University of Porto
Faculdade de Engenharia

Network Planning and Management PGRE

Lab work 1

Alena Tesařová
Rodrigo Caldas
Tiago Ribeiro

March 4, 2020

Contents

1 Introduction

The purpose of this assignment is to get familiar with web-based tools to analyze the use of HTTP server resources and proxies. We were purposed to use **Webalizer** and **AwStats** as a freeware tools for analysis from the server's log files. Both tools create HTTP service usage reports that will be described in section 4. As a HTTP server was used **Apache** (section 2) and Squid as a HTTP web proxy (section 3). At the end, a comparison with the **W3perl** tool will be made (section 4.4).

2 Apache

Apache HTTP server is an open source web server that's widely known and used. It has several features such as supporting server side programming languages (Perl, PHP and many others), popular authentication modules, allowing virtual hosts and configurable error messages. All the information concerning apache can be found on Apache official web pages[1]. [3, 4]

Our task was to create two virtual hosts to simulate hits to different sites. Then we had to put logs file to different files to be able to differentiate them using analysing tools. The configuration is described below.

Apache configuration of two Virtual Hosts

1. Go to Apache configuration location: */etc/apache2/sites-enabled/*.
2. Create 2 domains e.g. *domain.com.conf* and *domain2.com.conf* as described in Figure 1.

```
<VirtualHost *:85>

ServerAdmin webmaster@domain.com
ServerName domain.com
ServerAlias www.domain.com
DocumentRoot /var/www/domain.com/public_html

ErrorLog ${APACHE_LOG_DIR}/domain.com-error.log
CustomLog ${APACHE_LOG_DIR}/domain.com-access.log combined

</VirtualHost>
```

(a) Configuration of domain.com.conf

```
<VirtualHost *:8080>

ServerAdmin webmaster@domain2.com
ServerName domain2.com
ServerAlias www.domain2.com
DocumentRoot /var/www/domain2.com/public_html

ErrorLog ${APACHE_LOG_DIR}/domain2.com-error.log
CustomLog ${APACHE_LOG_DIR}/domain2.com-access.log combined

</VirtualHost>
```

(b) Configuration of domain2.com.conf

Figure 1: Apache configuration of domains

3. Find Apache configuration and add listeners to these new ports (*sudo vim /etc/apache2/ports.conf*).
4. Open the browser and look at the final result (in Figure 2).

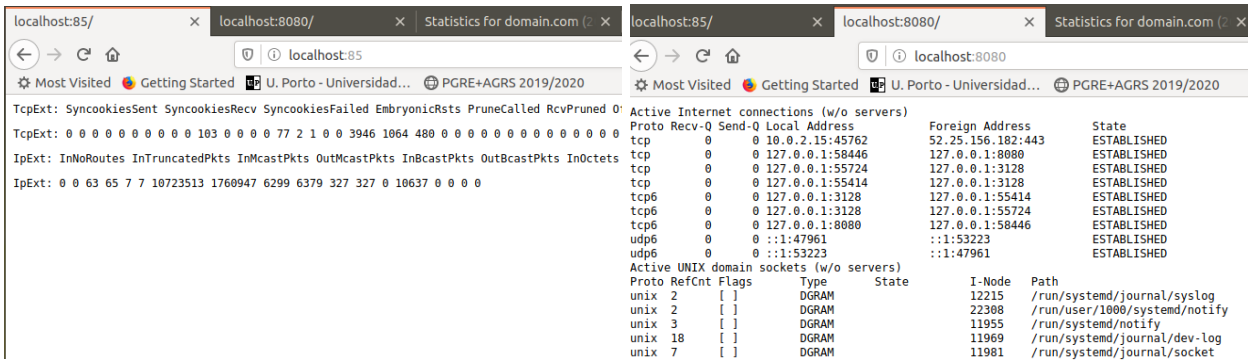


Figure 2: Site 1 and Site 2 after VH configuration

[1] http://httpd.apache.org

3 Proxy Squid

Squid is a popular caching HTTP web proxy that supports HTTP, HTTPS, FTP and more. The advantage of squid is that it reduces bandwidth and improves response time by reusing frequently requested web page. It copies only the content, rather than inefficiently copying everything. Configuration can be seen below. [2]

