it closes immediatly log file <i>file.log.sav</i> , and reopen file <i>file.log</i> . So only NEW hits are written to new log file. HTTP requests that are still running will write in old one.	Apache restart	04:06:02
H	logrotate starts compress the old log file <i>file.log.sav</i> into <i>file.log.gz</i>	Start compress	04:06:03
I	If some apache threads/processes are still running (because the kill sent was -USR1, so child process are waiting end of request before to stop), then those threads/processes are still writing to <i>file.log.sav</i> . If kill -HUP was used, all process are already restarted so all writes in new <i>file.log</i> .		
J	logrotate has finished to compress log file into <i>file.log.gz</i> . File <i>file.log.sav</i> is deleted.	End of compress End of logrotate	04:07:03
K	If signal was -USR1, some old childs can still run (when serving a very long request for example). So the log writing, still done in same file handle are going to a file that has been removed. So log writing are lost nowhere (this is only if -USR1 was used and if request was very long).		

The advantage of this solution is that it is a very common way of working, used by a lot of products, and easy to setup. You will notice that you can "lose" some hits:

If you use the `-HUP` signal, you will only lose all hits that were written during D and E. Note that you will also break all requests still running at G. In the example, it's a 1 minute lost (for small or medium web sites, it will be less than few seconds), so this give you an error lower than 0.07% (less for small web sites). This is not significant, above all for a "statistics" program.

If you use the `-USR1` signal, you will not kill any request. But you will lose all hits that were wrote during D and E (like with `-HUP`) but also all hits that are still running after H (all very long request that requires several minutes to be served). If hit ends during I, it is wrote in a log file already analyzed, if hit ends at K, it is wrote nowhere. In the example, it's also a 0.07% error plus error for other not visible hits that were finished during I or K, but number of such hits should be very low since only hits that started before G and not finished after H are concerned. In most cases a hits needs only few milliseconds to be served so lost hits could be ignored.

Note also that if you have x logrotate config files, with each of them a postrotate with a kill `-HUP`, you send a kill x times to your server process. So try to include several log files in same logrotate config file. You can have several awstats update command in the same prerotate section and you will send the `-HUP` only once, after all updates are finished. However, doing this, you will have a lap time between D and F (were some hits are lost) that will be higher.

- Another common way of working is to choose to run the AWStats update process only once the log file has been archived. This is required for example if you use the [cronolog](#) or [rotatelog](#) tools to rotate your log files. For example, Apache users can setup their Apache httpd config file to write log file through a pipe to [cronolog](#) or [rotatelog](#) using Apache *CustomLog* directive:

`CustomLog "|/usr/sbin/cronolog [cronolog_options] /var/logs/access.%Y%m%d.log" combined`

If you use a such feature, you can't trigger AWStats update process to be ran just BEFORE the rotate is done, so you must run it AFTER the rotate process, so on the archived log file.
To setup awstats to always point to last archive log file, you can use the 'tags' available for [LogFile](#).
The problem with that is that your data are refreshed only after a rotate has done. However, you will miss absolutely nothing (no hits) and your server processes are never killed.
- So, if you really want to not lose absolutely no hit and want to have updates more frequently than the rotate frequency, the best way is still an hybrid solution (i am not sure that it worth the pain, and remember that statistics are only statistics): You run the awstats update process from you crontab frequently, every hour for example, and half and hour before the rotate has done. See next FAQ to know how to setup a scheduled job.
Then, once the rotate has been done (by the logrotate or by a piped cronolog log file), and before the next scheduled awstats update process start, you run another update process on the archived log file using the `-logfile` option to force update on the archived log file and not the current log file defined in awstats config file. This will allow you to update the half hour missing, until the log rotate (AWStats will find the new lines). However don't forget that this particular update MUST be finished before the next cronolog update.

FAQ-COM130 : HOW TO RUN AWSTATS UPDATE PROCESS FREQUENTLY

PROBLEM:

AWStats must be ran frequently to update statistics. How can I do this ?

SOLUTION:

A good way of working is to run the AWStats update process as a preprocessor of your log rotate process. See previous FAQ ([FAQ-COM120](#)) for this.

But you can also run AWStats update process regularly by a scheduler:

With Windows, you can use the internal task scheduler. The use of this tool is not an AWStats related problem, so please take a look at your Windows manual. Warning, if you use "`awstats.pl -config=mysite -update`" in your scheduled task, you might experience problem of failing task. Try this instead

`"C:\WINNT\system32\CMD.EXE /C C:[awstats_path]\awstats.pl -config=mysite -update"`

or

`"C:[perl_path]\perl.exe C:[awstats_path]\awstats.pl -config=mysite -update"`

A lot of other open source schedulers are often better (otherwise there is also good sharewares or freewares).

With unix-like operating systems, you can use the "**crontab**".

This is examples of lines you can add in the cron file (see your unix reference manual for cron) :

To run update every day at 03:50, use :

`50 3 * * * /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=mysite -update >/dev/null`

To run update every hour, use :

`0 * * * * /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=mysite -update >/dev/null`

FAQ-COM140 : HOW CAN I EXCLUDE MY IP ADDRESS (OR WHOLE SUBNET MASK) FROM STATS ?

PROBLEM:

I don't want to see my own IP address in the stats or I want to exclude counting visits from a whole subnet.

SOLUTION:

You must edit the config file to change the [SkipHosts](#) parameter.

For example, to exclude:

- your own IP address 123.123.123.123, use [SkipHosts](#)="123.123.123.123"
- the whole subnet 123.123.123.xxx, use [SkipHosts](#)="REGEX[^\123\123\123\.]"
- all sub hosts xxx.myintranet.com, use [SkipHosts](#)="REGEX[\.myintranet\.com\$]" (This one works only if DNS lookup is already done in your log file).

FAQ-COM142 : HOW TO GET THE SCREEN SIZE AND BROWSER CAPABILITIES REPORT WORKING ?

PROBLEM:

I see in the AWStats features list that it can report the screen size used by visitors and other browsers' informations, (like if browser support Flash, Java, Javascript, PDF, MACromedia, Audio plugins, etc...). How can I do that ?

SOLUTION:

AWStats is a log analyzer, so to report the screen size of your visitor, we need to have information inside the log file itself. For this, the only way to do that, is to add some HTML tags inside some of your pages (the home page is enough to get use ratios). This tag will add call to a javascript that ask your browser to get a virtual URL that includes, in its parameters, the screen size resolution and all other informations about browser capabilities (Flash, Java, Javascript, PDF, Macromedia, Audio plgins...).

This is the code you must add (at bottom of your home page for example) :

```
<script language="javascript" type="text/javascript" src="/js/awstats_misc_tracker.js" ></script>
<noscript></noscript>
```

Note that you must also place the *awstats_misc_tracker.js* script (provided in /js directory with AWStats) inside a js directory stored in your web root.

Once this is done, load your home page with your browser and go to check that inside your log file if you can see a line that looks like that:

```
123.123.123.123 - - [24/Apr/2005:16:09:38 +0200] "GET
/js/awstats_misc_tracker.js?screen=800x600sid=awssession_id123 HTTP/1.1" 200 6237
"http://therefererwebsite.com/index.php" "Mozilla/5.0 (Linux) Gecko/20050414 Firefox/1.0.3"
```

If yes, you can then run the AWStats update process. Screen sizes information will be analyzed. All you have to do now is to edit your config file to tell AWStats to add the report on html output. For this, change the [ShowMiscStats](#) parameter.

[ShowMiscStats=anjdfgrqw](#)

Note: If you change directory where *awstats_misc_tracker.js* is stored (somewhere else than the /js directory), you must modify, according to your change:

- the html tags added
- the line: `var awstatsmiscstrackerurl="/js/awstats_misc_tracker.js";` inside the *awstats_misc_tracker.js* script
- the parameter [MiscTrackerUrl](#) inside AWStats configuration file.

FAQ-COM145 : HOW TO USE THE EXTRA SECTIONS FEATURES ?

PROBLEM:

I want to build personalized reports not provided in default AWStats reports. How can I setup the Extra Sections parameters in my AWStats config file to do so ?

SOLUTION:

Take a look at the [Using AWStats Extra Sections features](#)

FAQ-COM150 : BENCHMARK / FREQUENCY TO LAUNCH AWSTATS TO UPDATE STATISTICS

PROBLEM:

What is AWStats speed ?

What is the frequency to launch AWStats process to update my statistics ?

SOLUTION:

All benchmarks information and advice on frequency for update process are related into the [Benchmark page](#).

FAQ-COM200 : HOW REVERSE DNS LOOKUP WORKS. UNRESOLVED IP ADDRESSES

PROBLEM:

The reported page AWStats shows me has no hostnames, only IP addresses, countries reported are all "unknown".

SOLUTION:

When AWStats find an IP address in your log file, it tries a reverse DNS lookup to find the hostname and domain if the [DNSLookup](#) parameter, in your AWStats config file, is [DNSLookup=1](#) (Default value). So, first, check if you have the good value. The [DNSLookup=0](#) must be used only if your log file contains already resolved IP address. For example, when you

set up Apache with the *HostNameLookups=on* directive. When you ask your web server to make itself the reverse DNS lookup to log hostname instead of IP address, you will still find some IP addresses in your log file because the reverse DNS lookup is not always possible. But if your web server fails in it, AWStats will also fails (All reverse DNS lookups use the same system API). So to avoid AWStats to make an already done lookup (with success or not), you can set *DNSLookup=0* in AWStats config file. If you prefer, you can make the reverse DNS lookup on a log file before running your log analyzer (If you only need to convert a logfile with IP Addresses into a logfile with resolved hostnames). You can use for this [logresolve](#) tool provided with AWStats distribution (This tools is an improved version of *logresolve* provided with Apache).

FAQ-COM250 : DIFFERENT RESULTS THAN OTHER ANALYZER

PROBLEM:

I also use Webalizer, Analog (or another log analyzer) and it doesn't report the same results than AWStats. Why ?

SOLUTION:

If you compare AWStats results with an other log file analyzer, you will found some differences, sometimes very important. In fact, all analyzer (even AWStats) make "over reporting" because of the problem of proxy-servers and robots. However AWStats is one of the most accurate and its "over reporting" is very low where all other analyzers, even the most famous, have a VERY HIGH error rate (10% to 200% more than reality !).

