

Zugriffe auswerten mit dem Webalizer

[Martin Sommer](#)

Allgemeines

Der [Webalizer](#) ist ein Open Source-Programm zur Darstellung der Zugriffsstatistiken auf eine Homepage. Er ist auf die unterschiedlichsten Plattformen portiert, z.B. Linux auf PC, Alpha und PPC, Solaris auf Sparc oder Windows. Die Auswertungsmöglichkeiten sind recht umfangreich und hängen davon ab, wie die Konfigurationsdatei des *Webalizers* und die Konfigurationsdatei des Webservers eingerichtet sind. Da der größte Teil der Linux-Benutzer sicherlich Apache als Webserver benutzt, werden nur die nötigen Einstellungen der *Apache*-Konfigurationsdatei `httpd.conf` beschrieben. Sie sind prinzipiell auf andere Webserver übertragbar.

Der *Webalizer* ist im Lieferumfang der SuSE 7.1 Professional enthalten und per *YaST* installierbar. Im Netz ist er über den [Downloadbereich](#) der Webalizer-Homepage oder auf einem der Mirrors, z.B. bei [mrunix.net](#) erhältlich. Dort stehen die Sourcen und sogar fertige Binaries für alle Plattformen zur Verfügung, die nur entpackt werden müssen.

Für eine Installation in Deutsch (und in anderen Sprachen) stellen z.B. die [DLR](#) oder die schwedische Firma [Chalmers](#) die Sourcen zum Download bereit.

Installation

Wer die Installation mittels der Source-Dateien vornimmt, muß diese auf normalem Wege mit der Befehlsfolge

```
./configure  
make  
make install
```

kompilieren. Die genaue Anleitung mit den Optionen, die bei `./configure` möglich sind, kann man in der einfachen [Installationsanleitung](#) der Webalizer-Homepage nachlesen.

Wer die Binaries entpacken will, sollte die gezippte Datei in ein eigenes Verzeichnis legen und sie dort entpacken. Danach muß nur die Programmdatei `webalizer` in ein `bin`-Verzeichnis kopiert werden. Die Konfigurationsdatei kann im Webalizer-Verzeichnis oder in `/etc` liegen. Ein gutes Manual verbirgt sich in der Datei `webalizer.1`.

Benutzung

Die folgenden Angaben beziehen sich auf die Version 2.01 des *Webalizers*. Der *Webalizer* wertet Logfiles aus. Gewöhnlich will man das Haupt-Logfile der Zugriffe auf die Homepage mit allen Unterseiten auswerten. Es heißt standardmäßig `access_log` und steht im Verzeichnis `/var/log/httpd`. Ausgewertet werden verschiedene Parameter, die durch den Webserver mitgeloggt werden und in `access_log` gespeichert werden. Die wichtigsten davon sind:

- die IP-Adresse des Nutzers
- Datum und Uhrzeit der Zugriffe

- jede Datei, die vom Nutzer geladen wird

Apache-Konfiguration

Wer mehr benutzerbezogene Informationen haben möchte, muß dies in der Konfigurationsdatei des *Apache* Webservers `/etc/httpd/httpd.conf` einstellen. Dort sind vor allem zwei Optionen von Interesse: der "Referrer" und der "Agent", also die Seite, von der der Benutzer jeweils herkam und das Computersystem des Benutzers inklusive Betriebssystem und Browser. Ein Nachteil davon ist, daß die Logfiles, die bei oft besuchten Seiten sowieso schon erhebliche Größen haben, noch weiter wachsen.

Dazu ist in der `httpd.conf` lediglich die Zeile

```
# CustomLog /var/log/httpd/access_log combined
```

zu aktivieren, indem man das Kommentarzeichen (#) entfernt. Alternativ können noch die Agents und die Referrer in eigene Dateien geloggt werden (das Verzeichnis nimmt dann noch schneller an Größe zu). Dazu muß man die beiden Zeilen darüber:

```
# CustomLog /var/log/httpd/referer_log referer
# CustomLog /var/log/httpd/agent_log agent
```

aktivieren.

Konfiguration des Webalizers

Vorbemerkung: Prinzipiell ist eine Konfigurationsdatei nicht notwendig. Dann müssen jedoch alle Einstellungen, die nicht Default sind, von Hand dem Startbefehl hinzugefügt werden. Daher ist es natürlich wesentlich komfortabler, alle Einstellungen, die das Programm benutzen soll, in der Konfigurationsdatei festzulegen. Nach Installation des *Webalizers* enthält das Verzeichnis eine Standard-Konfigurationsdatei, in der viele sinnvolle Optionen voreingestellt sind. Andere sind als Beispiele angegeben, aber auskommentiert, so daß man sie bei Bedarf nur aktivieren muß, ohne sich um eine falsche Syntax kümmern zu müssen. Für den größten Teil der Optionen existiert eine Default-Einstellung, so daß prinzipiell kein Eintrag in der `.conf`-Datei nötig ist.