This is the most important reasons why you can find important differences:

- Some dynamic pages generated by CGI programs are not counted by some analyzer (ie Webalizer) like a "Page" (but only like a "Hit") if CGI prog does not end with a defined extension (.cgi, ...), so they are not included correctly in their statistics. AWStats use an opposite policy, assuming a file is a page except if type is in a list (See [NotPageList](#) parameter). Error rate with a such policy is lower.
- Some log analyzers use the "Hits" to count visitors. This is a very bad way of working : Some visitors use a lot of proxy servers to surf (ie: AOL users), this means it's possible that several hosts (with several IP addresses) are used to reach your site for only one visitor (ie: one proxy server download the page and 2 other servers download all images). Because of this, if stats of unique visitors are made on "hits", 3 users are reported but it's wrong. So AWStats considers only HTML "Pages" to count unique visitors. This decrease the error, not completely, because it's always possible that a proxy server download one HTML frame and another one download another frame, but this make the over-reporting of unique visitors less important.
- Another important reason to have difference is that an error log files is not always completely sorted but only "nearly" sorted because of cache and writing log engines used by server. Nearly all log analyzers (commercial and not) assumes that log file is "exactly" sorted by hit date to calculate visits, entry and exit pages. But there is nothing that guaranties this and some log files are only "nearly" sorted, above all log files on highly loaded servers. AWStats has an advanced parsing algorithm that is able to count correctly visits, entry and exit pages even if log file is only "nearly" sorted.
- AWStats does not count twice (with default setup) redirects made by server "rewrite rules". Such rule makes two hits into log files, so most log analyzer count them twice, but only one page were "viewed".
- Then, there is internal bugs in log analyzers that make reports wrong. For example, a lot of users have reported that Webalizer "doubles" the number of visits or visitors in some circumstances.
- AWStats is able to detect robots visits. Most analyzers think robots visits are human visitors. This error make them to report more visits and visitors than reality. When AWStats reports a "1 visitor", it means "1 human visitor" (even if it's not possible to detect all robots, most of them are detected). "Robots visitors" are reported separately in the "Robots/Spiders visitors" chart. AWStats is a log analyzer with one of the most important robot database. In fact, a lot of other log analyzer uses an update copy of the AWStats robot database for their own use. However, even if a robot database is up to date, there is still some robot hits that are not possible to detect using log analyzing. For this reason, AWStats still report 10% more visits than reality because of such robots. This is the major reason that create differences between a log analyzer and a HTML tagger system like Google Analytics.

Now let see other minor reasons. However those points explains only very small differences (

- To differentiate new visits from same visitor, log analysers uses a visit time-out. If value differs, then result differs (for visit count and entry and exit pages). A such time-out is a fixed value (For example 60 minutes) meaning if a visitor make a hit 59 minutes after downloading the previous page, it's the same visits, if he make it 61 minutes after, it's a new visit. Of course, there is no realy difference between 59 and 61, but counting visits without time-out is not possible. And because the most important is to have a time-out (and not really it's value), AWStats time-out is not an "exact" value but is "around" 60 minutes. This allows AWStats to have better speed processing time, so you also might experience little differences, in visit count, between AWStats and another log analyzer even if their time-out are both defined to same value (because AWStats time-out is not exactly but nearly value defined).
- There is also differences in log analyzers databases and algorithms that make details of results less or more accurate: AWStats has a larger browsers, os', search engines and robots database, so reports concerning this are more accurate. AWStats has url syntax rules to find keywords or keyphrases used to find your site, but AWStats has also an algorithm to detect keywords of unknown search engines with unknown url syntax rule.
Etc...

If you want to check how serious your log analyzer is, try to parse the following log file. It's a very common log file but results will show you how bad most log analyzers are (above all commercial products):

```
# This is a sample of log file that contains a lot of various data we can find
# in a log file. Great sample to test reliability and accuracy of any log
# analyzer.
# -----
# This sample log file contains 10 differents IPs that are :
```



```

# Viewed traffic: 5 different true human visitors making 6 human visits
# 21 hits on pages and 15 hits not pages (36 hits)
# Not viewed traffic: 13 pages, 15 hits
# -----
# 80.8.55.1 2 visits (start at 00:00:00 and at 12:00:00 with both entry page on /)
# And 2 hits to add favourites but first is non root hit with error meaning it's same "add"
# 80.8.55.2 Not a visit, only an image included into a page of an other site
# 80.8.55.3 1 visit (and add home page to favourites)
# 80.8.55.4 same visitor than 80.8.55.3 using aol proxy
# 80.8.55.5 Not a visit (but bot indexing)
# 80.8.55.6 1 visit (authenticated visitor)
# 80.8.55.7 1 visit (authenticated visitor with space in name)
# 80.8.55.8 Not a visit (try but failed twice with 404 and 405 error)
# 80.8.55.9 Not a visit (but a worm attack)
# 80.8.55.10 1 visit that come from a web page that is not a search engine

80.8.55.1 - - [01/Jan/2001:00:00:10 +0100] "GET /pagel.html HTTP/1.0" 200 7009 "-" "Mozilla/4.0 (compatible; MSIE
80.8.55.1 - - [01/Jan/2001:00:00:00 +0100] "GET / HTTP/1.0" 200 7009 "http://www.sitereferer/cgi-bin/search.pl?q=
80.8.55.1 - - [01/Jan/2001:00:00:20 +0100] "GET /page2.cgi HTTP/1.0" 200 7009 "http://localhost/pagel.html" "Moz
80.8.55.1 - - [01/Jan/2001:00:00:25 +0100] "GET /page3 HTTP/1.0" 200 7009 "-" "Mozilla/4.0 (compatible; MSIE 5.5;
80.8.55.1 - - [01/Jan/2001:00:00:30 +0100] "GET /image.gif HTTP/1.0" 200 7009 "-" "Mozilla/4.0 (compatible; MSIE
80.8.55.1 - - [01/Jan/2001:00:00:35 +0100] "GET /image2.png HTTP/1.0" 200 7009 "-" "Mozilla/4.0 (compatible; MSIE
80.8.55.1 - - [01/Jan/2001:00:00:40 +0100] "GET /dir/favicon.ico HTTP/1.0" 404 299 "-" "Mozilla/4.0 (compatible;
80.8.55.1 - - [01/Jan/2001:00:00:40 +0100] "GET /favicon.ico HTTP/1.0" 200 299 "-" "Mozilla/4.0 (compatible; MSIE

80.8.55.1 - - [01/Jan/2001:12:00:00 +0100] "GET / HTTP/1.0" 200 7009 "http://WWW.SiteRefereR:80/cgi-bin/azerty.pl
80.8.55.1 - - [01/Jan/2001:12:00:10 +0100] "GET /pagel.html HTTP/1.0" 200 7009 "-" "Mozilla/4.7 [fr] (Win95; I)"
80.8.55.1 - - [01/Jan/2001:12:00:20 +0100] "GET /page2.cgi HTTP/1.0" 200 7009 "-" "Mozilla/4.7 [fr] (Win95; I)"
80.8.55.1 - - [01/Jan/2001:12:00:25 +0100] "GET /page3 HTTP/1.0" 200 7009 "-" "Mozilla/4.7 [fr] (Win95; I)"
80.8.55.1 - - [01/Jan/2001:12:00:30 +0100] "GET /image.gif HTTP/1.0" 200 7009 "-" "Mozilla/4.7 [fr] (Win95; I)"
80.8.55.1 - - [01/Jan/2001:12:00:35 +0100] "GET /image2.png HTTP/1.0" 200 7009 "-" "Mozilla/4.7 [fr] (Win95; I)"
80.8.55.1 - - [01/Jan/2001:12:00:40 +0100] "GET /js/awstats_misc_tracker.js HTTP/1.1" 200 4998 "-" "Mozilla/4.7 [
80.8.55.1 - - [01/Jan/2001:12:00:45 +0100] "GET /js/awstats_misc_tracker.js?screen=1024x768y73036758306r9417d1073

80.8.55.2 - - [01/Jan/2001:12:01:00 +0100] "GET /hitfromotherssitetoimage.gif HTTP/1.0" 200 7009 "-" "Mozilla/5.0+

80.8.55.3 - - [01/Jan/2001:12:01:10 +0100] "GET / HTTP/1.0" 200 7009 "http://www.sitereferer:81/cgi-bin/azerty.pl
80.8.55.3 - - [01/Jan/2001:12:01:15 +0100] "GET /pagel.html HTTP/1.0" 200 7009 "-" "Mozilla/5.0 (Windows; U; Wind
80.8.55.3 - - [01/Jan/2001:12:01:20 +0100] "GET /page2.cgi?x=ailx=x HTTP/1.0" 200 7009 "-" "Mozilla/5.0 (Windows;
80.8.55.3 - - [01/Jan/2001:12:01:25 +0100] "GET /page3 HTTP/1.0" 200 7009 "-" "Mozilla/5.0 (Windows; U; Windows N
80.8.55.3 - - [01/Jan/2001:12:01:30 +0100] "GET /image.gif HTTP/1.0" 200 7009 "-" "Mozilla/5.0 (Windows; U; Windo
80.8.55.3 - - [01/Jan/2001:12:01:35 +0100] "GET /image2.png HTTP/1.0" 200 7009 "-" "Mozilla/5.0 (Windows; U; Wind

80.8.55.4 - - [01/Jan/2001:12:01:45 +0100] "GET /samevisitorthan80.8.55.3usingaolproxy.gif HTTP/1.0" 200 7009 "-"

80.8.55.5 - - [01/Jan/2001:12:02:00 +0200] "GET /robots.txt HTTP/1.0" 200 299 "-" "This is an unknwn user agent"
80.8.55.5 - - [01/Jan/2001:12:02:00 +0200] "GET /mydir/robots.txt HTTP/1.0" 200 299 "-" "This is an unknwn user a
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot1.html HTTP/1.0" 200 7009 "-" "GoogleBot"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot2a.html HTTP/1.0" 200 7009 "-" "This is bot xxx"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot2b.html HTTP/1.0" 200 7009 "-" "This_is_bot_xxx"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot3.html HTTP/1.0" 200 7009 "-" "This is sucker xxx"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot4.html HTTP/1.0" 200 7009 "-" "woozweb-monitoring"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot5.html HTTP/1.0" 200 7009 "-" "wget"
80.8.55.5 - - [01/Jan/2001:12:02:05 +0200] "GET /pagefromabot6.html HTTP/1.0" 200 7009 "-" "libwww"

80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /cgi-bin/order.cgi?x=aId=998HTTP/1.0" 200 7009 "http://www.goo
80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /images/image1.gif HTTP/1.0" 200 364 "http://www.google.fr/sea
80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /images/image2.gif HTTP/1.0" 200 364 "http://www.google.fr/sea
80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /images/image3.gif HTTP/1.0" 200 364 "http://www.google.fr/sea
80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /images/image4.gif HTTP/1.0" 200 364 "http://www.google.fr/sea
80.8.55.6 - john [01/Jan/2001:13:00:00 +0100] "GET /images/image5.gif HTTP/1.0" 200 364 "http://www.google.fr/sea

80.8.55.7 - John Begood [01/Jan/2001:13:01:00 +0100] "GET /cgi-bin/order.cgi;family=f=999HTTP/1.0" 200 7009 "http
80.8.55.7 - John Begood [01/Jan/2001:13:01:00 +0100] "GET /do/Show;jsessionid=6BEF030AB1677BEC333FFCC7BF4DF564?pa

80.8.55.8 - - [01/Jan/2001:14:01:20 +0100] "GET /404notfoundpage.html?paramnotpagefound=valparamnotpagefound HTTP
80.8.55.8 - - [01/Jan/2001:14:01:20 +0100] "GET /405error.html HTTP/1.0" 405 0 "http://refererto405error/pagesfro

80.8.55.9 - - [01/Jan/2001:15:00:00 +0200] "GET /default.ida?XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX HTTP/1.0" 40
80.8.55.9 - - [01/Jan/2001:15:00:00 +0200] "SEARCH / -" 411 - "-" "-"

80.8.55.10 - - [01/Jan/2001:16:00:00 -0300] "GET / HTTP/1.1" 200 70476 "http://us.f109.mail.yahoo.com/ym/ShowLett
80.8.55.10 - - [01/Jan/2001:16:00:00 -0300] "GET /index.html HTTP/1.1" 200 70476 "http://us.f109.mail.yahoo.com/y
80.8.55.10 - - [01/Jan/2001:16:00:00 -0300] "GET /index.php HTTP/1.1" 200 70476 "http://us.f109.mail.yahoo.com/ym
80.8.55.10 - - [01/Jan/2001:16:00:00 -0300] "GET / HTTP/1.1" 200 70476 "http://us.f109.mail.yahoo.com/ym/ShowLett

80.8.55.10 - - [01/Jan/2001:16:00:00 -0300] "GET / HTTP/1.1" 200 70476 "http://us.f109.mail.yahoo.com/ym/ShowLett
80.8.55.10 - - [01/Jan/2001:16:30:00 -0300] "GET /pagel.html HTTP/1.1" 200 70476 "http://www.freeweb.hu/icecat/fi
80.8.55.10 - - [01/Jan/2001:17:00:00 -0300] "GET /cgi-bin/awredir.pl?url=http://xxx.com/aa.html HTTP/1.1" 302 704

```

This is what you should find:
6 true human visits

5 different true visitors

1 bot visit

1 worm attack

The entry pages for true visits should be "/" (even for 80.8.55.1) or "/cgi-bin/order.cgi" but nothing else.

Note: I did not find any commercial log analyzer that can deal such a common log file correctly, so if you find, let me know !

FAQ-COM300 : DIFFERENCE BETWEEN LOCAL HOURS AND AWSTATS REPORTED HOURS

PROBLEM:

I use IIS and there's a difference between local hour and AWStats reported hour. For example I made a hit on a page at 4:00 and AWStats report I hit it at 2:00.

SOLUTION:

This is not a problem of time in your local client host. AWStats use only time reported in logs by your server and all time are related to server hour. The problem is that IIS in some foreign versions puts GMT time in its log file (and not local time). So, you have also GMT time in your statistics.

You can wait that Microsoft change this in next IIS versions. However, Microsoft sheet Q271196 "IIS Log File Entries Have the Incorrect Date and Time Stamp" says:

The selected log file format is the W3C Extended Log File Format. The extended log file format is defined in the W3C Working Draft WD-logfile-960323 specification by Phillip M. Hallam-Baker and Brian Behlendorf. This document defines the Date and Time files to always be in GMT. This behavior is by design.

So this means this way of working might never be changed, so another chance is to use the AWStats plugin 'timezone'.

Warning, this plugin need the perl module Time::Local and it reduces seriously AWStats speed.

To enable the plugin, uncomment the following line in your config file

```
LoadPlugin="timezone TZ"
```

where TZ is value of your signed timezone (+2 for Paris, -8 for ...)

FAQ-COM320 : WHAT DOES "EU" (EUROPEAN COUNTRY)" MEAN IN GEOIP COUNTRY REPORTS ?

PROBLEM:

I use the [AWStats Geolp country plugins](#) to report countries according to geolocalisation of IP address. In country report, I have some visitors said to come from "eu (European country)". If this visitor come from France, does this means it is reported twice ?

SOLUTION:

No.

"eu (European country)" means we are sure that visitor come from an european country but we can't tell wich one. It might be a major country like France, Great Britain, Spain, Germany... like a very small one.

So, in the country report, each visitor is in one and only one group. If reported in "eu (European country)", it is not counted in another country, and if reported in a particular country, it is not counted in "eu (European country)", even if this country is in Europe.

FAQ-COM350 : HOW CAN I PROCESS AN OLD LOG FILE ?

PROBLEM:

I want to process an old log file to include its data in my AWStats reports.

SOLUTION:

You must change your [LogFile](#) parameter to point to the old log file and run the update (or use the -LogFile option on command line to overwrite [LogFile](#) parameter). The update process can only accept files in chronological order for a particular month, so if you have already processed a recent file and forgot to run update on a log file that contains older data, you must reset all of your statistics (see [FAQ-COM500](#)) and restart all of the update processes for all past log files and in chronological order.

However, there is a "tip" that allows you to rebuild only the month were you missed data:

Imagine we are on 5th of July 2003, all your statistics are up to date except for the 10th of April 2003 (you forgot to run the update process for this day, so there is no visit for this day). You can :

- Reset the statistics for April only (this means remove the file `awstats042003.[config].txt` as explained in [FAQ-COM500](#)),
- Move the statistics history files for the month after April (file `awstats052003.[config].txt`, `awstats062003.[config].txt`,...) into a temp directory (so that it is no longer in the DirData directory; as if they were deleted).

- Run the update process on all log files for April (in chronological order). AWStats does not complain about "too old record" because there is no history files in DirData directory that contains compiled data more recent than records into log you process.

- Moved back the month history files you saved into your DirData directory.

Your statistics are up to date and the missing days are no longer missing.

FAQ-COM360 : HOW CAN I PROCESS SEVERAL LOG FILES IN ONE RUN ?

PROBLEM:

How can I update my statistics for several log file, in one run ?

SOLUTION:

A solution should be to setup your config file with something like:

`LogFile=mylog*.log`

However, with such a syntax, AWStats can't know in which order processing log files (which log file is the first, next or last). So to work like this you must use the following syntax:

```
LogFile="/path/to/logresolve/merge.pl mylog*.log |"
```

[Logresolve/merge](#) is a tool provided with AWStats (in tools directory) that merges several log files on the fly. It opens a pointer on each file and sends, line by line, the oldest record from this. Using such a tool as a pipe source for AWStats [LogFile](#) parameter is a very good solution because, it allows you to merge log files whatever their size with no memory use, no hard disk use (no temporary files built), it is fast, it prevents you from a bad order if your log files are not correctly ordered, etc...

This tool can also be used to process log files from load balanced systems (see [FAQ-COM400](#))

FAQ-COM400 : HOW CAN I UPDATE MY STATISTICS WHEN I USE A LOAD BALANCING SYSTEM THAT SPLITS MY LOGS ?

PROBLEM:

How can I update my statistics when I use a load balancing system that splits my logs ?

SOLUTION:

First solution is to merge all split log files resulted from all your load balanced servers into one. For this, you can use the [logresolve/merge](#) tool provided with AWStats :

```
logresolve/merge.pl file1.log file2.log ... fileN.log > newfiletoprocess.log
```

And setup the [LogFile](#) parameter in your config file to process the *newfiletoprocess.log* file or use the `-LogFile` command line option to overwrite [LogFile](#) value.

As an other solution, if you miss disk space, or to save time, you can ask [logresolve/merge](#) to merge log files on the fly during the AWStats update process. For this, you can use the following syntax in your AWStats config file:

```
LogFile="/path/to/logresolve/merge.pl file*.log |"
```

See also [FAQ-COM360](#) for explanation on [logresolve/merge](#) use.

FAQ-COM500 : HOW CAN I RESET ALL MY STATISTICS ?

PROBLEM:

I want to reset all my statistics to restart the update process from the beginning.

SOLUTION:

All analyzed data are stored by AWStats in history files called *awstatsMMYYYY.[config].txt* (one file each month). You will find those files in directory defined by [DirData](#) parameter (same directory than *awstats.pl* by default).

To reset all your statistics, just delete all files *awstatsMMYYYY.txt*

To reset all your statistics built for a particular config file, just delete all files *awstatsMMYYYY.myconfig.txt*

Warning, if you delete those data files, you won't be able to recover your stats back, unless you kept old log files somewhere. You will have to process all past log files (in chronological order) to get your statistics back.

FAQ-COM600 : HOW CAN I COMPILE AND BUILD STATISTICS ON A DAILY BASIS ONLY ?

PROBLEM:

How can I compile and build statistics on a daily basis. I mean I want to have a full report with all charts with data for a particular day only and want one report for each day of month.

SOLUTION:

If you use version 6.5 or higher:

To build statistics:

What you can do is rerun the update process by adding the parameter `-databasebreak=hour` or `-databasebreak=day`.

Providing no option is similar than using `-databasebreak=month`, the default and old behaviour of AWStats.

Using this hidden option will ask AWStats to build a different database file for each break entity, this means that several reports are done for each hour or day, depending on option used.

To read a report:

Add same option `-databasebreak=hour` or `-databasebreak=day` with `-output` option when AWStats report is statically built from command line, or add `&databasebreak=hour` or `&databasebreak=day` if AWStats is called as a CGI. Also, complete options month and year used to choose month and year of report with other option *day* (when *databasebreak* option is 'day' or 'hour') and *hour* (only when *databasebreak* is 'hour')

So use `-day=XX` and/or `-hour=XX` when AWStats is run from command line. Use `&day=XX` and/or `&hour=XX` if AWStats is called as a CGI.

This feature is recent so may have results not completely reliable, that's why it is not yet fully documented.

If you use version 6.4 or older:

This is an non documented and not supported trick, as this is not the standard way of working:

First, run the update process at midnight (or on a log file that was rotated at midnight so that it contains only data for this particular day (you can choose another hour in night if you want to have days that "start" at an different hour).

Once the update process has been ran, MOVE (and not copy) the history file built by AWStats. For example on Unix like systems:

`mv mydirdata/awstatsMMYYYY.mydomain.txt mydirdate/awstatsDDMMYYYY.mydomain.txt`

Note that the name has been changed by adding the day. Repeat this each day after the update process.

With this you will have one history file for each day. You can then see full stats for a particular day by adding the non documented parameter `-day=DD` on command line (with others like `-month=MM` and `-year=YYYY`). If ran from a browser you can also add on URL.

However, if you have full day by day statistics, you don't have anymore statistics for full month, except if you create a second config file that whose history files would not be moved.