Ansprache der Konfigurationsdatei

Um den Rahmen des Artikel nicht zu sprengen, werden hier nur die wichtigsten Optionen der Konfigurationsdatei besprochen. Die Datei heißt standardmäßig `webalizer.conf` und sollte, damit sie beim Start des *Webalizers* ohne Pfadangabe gefunden wird, am besten in `/etc/` stehen. Um sie zu benutzen, wird der Durchlauf dann einfach mit `webalizer` gestartet. Benutzt man verschiedene Konfigurationsdateien für verschiedene Aufgaben, so muß außer bei Benutzung von `/etc/webalizer.conf` als Konfigurationsdatei dem Programm stets der Pfad mit der Option `-c` mitgegeben werden. Bei Benutzung von z.B. der Datei `webalizer.test` aus dem Verzeichnis `/etc/` lautet der Programmaufruf:

```
webalizer -c /etc/webalizer.test
```

Logfile und Ausgabeverzeichnis

In der Konfigurationsdatei **muß** man angegeben, welche Logdatei benutzt werden soll, d.h. es gibt hier kein Default, voreingestellt ist allerdings schon `/var/log/httpd/access_log`. Die Datei wird in folgende Zeile eingetragen:

LogFile /var/log/httpd/access_log

Es gibt mehrere Logfile-Formate, die benutzt werden können, das Standardformat heißt `clf`. Ebenso funktioniert der Durchlauf mit gezippten Logfiles im `gz`-Format, was man vielleicht nutzen möchte, weil man ab und zu große Logfiles packen will, um Plattenplatz zu sparen.

Ebenso sollten Sie das Verzeichnis angeben, in dem die Ergebnisse gespeichert werden sollen. Es steht in folgender Zeile:

OutputDir /usr/local/httpd/htdocs/webalizer

Es empfiehlt sich natürlich, eigene Verzeichnisse für die Ausgabe zu erstellen. Falls man mit verschiedenen Konfigurationsdateien verschiedene Jobs erledigt, sollte man natürlich auch in der jeweiligen `.conf`-Datei das jeweilige Ausgabeverzeichnis angeben, da sonst alte Daten überschrieben werden, bzw. je nach Einstellung neue an alte angehängt werden, die überhaupt nicht zu ihnen passen, da sie z.B. ein anderes Logfile durchsucht oder sonstige abweichende Einstellungen haben.

Da der *Webalizer* HTML-Ausgaben generiert, wollen Sie vielleicht Ihre Seiten im Browser unterhalb der Document Root (meist `/usr/local/httpd/htdocs/`) sehen, d.h. die Seiten liegen dann in einem Verzeichnis, das per `http` erreichbar ist, z.B. in `http://www.ihredomain.de/webalizer`. Hierbei müssen Sie natürlich bedenken, daß dann Ihre Statistik auch von außen abrufbar ist. Falls Sie das nicht wünschen, müssen Sie das Verzeichnis mit einem Paßwortschutz versehen oder es oberhalb der Document Root plazieren. Im letzten Fall läßt es sich allerdings nur als File in den Browser laden: `file:///Pfad_zum_Webalizerverzeichnis/index.html`. Bei Firmen sind die Daten womöglich betriebswirtschaftlich relevant und sollten natürlich nicht jedem zugänglich sein und zum besseren Schutz vor Hackern möglichst über der Document Root liegen.

Inkrementell oder nicht?

Nach den beiden sehr wichtigen Einträgen für das Logfile und das Ausgabeverzeichnis stehen Sie im weiteren Verlauf der Konfigurationsdatei vor der Frage `Incremental yes` oder `no`. Kurze Antwort: Viele Zugriffe: `yes`, wenige Zugriffe: `no`. Bei hoher Zugriffszahl ist es günstig, das `access_log`-file öfter zu zippen und neu beginnen zu lassen. Damit die vorigen Durchläufe dann trotzdem mitausgewertet werden, hat der *Webalizer* eine interne History, die mit `Incremental yes` aktiviert wird. Bei kleinen rein privaten Webseiten ist dies in der Regel nicht nötig.