FAQ-COM700 : CAN I SAFELY REMOVE A LINE IN HISTORY FILES (awstatsMMYYYY*.txt) ?

PROBLEM:

After processing a log file I want to change my statistics without running AWStats update process but changing directly data in AWStats historical database files.

SOLUTION:

If you remove a lines starting with "BEGIN_" or "END_", AWStats will find your file "corrupted" so you must not change those two kinds of lines.

You can change, add or remove any line that is in any sections but if you do this, you must also update the MAP section (lines between BEGIN_MAP and END_MAP) because this section contains the offset in file of each other sections for direct I/O access. If history file is the last one, you can easily do that by removing completely the MAP section and run an update process. Like that AWStats will rewrite the history file and the MAP section will be rewritten (MAP section is not read by update process, only written). You do this at your own risk. The main risk is that some charts will report wrong values or be unavailable.

FAQ-SET050 : ERROR "MISSING \$ ON LOOP VARIABLE ..."

PROBLEM:

When I run awstats.pl from command line, I get:

"Missing \$ on loop variable at awstats.pl line xxx"

SOLUTION:

Problem is in your Perl interpreter. Try to install or reinstall a more recent/stable Perl interpreter.

You can get new Perl version at [ActivePerl \(Win32\)](#) or [Perl.com \(Unix/Linux/Other\)](#).

FAQ-SET100 : I SEE PERL SCRIPT'S SOURCE INSTEAD OF ITS EXECUTION

PROBLEM:

When I try to execute the Perl script through the web server, I see the Perl script's source instead of the HTML result page of its execution !

SOLUTION:

This is not a problem of AWStats but a problem in your web server setup. awstats.pl file must be in a directory defined in your web server to be a "cgi" directory, this means, a directory configured in your web server to contain "executable" files and not to documents files. You have to read your web server manual to know how to setup a directory to be an "executable cgi" directory (With IIS, you have some checkbox to check in directory properties, with Apache you have to use the "ExecCGI" option in the directory "Directive").

FAQ-SET150 : INTERNAL ERROR 500 IN MY BROWSER

FAQ-SET200 : ERROR "... COULDN'T CREATE/SPAWN CHILD PROCESS..."

PROBLEM:

AWStats seems to run fine at the command prompt but when ran as a CGI from a browser, I get an *"Internal Error 500"*.

I also also might have the following message in my Apache error log file (or in browser with Apache 2.0+):

...couldn't create/spawn child process: c:/mywebroot/cgi-bin/awstats.pl

SOLUTION:

First, try to run awstats.pl from command line to see if file is correct. If you get some syntax errors and use a Unix like OS, check if your file is a Unix like text file (This means each line end with a LF char and not a CR+LF char).

If awstats.pl file runs correctly from command line, this is probably because your web server is not able to known how to run perl scripts. This problem can occur with Apache web servers with no internal Perl interpreter (mod_perl not active). To solve this, you must tell Apache where is your external Perl interpreter.

For this, you have 2 solutions:

1) Add the following directive in your Apache **httpd.conf** config (or remove the # to uncomment it if line is already available)

ScriptInterpreterSource registry

Then restart Apache. This will tell Apache to look into the registry to find the program associated to .pl extension.

2) Other solution (not necessary if first solution works): Change the first line of awstats.pl file with the full path of your Perl interpreter.

Example with Windows OS and ActivePerl Perl interpreter (installed in C:\Program Files\ActiveState\ActivePerl), you must change the first line of awstats.pl file with:

FAQ-SET220 : CRASH WHILE RUNNING AWSTATS.PL OR PAGE CONTENT ONLY PARTIALLY LOADED ON WINDOWS XP

PROBLEM:

Sometimes my browser (Most often IE6) crash while running awstats.pl with some AWStats configuration. With some other versions or browsers, page content is partially loaded.

SOLUTION:

Problem was with WinXP and WinXPpro as documented at MS site Q317949;

"Socket Sharing Creates Data Loss When Listen and Accept Occur on Different Processes"

Result was that MSIE would crash or display nothing. Netscape and Opera handled the socket better but displayed the pages partially.

The effect of the bug was more pronounced as the page contents increased (above 30k).

<http://support.microsoft.com/default.aspx?scid=kb;EN-US;q317949>

And also at Apache.org

<http://www.apache.org/dist/httpd/binaries/win32/>

MS produced a Hotfix which is now included in SP1.

But the best solution is to use a better web browser. Take a look at [Firefox](#), one of the best and most popular web browser.

FAQ-SET270 : ONLY CORRUPTED OR DROPPED RECORDS

PROBLEM:

After running an AWStats update process, all my records are reported to be corrupted or dropped

SOLUTION:

First, if you have only a small percent of corrupted or dropped records, don't worry. This is a normal behaviour. Few corrupted or dropped records can appear in a log file because of internal web server bug, virus attack, error writing, log purge or rotate during a writing, etc...

However, if ALL your records are reported to be corrupted or dropped, check the following things:

If they are all dropped, run the update process from command line adding the option `-showdropped`

→ You will be able to know why a dropped record is discarded. In most cases, this is because you use a too large or bad filter parameter (SkipFiles, SkipHosts, OnlyFiles ...).

If they are all corrupted, run the update process from command line adding the option `-showcorrupted`

→ You will be able to know why a corrupted record is discarded.

If this is because of the log format, check the [FAQ-SET350](#) about log format errors.

If this is because the date of a record is said to be lower than date of previous, this means that you ran update processes on different log files without keeping the chronological order of log files.

If this is because the date is invalid, you might have a problem of date not computed correctly this it happens in some Pentium4/Xeon4 processors:

On some (few) Intel Pentium4 (also Xeon4) based host systems, log file time can not be computed correctly. This is not an issue of AWStats itself. This error usually occurs on source-based linux distributions (gentoo, slackware etc.), where all system libraries are compiled with CPU optimization. AWStats is a highly developed PERL application. PERL itself relies on some system libraries, for example GLIBC. The GLIBC library usually is buggy in this case. There is an easy way to figure out whether the problem described here is responsible for AWStats problems on your system:

If you have shell access to your machine, simply type the following command:

```
perl -e "print int ('541234567891011165415658')"
```

(NOTE: any 25-digit number works, there is no need to type this exact number)

If everything goes fine, you should see a floating point number as output:

```
5.41234567891011e+23
```

In this case, please do more research on your log file formats. Your host system itself is not responsible for the error.

But if simply a "0" returns or some other error, this is an indication of your glibc being corrupt.

ATTENTION: The only solution in this case might be to recompile your GLIBC. This can be a quite tricky task. Please consult the documentation and F.A.Q.s of your linux distribution first!! (experts: first check your global compile flags, eg. `march=Pentium4`. Trying with other compile flags can solve problem quickly in some cases.

NOTE: In some cases, this error might occur "suddenly", even though AWStats was already running perfect already.

FAQ-SET280 : ERROR "NOT SAME NUMBER OF RECORDS OF..."

PROBLEM:

When I run AWStats from command line (or as a cgi from a browser), I get a message "Not same number of records of ...".

SOLUTION:

This means your AWStats reference database files (operating systems, browsers, robots...) are not correct. First try to update to last version. Then check in your disk that you have only ONE of those files. They should be in '**lib**' directory ('db' with 4.0) where awstats.pl is installed:

browsers.pm

browsers_phone.pm

domains.pm

operating_systems.pm

robots.pm
search_engines.pm
worms.pm
status_http.pm
status_smtp.pm

FAQ-SET300 : ERROR "COULDN'T OPEN FILE ..."

PROBLEM:

I have the following error:

"Couldn't open file /workingpath/awstatsmmyyyy.tmp.9999: Permission denied."

SOLUTION:

This error means that the web server didn't succeed in writing the working temporary file (file ended by .tmp.9999 where 9999 is a number) because of permissions problems.

First check that the directory */workingpath* has "Write" permission for user *nobody* (default user used by Apache on Linux systems) or user *IUSR_SERVERNAME* (default used user by IIS on NT).

With Unix, try with a path with no links.

With NT, you must check NTFS permissions ("Read/Write/Modify"), if your directory is on a NTFS partition.

With IIS, there is also a "Write" permission attribute, defined in directory properties in your IIS setup, that you must check.

With IIS, if a default cgi-bin directory was created during IIS install, try to put AWStats directly into this directory.

If this still fails, you can change the DirData parameter to say AWStats that you want to use another directory (A directory you are sure that the default user, used by web server process, can write into).

FAQ-SET320 : ERROR "MALFORMED UTF-8 CHARACTER (UNEXPECTED ...)"

PROBLEM:

When running AWStats from command line, I get one or several lines like this on my output:

Malformed UTF-8 character (unexpected non-continuation byte 0x6d, immediately after start byte 0xe4) at /www/cgi-bin/lib/xxx.pm line 999.

SOLUTION:

This problem appeared with RedHat 8 and Perl 5.8.

I don't know if RedHat provides a fix for this, but some users had reported that you can remove those harmless messages by changing your LANG environment variable, removing the ".UTF-8" at the end. For example, set *LANG="en_US"* instead of *LANG="en_US.UTF8"*

FAQ-SET350 : EMPTY OR NULL STATISTICS REPORTED

PROBLEM:

AWStats seems to work but I'm not getting any results. I get a statistics page that looks like I have no hits.

SOLUTION:

That's one of the most common problem you can get and there is 3 possible reasons :

1) Your log file format setup might be wrong.

If you use Apache web server

The best way of working is to use the "combined" log format (See the [Setup and Use](#) page to know the way to change your Apache server log from "common" log format into "combined"). Don't forget to stop Apache, reset your log file and restart Apache to make change into combined effective. Then you must setup your AWStats config file with value [LogFormat=1](#). If you want to use another format, read the next FAQ to have examples of LogFile value according to log files format.

If you use IIS server or Windows built-in web server

The Internet Information Server default W3C Extended Log Format will not work correctly with AWStats. To make it work correctly, start the IIS Snap-in, select the web site and look at its Properties. Choose W3C Extended Log Format, then Properties, then the Tab Extended Properties and uncheck everything under Extended Properties. Once they are all unchecked, check off the list given in the [Setup and Use](#) page ("With IIS Server" chapter).

You can also read the next FAQ to have examples of [LogFormat](#) value according to log files format.

2) You are viewing stats for a year or month when no hits was made on your server.

When you run awstats, the reports is by default for the current month/year.

If you want to see data for another month/year you must:

Add *-year=YYYY -month=MM* on command line when building the html report page from command line.

Use an URL like *http://myserver/cgi-bin/awstats.pl?config=xxxMM* if viewing stats with AWStats used as a CGI.

3) When you read your statistics, AWStats does not use the same config file than the one used for the update process.

Scan your disk for files that match *awstats.*conf* and remove all files that are not the config file(s) you need (awstats.conf files, if found, can be deleted. It is better to use a config file called awstats.mydomain.conf).

FAQ-SET360 : STATISTICS REPORTED EXCEPT FOR OS, BROWSERS, ROBOTS AND KEYWORDS/KEYPHRASES

PROBLEM:

AWStats seems to report my statistics however some charts, like robots, os', browsers, search engines, or keywords/keyphrases are empty.

SOLUTION:

If only robots, search engines or keywords/keyphrases are empty, this simply means your web site was not yet visited by any robots and noone found your site using a search engines (this happens particularly for Intranet which are not referenced on search engines). If all of them are empty or with only unknown values, even after several updates, this probably means that your logfile does not contains all informations, this happens with Apache when using the standard "common" log format instead of the standard "combined" log format.

You may also use LogFormat=4 into your AWStats config files instead of 1.

Read AWStats setup documentation to know how to setup your Apache Web server to report logs in a "combined" log format then set [LogFormat=1](#) into your AWStats config file.

FAQ-SET400 : PIPE REDIRECTION TO A FILE GIVE ME AN EMPTY FILE

PROBLEM:

I want to redirect awstats.pl output to a file with the following command :

```
>awstats.pl -config=... [other_options] > myfile.html
```

But myfile.html is empty (size is 0). If I remove the redirection, everything works correctly.

SOLUTION:

This is not an AWStats bug but a problem between Perl and Windows.

You can easily solve this running the following command instead:

```
perl awstats.pl -config=... [other_options] > myfile.html
```

FAQ-SET450 : NO PICTURES/GRAPHICS SHOWN

PROBLEM:

AWStats seems to work (all data and counters seem to be good) but I have no image shown.

SOLUTION:

With Apache web server, you might have troubles (no picture shown on stats page) if you use a directory called "icons" (because of Apache pre-defined "icons" alias directory), so use instead, for example, a directory called "icon" with no s at the end (Rename your directory physically and change the [Dirlcons](#) parameter in config file to reflect this change).

FAQ-SET700 : MY VISITS ARE DOUBLED FOR OLD MONTH I MIGRATED FROM 3.2 TO 5.X

PROBLEM:

After having migrated an old history file for a month, the number of visits for this month is doubled. So the number of "visits per visitor" is also doubled and "pages per visit" and "hits per visit" is divided by 2. All other data like "pages", "hits" and bandwidth are correct.

SOLUTION:

This problem occurs when migrating history files from 3.2 to 5.x.

To fix this you can use the following tip (warning, do this only after migrating from 3.2 to 5.x and if your visit value is doubled). The goal is to remove the line in history file that looks like this

```
YYYYMM00 999 999 999 999
```

where YYYY and MM are year and month of config file and 999 are numerical values.

So if your OS is Unix/Linux

```
grep -vE '[0-9]{6}00' oldhistoryfile > newhistoryfile  
mv newhistoryfile oldhistoryfile
```

And then run the migrate process again on the file.

If your OS is windows and got cygwin

You must follow same instructions than if OS is Unix/Linux BUT you must do this from a cygwin 'sh' shell and not from the DOS prompt (because the ^ is not understood by DOS).

And then run the migrate process again on the file.

In any other case (in fact works for every OS)

You must remove manually the line YYYYMM00 999 999 999 999 (must find one and only one such line) and then run the migrate process again on the file.

FAQ-SET750 : AWSTATS RUN OUT OF MEMORY DURING UPDATE PROCESS WITH CYGWIN PERL

PROBLEM:

When I run the update process on a large log file with cygwin Perl, AWStats run out of memory but I am sure that I have enough memory to run AWStats according to the 'memory' column in benchmark chart available in AWStats documentation ([benchmark page](#)).

SOLUTION:

It might be a limit inside Cygwin Perl. Try to increase the Cygwin parameter *heap_chunk_in_mb*.

FAQ-SET800 : AWSTATS SPEED/TIMEOUT PROBLEMS ?

PROBLEM:

When I analyze large log files, processing times are very important (Example: update process from a browser returns a timeout/internal error after a long wait). Is there a setup or things to do to avoid this and increase speed ?

SOLUTION:

You really need to understand how a log analyzer works to have good speed. There is also major setup changes you can do to decrease your processing time.

See [important advices](#) in benchmark page.

FAQ-SEC100 : CAN AWSTATS BE USED TO MAKE CROSS SITE SCRIPTING ATTACKS ?

PROBLEM:

If a bad user use a browser to make a hit on an URL that include a < SCRIPT > ... < /SCRIPT > section in its parameter, when AWStats will show the links on the report page, does the script will be executed ?

SOLUTION:

No. AWStats use a filter to remove all scripts codes that was included in an URL to make a Cross Site Scripting Attack using a log analyzer report page.

FAQ-SEC150 : HOW CAN I PREVENT SOME USERS TO SEE STATISTICS OF OTHER USERS ?

PROBLEM:

I don't want a user xxx (having a site www.xxx.com) to see statistics of user yyy (having a site www.yyy.com). How can i setup AWStats for this ?

SOLUTION:

Take a look at the [security page](#).

FAQ-SEC200 : HOW TO MANAGE LOG FILES (AND STATISTICS) CORRUPTED BY 'WORMS' ATTACKS ?

PROBLEM:

My site is attacked by some worms viruses (like Nimba, Code Red...). This make my log file corrupted and full of 404 errors. So my statistics are also full of 404 errors. This make AWStats slower and my history files very large. Can I do something to avoid this ?

SOLUTION:

Yes.

'Worms' attacks are infected browsers, robots or server changed into web client that make hits on your site using a very long unknown URL like this one:

`/default.ida?XX%40%50...%40%50`

URL is generated by the infected robot and the purpose is to exploit a vulnerability of the web server (In most cases, only IIS is vulnerable). With such attacks, you will will always find a 'common string' in those URLs. For example, with Code Red worm, there is always default.ida in the URL string. Some other worms send URLs with cmd.exe in it.

With 6.0 version and higher, you can set the [LevelForWormsDetection](#) parameter to "2" and [ShowWormsStats](#) to "HBL" in config file to enable the worm filtering and reporting.

However, this feature reduce seriously AWStats speed and the worms database (lib/worms.pm file) can't contain all worms signatures. So if you still have rubbish hits, you can modify the worms.pm file yourself or edit your config file to add in the [SkipFiles](#) parameter some values to discard the not required records, using a regex syntax like example :

`SkipFiles="REGEX[^\default\.ida] REGEX[\\winnt\\system32\\cmd\\.exe]"`



AWStats License / Copyright

AWStats is distributed under the [GNU General Public License \(GPL\)](#).

So you must follow the line "Free software – Copylefted – GPL" to know what are major license agreements with AWStats.

Licence family			License name	Download, and use software at no charge	Sources available	Use, Modify or Distribute for free or just for a fee for copy.
Liberty policy ⁽¹⁾	Copyleft ⁽²⁾	Copyright				
Free	Non Copylefted	Non Copyrighted	Public Domain software	You can ⁽⁵⁾	Yes	You can ⁽⁵⁾
			MIT License	You can	Not always	You can
			BSD Licence	You can	Not always	You can ⁽⁷⁾
	Copylefted	Copyrighted to author	Apache License	You can	Not always	You can ⁽⁸⁾
			LGPL	You can	Yes	You can ⁽³⁾
			GPL	You can	Yes	You can ⁽³⁾
Semi Free Software		Copyrighted to author	Semi-Free Software	Yes if you are individual and for non profit purpose only	No	You can't ⁽⁴⁾
Proprietary Software		Copyrighted to author	Freeware (price=0)	You can	No	You can't ⁽⁴⁾
			Shareware (price>0)	Only for evaluation purpose	No	You can't ⁽⁴⁾
			Commercial licence	You can't ⁽⁴⁾	No	You can't ⁽⁴⁾

Licence chart updated on 2002-09-21

(1) Don't forget that "free" software refers to "liberty" to use and distribute it. So don't use "free software" for a program which price is null.

(2) Copyleft means that changes and distributions can be made with no additional restrictions. So a gratis software must be kept gratis.

(3) Sources must be provided

(4) You can if author give its authorization

(5) Everyone who modify a non copylefted and non copyrighted software can use its own licence for his version. So some modified/distributed version

(6) Your modified version can become a Proprietary Software if you want

(7) Name of authors/contributors can't be used to endorse or promote products derived from the software.

(8) A modified version can't be called with its original name. Name of authors/contributors can't be used to endorse or promote products derived from

(9) You can link a LGPLed library into a commercial program but you must allow users to use another version of this library.



Plugin Development

AWStats has a very flexible plugin architecture that is easy to use and allows for powerful extensibility. Here is the information you need to get started rolling your own. In this documentation, the terms plugin and module will be used interchangeably.

- [Plugin Files, Location](#)
- [Hooks](#)
- [Required Variables](#)
- [Accessible Variables](#)
- [Accessible Functions](#)
- [Available Hooks](#)

Plugin Files, Location

AWStats plugins are implemented as Perl modules with a file extension of .pm. Every time you run AWStats, either in update mode or HTML output mode, the configuration file will be parsed for the names of plugins to load. Then AWStats will scan the plugins folder for matching modules and load them into memory, executing hooks at the appropriate time during a run. Thus, when you create a plugin, you have to store the file in the plugins directory under the folder where awstats.pl resides.

Hooks

In order to be useful, your plugin must implement any number of different "hooks" that will be called at various points during the AWStats run. A hook is simply a Perl sub routine that will receive various parameters, perform whatever actions you desire such as calculations, modifications or output, and optionally return a value.

Note: all plugins MUST implement the Init_ hook to initialize the module and determine if the plugin will run under the current version of AWStats.

For information on the available hooks, [view the Hooks document](#).

Required Variables

Each plugin must implement three required, local variables including the name, hooks, implements and required AWStats Version. Typically you implement these at the top of your plugin file as in this example code:

```
#-----  
# PLUGIN VARIABLES  
#-----  
# <-----  
# ENTER HERE THE MINIMUM AWSTATS VERSION REQUIRED BY YOUR PLUGIN  
# AND THE NAME OF ALL FUNCTIONS THE PLUGIN MANAGE.  
my $PluginNeedAWStatsVersion="5.5";  
my $PluginHooksFunctions="GetCountryCodeByAddr GetCountryCodeByName ShowInfoHost";  
my $PluginName = "geoipfree";  
my $PluginImplements = "mou";  
# ----->
```

The \$PluginNeedAWStatsVersion indicates the minimum version of AWStats that your plugin requires to run properly. If a user attempts to implement your plugin with an older version of the program, the plugin will not load.

\$PluginHooksFunctions is a space delimited list of the different hooks that your plugin will implement. This list should only include names defined in the [hooks list](#). You should not list any private module functions or the Init_ hook in this list. The naming convention for all hooks is HookName_PluginName. The hooks like only includes the hook name without the underscore.