Die Ausgabe

Nach der letzten Frage folgen viele eher unwichtige bzw. defaultmäßig richtig oder sinnvoll eingestellte Parameter wie der DNS Lookup, wo geregelt wird, welche Datenbank verwendet wird, um die IP-Adressen aufzulösen, d.h. sie in richtige Webadressen zu übersetzen. Viele Ausgabeparameter können eingestellt werden, die die Ausgabe des Textes betreffen (meist beginnend mit `HTML`). Wichtig ist hier nur die Angabe, welcher Dateityp als "page" gezählt werden soll und schließlich als "Page Impression" ausgegeben wird (Zeilen mit `PageType`). Voreingestellt sind hier `htm*` und `cgi`. Benutzt man `php` und/oder `Perl`, so sind die entsprechenden Zeilen einfach zu aktivieren bzw. bei anderen Formaten hinzuzufügen (z.B. `PageType asp`).

Interessant wird es dann erst wieder weiter unten, wo festgelegt wird, welche "Top Tables" in welcher Größe angezeigt werden. Allerdings kann auch hier getrost die Default-Einstellung genommen werden, aber man sollte damit spielen, um eine Ausgabe zu bekommen, die dem eigenen Geschmack entspricht. Der Agent und der Referrer, die hier angegeben werden können, werden wie oben besprochen nur ausgegeben, wenn sie in der *Apache*-Konfiguration aktiviert sind.

Falls nicht `index.html` als Standard-Startseite für Verzeichnisse verwendet wird, sondern z.B.

home.html, so ist dies im Abschnitt `IndexAlias` zu definieren.

Der nun folgende Abschnitt mit den `Hide-`, `Group-`, `Ignore-` und `Include-`Schlüsselwörtern ist wieder von größerer Wichtigkeit für eine vernünftige Auswertung der Zugriffe. In diesem Abschnitt kann man die Zugriffe z.B. von der eigenen Maschine, von anderen Rechnern des gleichen Netzwerks (z.B. alle Rechner der eigenen Firma) oder von ungeliebten Nutzern ausblenden oder sogar völlig ignorieren. Auf der anderen Seite kann man (z.B. für interne Zwecke) alle Nutzer ausblenden und nur explizit ganz bestimmte anzeigen lassen. Ausblenden kann (und sollte) man auch die Zugriffe auf die Bilder (oder bestimmte andere Dateitypen der Homepage, z.B. `txt` oder `tpl`), da sonst jeder Button als Hit gezählt wird. Wählt man das Schlüsselwort `Hide`, um bestimmte Angaben zu verstecken, werden die jeweiligen Zahlen in den Tabellen und den Graphen der "Top"-Statistiken ignoriert. Sie tauchen jedoch in den "Total"-Tabellen am Anfang der Webalizer-Ausgabe auf bzw. werden dort mitgezählt. Wählt man hingegen `Ignore`, werden diese Zugriffe völlig ignoriert, auch in den "Total"-Tabellen. Weiterhin stecken hier einige Gruppierungsfunktionen, mit denen man bestimmte Parameter gruppieren kann. Spielen mit der Gruppierungsfunktionen kann ggf. Ausgabe übersichtlicher machen.


Die Suchwortfunktion

Als letzter interessanter Abschnitt folgt kurz vor Ende der Datei die Festlegung der Suchwörter in den Suchmaschinen (Zeilen mit `SearchEngine`). Hier ist für einige wichtige Suchmaschinen schon voreingestellt, hinter welcher Variablen das Suchwort steckt. Damit kann ausgewertet werden, was in die Suchmaschinen eingegeben wurde, um die Seite zu finden. Allerdings wird bei der Ausgabe hier nicht zwischen den Suchmaschinen unterschieden, sondern nur eine Hitliste der Suchwörter aufgestellt. Dies ist hilfreich um z.B. das Metatag `keywords` der Homepage zu optimieren.

Wesentlich interessanter kann diese Funktion allerdings werden, wenn bei großen Webseiten, die eine eigene Suchfunktion bereitstellen, ausgewertet werden soll, nach was in der eigenen Site gesucht wurde. Dadurch kann die Homepage besser den Nutzerwünschen angepaßt werden, da ersichtlich ist, was die Besucher wissen wollen. Dazu ist es am besten, man deaktiviert zuerst die anderen Suchmaschinen, indem man sie hier mit `#` auskommentiert. Dann fügt man für das Beispiel der Open Source-Suchmaschine `htdig`, die auch im Portal benutzt wird, folgende Zeile hinzu:

```
SearchEngine htsearch words=
```

Man muß `htsearch` anstatt `htdig` angeben, da hier ein Substring aus der URL verlangt wird. Nach einer Suche mit `htdig` steht in der URL der string `htsearch` und nicht der String `htdig`. Der String `words=` wird hier eingegeben, da `words` der Name der Variablen ist, in der `htdig` die Suchbegriffe speichert.

Eine funktionierende Beispieldatei finden Sie [hier](#). In dieser Datei wurden zur besseren Übersicht alle Kommentare und überflüssigen bzw. defaultmäßig sinnvoll eingestellten Parameter entfernt. 

Linux auf dem Server 22.03.2001