\$PluginName is simply the name of your plugin, exactly as it appears in the hooks and file name. This will be used by AWStats on load.

\$PluginImplements is a list of letter codes mapped to operations that your plugin performs. Without at least one of these

letter codes, your plugin will never run. The codes are:

- "m" – a Menu Handler plugin that provides links to navigate around reports. The plugin will be called any time a menu is displayed, such as in the left frame in cgi mode or top navigation in static mode.
- "o" – an Output plugin that will be loaded when AWStats is generating a report via dynamic CGI or static HTML
- "u" – an Update plugin that will process data and is run when AWStats is parsing log files and updating the history data files.

Accessible Variables

Your plugin has access to all of the global variables declared at the top of the AWStats.pl program. While you can write to these variables, it's best to only read them as another plugin may make unexpected modifications. However you can declare global variables within your own plugin and share those across other plugins. Just declare them inside the normal use vars qw/ ... / block within your own module.

Thus you can (and should) use settings from the configuration file and determine the debug level.

Accessible Functions

Plugins have access to all of the functions declared in the main AWStats.pl file. For debugging and error handling, you should use the debug and error functions. Below are some common functions that plugins take advantage of (remember you don't have to re-invent the wheel):

debug("debug message", debug_level) – Writes the "debug message" to the standard output if the (integer) debug_level is lower or equal to that set by the user at runtime. The higher the debug level, the less important or more informational the message. After outputting the message, the program continues running.

error("error message") – Writes the "error message" to the standard output and halts program execution.

Format_Bytes(bytes) – Converts the incoming decimal value to Kilobytes, Megabytes, Gigabytes and so forth. So if you put in 1024.5 it will spit out "1 KB"

Format_Date(YYYYMMDDHHMMSS) – Converts the incoming timestamps to something like 30 Apr 2010 – 16:55

Format_Number(number) – Adds commas or a user defined character where appropriate to separate numbers for easier reading.



Plugin Graphs

Charts, graphs and maps can be generated by your plugin and make AWStats reports look nice. AWStats generates arrays of values, label names and titles so that your plugin only has to handle the final data and display it as you wish.

NOTE: Only one graphing plugin should be chosen by a user at any given time, thus your plugin should generate as many graphs as possible. If more than one graphing plugin is enabled at any given time, multiple graphs will appear in the output for each graph type.

Below is a list of variables passed to the graphin hook and how you should use them.

- Title
- Type
- Block Label
- Val Label
- Val Color
- Val Max
- Val Total
- Val Average
- Val Data

Title

This is simply the title of the graph as it should be displayed to the end user. This could be "Hosts (Top 5)" or "Monthly History" or anything else. Of course, you don't have to show the title unless you want to.

Type

The type is important as your graphing plugin should determine what type of graph to output depending on the type AWStats is requesting. For information on types and the data order, [check the Types and Data section](#).

Block Label

This is a label for each group of data, usually labels for the X axis such as the days of the week, months of the year, etc. Your plugin can use the scalar count of values in this array as the number of times a loop needs to iterate through an axis. For example:

```
foreach my $j (1.. (scalar @$blocklabel)) { ... }
```

Val Label

@vallabel contains a list of labels for each of the data sets such as "pages", "hits", etc. Use the scalar count of values when accessing the data matrix.

Val Color

This is an array of color values associated with each type of data as assigned by the configuration file or the defaults in AWStats. Each color is a web formatted RGB hex encoded value such as "FFFFFF" or "000000".

Val Max

This array contains the maximum individual numeric value for each of the types of data stored in the data array. The count is the same as the the color array.

Val Total

This array contains the total numeric value for each type of data stored in the data array. The count is the same as the the color array.

Val Average

This array contains the average numeric value for each type of data stored in the data array. The count is the same as the the color array.

Val Data

This array contains the actual data values for each type of data indexed by each data type for a block of data, then the next set of types for a block and so on. Thus if we were looking at the monthly graph, the data would be laid out thusly:

```
$valdata[0] --> January Unique Visitors  
$valdata[1] --> January Number of Visits  
$valdata[2] --> January Pages  
$valdata[3] --> January Hits  
$valdata[4] --> January Bandwidth  
$valdata[5] --> February Unique Visitors  
$valdata[6] --> February Number of Visits  
....
```

Graph Types and Data

Below is a list of the different graph types and the order in which data appears in the associated arrays.

- month – a graph of each month for a given year
 - ◆ Unique Visitors
 - ◆ Visits
 - ◆ Pages
 - ◆ Hits
 - ◆ Bandwidth
 - daysofmonth – each day of the month
 - ◆ Visits
 - ◆ Pages
 - ◆ Hits
 - ◆ Bandwidth
 - daysofweek – an averaged data set for each day of the week
 - ◆ Pages
 - ◆ Hits
 - ◆ Bandwidth
 - hours – an averaged data set for each hour of a day
 - ◆ Pages
 - ◆ Hits
 - ◆ Bandwidth
 - cluster – the top 5 clustered servers if the cluster option is enabled
 - ◆ Pages
 - filetypes – the top 5 file types reported on
 - ◆ Pages
 - httpstatus – the top 5 HTTP status codes
 - ◆ Pages
 - browsers – the top 5 browsers
 - ◆ Pages
 - oss – the top 5 operating systems
 - ◆ Pages
 - hosts – the top 5 hosts
 - ◆ Pages
 - countries_map – a list of all countries and their hit counts
 - ◆ Hits
-



Plugin Hooks

The following is a list of hooks available to plugin developers. At various steps in the reporting or parsing process, AWStats will scan the plugin list for any matching hooks and execute them. All hooks start with the name of the hook, an underscore and then the plugin name. So for example, if I was creating a plugin called "mypluginname" then the initialization hook would look like:

```
sub Init_mypluginname()
```

If you need a hook that isn't listed, please contact the developers through Source Forge and place a request.

- [Required Hooks](#)
- [Common Hooks](#)
- [Processing/Update Hooks](#)
- [Output/HTML Hooks](#)

Required Hooks

Init_pluginname

Type: All

Parameters: Any parameters passed from the configuration file

Called: After loading configuration file and parsing plugins. Before parsing or HTML output

Description: The initialization hook is used to load parameters from the configuration file and initialize variables used in your plugin before any other hooks are called. This is a good place to initialize any hash tables you need or open data files.

Note: Init_ must return a list of hooks that your plugin implements in order for it to be used by AWStats. For example:

```
sub Init_geopip_region_maxmind {  
my $checkversion=&Check_Plugin_Version($PluginNeedAWStatsVersion);  
return ($checkversion?$checkversion:"$PluginHooksFunctions");  
}
```

If your plugin will accept a number of parameters, insert a line similar to the following:

```
my $InitParams=shift;  
my ($mode,$datafile,$override)=split(/\s+/, $InitParams,3);
```

Be sure to perform validation on the local variables and change the number of splits to the number of parameters you expect to read.

Common Hooks

SectionInitHashArray_pluginname

Type: All

Parameters: None

Called: When AWStats InitHash is called, such as at startup, when purging a history file or when a new month begins during processing.

Description: Use to initialize or clear any hash variables local to your plugin

Processing/Update Hooks

SectionWriteHistory_pluginname

Type: Update

Parameters: \$xml, \$xmlbb, \$xmlbs, \$xmlbe, \$xmlrb, \$xmlrs, \$xmlre, \$xmleb, \$xmlee

Called: Whenever the history file is written in AWStats

Description: Lets your plugin write a section to the history file. Please note that if your plugin stores a lot of data, it could greatly increase the history file size and slow down loading and processing. In such a situation, you may want to use this as a trigger to write to a separate history file and use the SectionReadHistory_ hook as a trigger to read that data.

XML parameters are the pre-defined XML tags the history file is using

SectionReadHistory_pluginname

Type: Update, Output

Parameters: \$issectiontoload, \$readxml, \$xmlele, \$countlines

Called: Whenever the history file is loaded in AWStats

Description: Lets your plugin read a section that it had written to the history file. The history log is always loaded before parsing log entries and before outputting HTML.

\$issectiontoload – the name of the plugin section found in the history file. If this is true, then the plugin should process the field data

\$readxml – True if the history file is in XML format

\$xmlele – XML element opening tag

\$countlines – how many lines have been read from the history file so far

SectionProcessHostname_pluginname

Type: Update

Required: No

Parameters: Resolved hostname

Called: On each line of a log file IF the hostname was resolved

Description: Processes a resolved host name, i.e. www.google.com and is useful for counting hits to the host or looking up additional information.

SectionProcessIP_pluginname

Type: Update

Parameters: \$Host

Called: On each line of a log file IF DNS resolution is turned off or a host was not resolved

Description: Processes an IP address i.e. 192.168.1.1 and is useful for counting hits to the host or looking up additional information.

Output/HTML Hooks

AddHTMLMenuLink_pluginname

Type: Output

Parameters: \$categ, \$menu, \$menulink, \$menutext

Called: Each time a new menu category is printed in the navigation frame or section

Description: Inserts a menu item into the navigation area that will link back to the AddHTMLGraph_hook defined in your plugin. If that hook is undefined, the user will experience an error. Requires AddHTMLGraph_hook

\$categ – name of the main navigation category such as 'who' or 'hosts'

\$menu – passback where you assign the position your link should appear in the menu

\$menulink – passback declaring the type of link

\$menutext – passback of the text to be displayed to the user

AddHTMLGraph_pluginname

Type: Output

Parameters: None

Called: When the AddHTMLMenuLink_created link is clicked, after HTML headers are sent and before HTML footers are sent

Description: Used to generate a dedicated report page with information specific to your plugin such as the Hosts report or Regions page. You are only responsible for the HTML between the standard AWStats header and footer.

AddHTMLStyles_pluginname

Type: Output

Parameters: None

Called: Whenever HTML headers are printed IF the configuration file does not define a specific style sheet URL.

Description: Use this section to output style information to the HTML header. Only print the individual style definitions, not the beginning or ending style tags (i.e. <style></style>)

AddHTMLHeader_pluginname

Type: Output

Parameters: None

Called: When HTML headers are printed, just before the closing </head> tag

Description: If your plugin requires extra HTML header information such as java script includes or meta tags, print them within this hook.

AddHTMLFooter_pluginname

Type: Output

Parameters: None

Called: When HTML footers are printed

Description: Prints footer HTML code before the AWStats copyright and closing </html> tag. Useful if you need to output tracking code or banner ads

AddHTMLBodyHeader_pluginname

Type: Output

Parameters: None

Called: After the HTML header is printed

Description: Used to print information to the main HTML body before the banner or any other information is printed

AddHTMLMenuHeader_pluginname

Type: Output

Parameters: None

Called: After the HTML header and HTML body header

Description: Used to print a header for the navigation menu whether it's inline on the page or in the navigation frame.

AddHTMLMenuFooter_pluginname

Type: Output

Parameters: None

Called: After the HTML navigation menu is printed

Description: Used to print a footer for the navigation menu whether it's inline on the page or in the navigation frame.

AddHTMLContentHeader_pluginname

Type: Output

Parameters: None

Called: After the HTML navigation menu is printed

Description: Outputs after the navigation menu but before any of the content tables in the report pages.

TabHeadHTML_pluginname

Type: Output

Parameters: \$title

Called: When a new content table is beginning

Description: Lets your plugin override the default table header style. Generally not used.

\$title – title to display to the user

ShowInfoHost_pluginname

Type: Output

Parameters: \$host

Called: Each time a new host row is displayed, such as the main page host section or the full host list page.

Description: Used to display a table column with detailed information about the host. The incoming parameter can be one of three values:

'__title__' – literal value that indicates we're printing a column header

\$Resolved Hostname – name of the host

\$IP Address – an unresolved dotted decimal IP

ShowInfoUser_pluginname

Type: Output

Parameters: \$user

Called: Each time a new authenticated user row is displayed

Description: Used to display a table column with detailed information about the user. The incoming parameter can have one of two values:

'__title__' – literal value that indicates we're printing a column header

\$user – the authenticated user name

ShowInfoCluster_pluginname

Type: Output

Parameters: \$cluster

Called: Each time a new cluster row is displayed

Description: Used to display a table column with cluster information. The incoming parameter can have one of two values:

'__title__' – literal value that indicates we're printing a column header

\$cluster – the cluster name

ShowInfoURL_pluginname

Type: Output

Parameters: \$url

Called: Each time a new page URL or referrer URL row is displayed

Description: Used to display a table column with additional url information. The incoming parameter can have one of two values:

'__title__' – literal value that indicates we're printing a column header

url – the page or referrer URL

ShowInfoGraph_pluginname

Type: Output

Parameters: \$title, \$type, \$display, @blocklabel, @vallabel, @valcolor, @valmax, @valtotal, @valaverage, @valdata

Called: Any time a graph, chart or map is output to HTML

Description: Used to display charts or graphs. AWstats will generate arrays of data, colors and labels that your graphing plugin should be able to handle. For more information, [see the graph plugin page](#)

ShowPagesAddField_pluginname

Type: Output

Parameters: \$key

Called: Each time a new page URL row is displayed on detailed URL pages and main summary

Description: Used to display a table column with additional url information. The incoming parameter can have one of three values:

'title' – literal value that indicates we're printing a column header

\$key – URL key to lookup in your hash

" – empty literal indicated the final row. Useful if you want to show total or averages



Plugins, contribs and related programs

You will find here description for different kinds of ressource files :

- [AWStats plugins](#)
- [Other AWStats contribs/tools/package](#)
- [Other Non AWStats contribs/tools \(not related to AWStats but related to log analysis\)](#)
- [Documents on AWStats or related to log analysis](#)

AWStats plugins

Plugins are .pm files you can put in your AWStats plugins directory to add new features.

Note that you must enable the plugin by adding a new line in your config files to make it works (See [LoadPlugin](#) parameter).

Standard free plugins : [tooltips](#), [decodeutfkeys](#), [ipv6](#), [hashfiles](#), [userinfo](#), [hostinfo](#), [clusterinfo](#), [urlalias](#), [timehires](#), [timezone](#), [rawlog](#), [graphapplet](#), [graphgooglechartsapi](#) [geoipfree](#)

GeoIP Plugins (may require other licensed product to work) : [geoip](#), [geoip_city_maxmind](#), [geoip_asn_maxmind](#), [geoip_region_maxmind](#), [geoip_isp_maxmind](#), [geoip_org_maxmind](#),

List of standard plugins (Provided with AWStats) :

- Tooltips**
Required Modules: None
Parameters: None
Description: Add tooltips pop-up help boxes to HTML report pages.
Provided with AWStats (5.4+).
- DecodeUTFKeys**
Required Modules: Encode and URI::Escape
Parameters: None
Description: Allow AWStats to show correctly (in language charset) keywords/keyphrases strings even if they were UTF8 coded by the referer search engine.
Plugin provided with AWStats (6.0+).
- IPv6**
Required Modules: Net::IP and Net::DNS
Parameters: None
Description: This plugin gives AWStats capability to make reverse DNS lookup on IPv6 addresses.
Note: IPv6 addresses are currently incompatible with the GeoIP databases
Plugin provided with AWStats (5.5+)
- HashFiles**
Required Modules: Storable
Parameters: None
Description: AWStats DNS cache files are read/saved as native hash files. This improves cache file loading speed, especially for very large web sites.
Plugin provided with AWStats (5.1+)
- UserInfo**
Required Modules: None
Parameters: None
Description: Add text (Firstname, Lastname, Office Department, ...) to authenticated user reports for each login value. A text file called userinfo.myconfig.txt, with two fields (first is login, second is text to show, separated by a tab char) must be created in DirData directory.
Plugin provided with AWStats (5.5+)
- ClusterInfo**
Required Modules: None

Parameters: None

Description: Add text (for example a full hostname) to cluster reports for each cluster number. A text file called clusterinfo.myconfig.txt, with two fields (first is cluster number, second is text to show) separated by a tab character must be created in DirData directory.

Note: This plugin is useless if ShowClusterStats is set to 0 or if you don't use a personalized log format that contains %cluster tag.

Plugin provided with AWStats (6.2+)

HostInfo

Required Modules: Net::XWhois

Parameters: None

Description: Add a column into host chart with a link to open a popup window that shows info on host (like whois records).

Plugin provided with AWStats (6.0+)

UrlAliases

Required Modules: None

Parameters: None

Description: Add text (Page title, description...) to URL reports before URL value. A text file called urlalias.myconfig.txt, with two fields (first is URL, second is text to show, separated by a tab char) must be created in DirData directory.

You can build your urlalias file manually or use [urlaliasbuilder.pl](#) tool (provided with 5.4+)

Plugin provided with AWStats (5.2+)

TimeHiRes

Required Modules: Time::HiRes

Parameters: None

Description: Time reported by --showsteps option is in millisecond. For debug purpose.

Plugin provided with AWStats (5.1+)

TimeZone

Required Modules: Time::Local

Parameters: <offset or local timezone>

Description: Converts the timezone of log files before processing using the offset provided in the parameters. This is useful for some IIS users if logs are stored in GMT or for users who have servers in a different timezone. You must provide either the abbreviated name of the timezone you wish to convert to or the offset value in hours. For example

LoadPlugin="timezone +2" would add two hours to the log timestamps

LoadPlugin="timezone CET" would convert the log from GMT to CET

Note: This plugin reduces AWStats speed by about 40%!

Plugin provided with AWStats (5.1+)

Rawlog

Required Modules: None

Parameters: None

Description: This plugin adds a form in AWStats main page to allow users to see raw content of current log files. A display filter will also be displayed.

Plugin provided with AWStats (5.7+)

GraphApplet

Required Modules: None

Parameters: None

Description: Supported charts are built by a 3D graphic applet.

LoadPlugin="graphapplet" # EXPERIMENTAL FEATURE

Plugin provided with AWStats (6.0+)

GraphGoogleChartAPI

Required Modules: None

Parameters: None

Description: Replaces the standard charts with free Google API generated images in HTML reports. If country data is available and more than one country has hits, a map will be generated using Google Visualizations.

Note: The machine where reports are displayed must have Internet access for the charts to be generated. The only data sent to Google includes the statistic numbers, legend names and country names.

Plugin provided with AWStats (6.0+)

GeoIPfree

Required Modules: Geo::IPfree version 0.2+ (from Graciliano M.P.)

Parameters: None

Description: Country chart is built from an Internet IP–Country database. This plugin is useless for intranet only log files.

Note: You must choose between using this plugin (need Perl Geo::IPfree module, database is free but not up to date) or the GeoIP plugin (need Perl Geo::IP module from Maxmind, database is also free and up to date).

Note: Activestate provide a corrupted version of Geo::IPfree 0.2 Perl module, so install it from elsewhere (from www.cpan.org)

for example).

This plugin reduces AWStats speed by about 10% !

Plugin provided with AWStats (5.5+).

You can find Geo::IPfree Perl Module on :

– All OS: [CPAN site](#)

– RPM compatible OS (RedHat, Mandrake...): [Lafraia rpm](#)

– ActiveState: GeoIPfree module provided by Activestate is not working correctly so I removed the link. You can follow the following setup instead: Download from [CPAN site](#) and copy 'Geo' directory (found in 'lib') into AWStats plugins directory. This is a fast tip to install Geo-IPfree perl module to be used by AWStats geoipfree plugin.

Note: The Geo::IPfree database has not been updated in a long time, so using the [geoip](#) plugin (using the Geo::IP Perl module, now free for countries) is highly recommended.

You can use the following tool to test your geoipfree setup: [testgeoipfree.pl](#). Just put this file into your plugins directory and run it.

Maxmind GeoIP Plugin Notes

Maxmind is a company that sells geolocation technology that matches IP addresses to countries, regions, cities and ISPs. All of their databases are available for an initial fee and a monthly update subscription and include greater accuracy and a higher frequency of updating. Some databases are available for free but they are not updated as often and may not be as accurate.

Install:

First choose a database. When downloading a database for AWStats, please download the Binary Format version. Current databases AWStats supports are:

Commercial: [Country](#), [Region](#), [City](#), [Organization](#), [ISP](#)

Open Source [Country](#), [City](#), [ASN](#)

If you choose to purchase a commercial database for country information, you only need to purchase the one with the greatest level of detail you want to appear in your report. For example, if you only want country information, you can purchase just the Country database. If you want City detail, you do not need to purchase the Country and Region databases, just the City database. (Organization and ISP databases do not include country or region info so you must purchase those in addition to Country, Region or City if you desire more information)

Next, install the Geo::IP Perl module. Then choose a Maxmind API to access the database. Maxmind also has two Perl API's available for interacting with their databases. There is a C/Perl library that will run quickly and there is a pure Perl API that is a little slower but does not require any C dependencies. Choose [one of the APIs](#) and install it on your machine.

Parameters: Each GeoIP database can be loaded entirely into memory or lookups can scan through the file. When editing your AWStats configuration, you have to specify if you want to use memory or file lookups. Specify GEOIP_STANDARD for file lookups or GEOIP_MEMORY_CACHE for memory loading. Please note that the memory method may crash on some Linux distributions so if that happens, try the standard method. Also, using the C API with Activestate Perl may crash on Windows machines, so use the PurePerl module instead.

You may also override some databases by generating a comma separated text file with IP addresses and associated data. This is useful if you are reporting on an Intranet site, have a mixture of internal and external users or the data for a specific IP is incorrect in the database. You can use the included geoip_generator tool to create override files. (Overrides only available in AWStats version 6.96 and later)

Override Formats:

GeoIP – ip_address,"countrycode"

GeoIP City – ip_address,"countrycode","regioncode","cityname","postalcode",latitude,longitude,"metrocode","areacode"

GeoIP ASN – ip_address,"asn ispname"

GeoIP Region – ip_address,"countrycode","regioncode"

GeoIP Org – ip_address,"organisation"

GeoIP ISP – ip_address,"ISP"

Note: AWStats does not currently support IPv6 for GeoIP lookups as Maxmind only offers the country DB with IPv6 at this time. As Maxmind includes IPv6 support in more DBs, we will update AWStats.

Enabling GeoIP plugins will reduce the speed of AWStats processing.

GeoIP

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)

Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/pathto/geoip.dat] </pathto/override.txt>

Description: Country chart is built from an Internet IP-Country database.

Note: You must choose between using this plugin or the GeoIPfree plugin (need Perl Geo::IPfree module, database is free but not up to date).

Example LoadPlugin="geoip GEOIP_STANDARD /usr/local/geoip.dat /usr/local/geoip_override.txt"

Plugin provided with AWStats (5.2+)

GeoIP_City_Maxmind

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)
Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/path/to/GeoIPCity.dat] </path/to/override.txt>
Description: This plugin adds a chart of hits by cities (with country and regions informations for major countries).
By enabling this plugin, you will see a new link called 'Cities' on reports menu.
Example LoadPlugin="geoip_city_maxmind GEOIP_STANDARD /usr/local/geoip.dat /usr/local/GeoIPCity_override.txt"
Plugin provided with AWStats (6.2+)

GeoIP_ASN_Maxmind

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)
Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/path/to/GeoIPASN.dat] </path/to/override.txt> <link to AS lookup>
Description: This plugin adds a chart of AS numbers where the host IP address is registered. This plugin can display some ISP information if included in the database. You can also provide a link that will be used to lookup additional registration data. Put the link at the end of the parameter string and the report page will include the link with the full AS number at the end.
Example LoadPlugin="geoip_asn_maxmind GEOIP_STANDARD /usr/local/geoip.dat /usr/local/GeoIPASN_override.txt http://enc.com.au/itools/aut-num.php?autnum="
Plugin provided with AWStats (6.9+)

GeoIP_Region_Maxmind (Commercial Only)

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)
Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/path/to/GeoIPRegion.dat] </path/to/override.txt>
Description: This plugin adds a chart of hits by regions. Only regions for US and Canada can be detected at this time.
By enabling this plugin, you will see a new link called 'Regions' on reports menu. For the moment this link works only when AWStats reports are build as CGI (static built not yet supported).
Example LoadPlugin="geoip_region_maxmind GEOIP_STANDARD /usr/local/GeoIPRegion.dat /usr/local/geoip_override.txt"
Plugin provided with AWStats (6.2+)

GeoIP_ISP_Maxmind (Commercial Only)

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)
Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/path/to/GeoIPISP.dat] </path/to/override.txt>
Description: This plugin adds a chart of hits by ISP.
By enabling this plugin, you will see a new link called 'ISP' on reports menu. For the moment this link works only when AWStats reports are build as CGI (static built not yet supported).
Example LoadPlugin="geoip_isp_maxmind GEOIP_STANDARD /usr/local/GeoIPISP.dat /usr/local/geoip_override.txt"
Plugin provided with AWStats (6.3+)

GeoIP_Org_Maxmind (Commercial Only)

Required Modules: Geo::IP or Geo::IP::PurePerl (from Maxmind)
Parameters: [GEOIP_STANDARD | GEOIP_MEMORY_CACHE] [/path/to/GeoIPOrg.dat] </path/to/override.txt>
Description: This plugin adds a chart of hits by Organization names.
By enabling this plugin, you will see a new link called 'Organizations' on reports menu. For the moment this link works only when AWStats reports are build as CGI (static built not yet supported).
Example LoadPlugin="geoip_org_maxmind GEOIP_STANDARD /usr/local/GeoIPOrg.dat /usr/local/geoip_override.txt"
Plugin provided with AWStats (6.3+)

Other AWStats contribs/tools/package

All following files/products were developped and submitted by contributors. I haven't tried them, so I can't tell you if they work and how they work...
They can enhance AWStats use or save you time giving you a start for a new development.

Other tools:

Replacement Frontend:

[BetterAWStats](#) is a PHP Frontend to show you AWStats data file differently.
[AWStats PhpNuke module](#) is a module for the open source CMS PHP Nuke.
[Ravencore](#) is an Open Source Hosting Control Panel that include AWStats as part of its statistics solutions.
[summary4severaldomains.pl](#) is a small perl script that build an index.html page with AWStats txt summary information for each domains on same page (number of pages, hits and bandwidth for month).
[AWStats Totals](#) is a php page that use AWStats database to show a summary for several config files, on the same page.

Miscellaneous:

[A plugin for WordPress](#) that allow WordPress users to add tags required by AWStats to report the "miscellaneous chart".

[Exim2Awstats](#) is a tool to convert Exim MTA log file to a format readable by AWStats.

[Logprocess](#) automate the collection, merging, processing and archiving of IIS log files in a hosting (mult-domain) environment using AWStats to generate and manage stats/reports.

[XlencArIm](#) is a tool to enhance use of AWStats with IIS (Catching log files from FTP, doing GMT correction on log files, ...). Don't know if it works correctly but seems to be interesting.

[Configurador](#) is a piece of CGI perl scripts to allow you to edit an AWStats config file from the web. Code comments are in Spanish but it may be a good start to develop your own tool. *Hector García Alvarez*

[logtrans.py](#) is a python small example on how to convert an old log file from IIS4.0 to a new format compatible with AWStats. Don't know if it works, just here to provide you a code sample if you like python.

[isa2w3c.pl](#), is a convertor to have a more standard log file from an ISA log file.

[go_awstats](#), is a tool to automate build of AWStats statics reports for all config files but also for all month.

[Stats Letter](#), is a Perl tool to send AWStats reports to one or several user for one or several reports. Documentation in French only.

Other Non AWStats contribs/tools (not related to AWStats but related to log analysis)

[maillog2commonlog.pl](#) converts mail logs in smail or qmail format to the common log format. Note that if you want to analyze, with AWStats, mail log files from postfix, sendmail or qmail, a better solution is to use instead maillogconvert.pl preprocessor (See AWStats [F.A.Q](#) about analyzing mail logs).

Joey Hess, freeware.

[maillogconv.pl](#) is another mail log processor that can be used to change mail log files into a log file that AWStats can understand.

Note that a derivated and enhanced tool called maillogconvert.pl is provided with AWStats since 5.7. See [F.A.Q](#) about analyzing mail logs to know how to setup AWStats with maillogconv.pl or maillogconvert.pl to analyze Postfix, Sendmail, Qmail or MDAemon log files.

[mlc](#) is another postfix mail logs converter (written in C). Try it if the maillogconvert.pl tool provided with AWStats does not work on your postfix log file.

[access_referer_agent2combined.pl](#) is a perl script that converts a directory of common log format (CLF) access-files, referer-files and agent-files to a single combined.log This is usefull if you have separeate access/referer/agent logfiles and want to use all features of a loganalyzer that require a combined log file.

[anonlog](#) "anonymizes" your logfile by encoding sensitive information.

Stephen Turner, freeware.

If you don't like AWStats, you can try thoose other popular similar tools :

[Analog](#). A very old log analyzer (written in C).

Stephen Turner, free software.

[Webalizer](#). Another very old log analyzer (written in C).

There is also commercial products you can find doing a search on 'log file analyzer' with [Google](#).

Documents on AWStats or related to log analysis

AWStats official documentation for last version can be found here: [HTML On line](#) or [PDF file](#)

[Additional resources for AWStats](#) provided by Antezeta (or in Italian)

An intro and install doc 'Using AWStats on GNU/Linux or Windows' in [Chinese](#)

A setup tutorial in [Danish](#)

A setup tutorial in [French from Funix: How to install AWStats and Webalizer](#)

A setup tutorial in [French for Windows users](#)

A setup tutorial in [French from Fpoeserv](#)

A setup tutorial for debian users with a logrotate tool in [German](#)

Another way to explain [AWStats setup](#)



AWStats Webmin module

This page is designed to help users of the 'Webmin' administration tool (www.webmin.com) to manage their AWStats config/tools. If you don't use it, you don't need to read what is described here.

- Adding the AWStats Webmin module to Webmin interface
- Setup of the AWStats Webmin module
- Using the AWStats Webmin module

Adding the AWStats Webmin module to Webmin interface

- Run Webmin, go to "Webmin Configuration".
- Click on "Webmin Modules".
- Enter in the "Install module", the path of the `awstats-x.x.wbm` file (this file is provided with AWStats in `tools/webmin` directory or can be downloaded separately from AWStats web site) and click "Install module".
After that a new entry "AWStats LogFile Analysis" should appear in your Webmin menu.

Setup of the AWStats Webmin module

- Run Webmin, go to the "AWStats LogFile Analysis" menu.
- Click on the "Module configuration".
- You must now enter here values for:
 - Absolute filesystem path to AWStats (CLI)
It's the full path of the `awstats.pl` script. Path depends on package you use.
It could be `/usr/local/awstats/wwwroot/cgi-bin/awstats.pl`
 - Absolute or relative URL path to AWStats (CGI)
It's absolute or relative URL you must enter to run AWStats as a CGI. This depends on the way you setup your web server and cgi directories.
It could be `/cgi-bin/awstats.pl` or `http://127.0.0.1/awstats/awstats.pl`.
 - Sample AWStats configuration file
This is the sample file used to create a new config file.
In most cases it must be `/etc/awstats/awstats.model.conf`.

Using the AWStats Webmin module

- Just run Webmin, go to the "AWStats LogFile Analysis" menu.
 - You can now, add, edit or delete config files.
 - You can also update or view statistics for a particular config file.
-



AWStats Dolibarr ERP module

A lot of ERP and CRM softwares (software to manage a company like invoices, commercial proposal, products, agenda, contacts, stocks, point of sale...), exists now in OpenSource. On of the most famous and easy to ease is [Dolibarr ERP/CRM](http://www.dolibarr.org). So to offer a complete solution for companie's manager, I added a module to add AWStats reporting inside this software.

This is links to have more informations:

- * Link to dolibarr ERP/CRM project: <http://www.dolibarr.org>
 - * Link to doliStore where to download AWStats module for Dolibarr: <http://www.dolistore.com>
 - * Link to wiki documentation of module: http://wiki.dolibarr.org/index.php/Module_AWStats_En
-