Squid server configuration

1. Install squid in command line.
2. Start and enable squid.
3. Change the configuration squid file (located in */etc/squid/squid.conf*) as Figure 3 shows.

```
82 acl localnet src 127.0.0.1       # enabling local host
83 acl Safe_ports port 8080         # port for domain2.com
84 acl Safe_ports port 85           # port for domain.com
85
86 acl SSL_ports port 443
87 acl Safe_ports port 80           # http
88 acl Safe_ports port 21           # ftp
89 acl Safe_ports port 443          # https
90 acl Safe_ports port 70           # gopher
91 acl Safe_ports port 210          # wais
92 acl Safe_ports port 1025-65535  # unregistered ports
93 acl Safe_ports port 280          # http-mgmt
94 acl Safe_ports port 488          # gss-http
95 acl Safe_ports port 591          # filemaker
96 acl Safe_ports port 777          # multiling http
97 acl CONNECT method CONNECT
```

Figure 3: Squid configuration file

4. Restart squid.
5. **Change the network setting of the browser** (in our case Firefox) to be able to access Squid locally.
   - In *about:config* settings enable *network.proxy.allow_hijacking_localhost*.
   - Configure Proxy Access To the Internet and set Manual proxy configuration (in Figure 4).

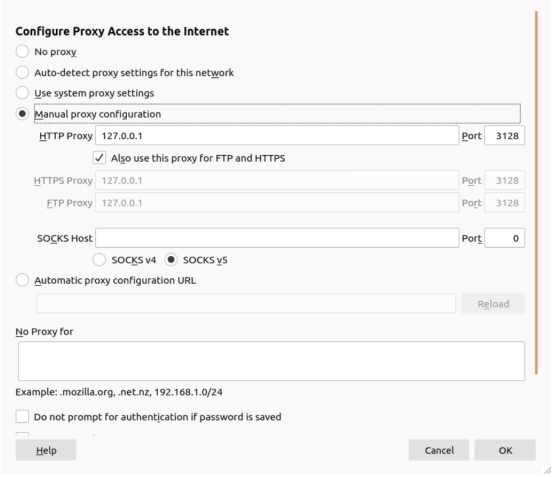


Figure 4: Setting manual configuration for proxy in Firefox

4 Web page analysis

In this section we will describe two popular analysis tools Webalizer and AwStats that provide web statistics using raw server log files. In the end of this section we will compare them with W3perl which is also a logfile analyzer.

4.1 Webalizer

The Webalizer is a fast, free web server log file analysis program. It produces highly detailed, easily configurable usage reports in HTML format, for viewing with a standard web browser. [7]

Configuration of Webalizer

After installing webalizer, it was necessary to create configuration file for each port (in our case port 8080 in Figure 5a and port 85 in Figure 5b) to specify the **output directory** and name of **log files**.

```
LogFile /var/log/apache2/domain.com-access


Incremental no

# LogType defines the log type being processed.  Normally, the Webalizer
# expects a CLF or Combined web server log as input.  Using this option,
# you can process ftp logs (xferlog as produced by wu-ftp and others),
# Squid native logs or W3C extended format web logs. Values can be 'clf',
# 'ftp', 'squid' or 'w3c'.  The default is 'clf'.

#LogType        clf

# OutputDir is where you want to put the output files.  This should
# should be a full path name, however relative ones might work as well.
# If no output directory is specified, the current directory will be used.

OutputDir /var/www/domain.com/stats
```

(a) Webalizer configuration for port 8080

```
LogFile /var/log/apache2/domain2.com-access.log


Incremental no

# LogType defines the log type being processed.  Normally, the Webalizer
# expects a CLF or Combined web server log as input.  Using this option,
# you can process ftp logs (xferlog as produced by wu-ftp and others),
# Squid native logs or W3C extended format web logs. Values can be 'clf',
# 'ftp', 'squid' or 'w3c'.  The default is 'clf'.

#LogType        clf

# OutputDir is where you want to put the output files.  This should
# should be a full path name, however relative ones might work as well.
# If no output directory is specified, the current directory will be used.

OutputDir /var/www/domain2.com/stats
```

(b) Webalizer configuration for port 85

Figure 5: Configuration of Webalizer in */etc/webalizer/* folder

Our results

After running Webalizer, we got several reports (html) and graphics for each month processed. The *index.html* shows statistics for a 12 month period and links to each month where is located detailed statistics with additional links. Generated report (example in Figure 6a) gives us information about:

- **Hits** – any request made to the server which is logged (can be also graphics images, audio files etc.),
- **Files** – requests that require the server to send something back to the client,
- **Pages** – any HTML document,[2]
- Sites – shows how many unique IP addresses made request to the server (doesn't equal to number of visitors),
- **KBytes** – amount of data that was sent out by the server.

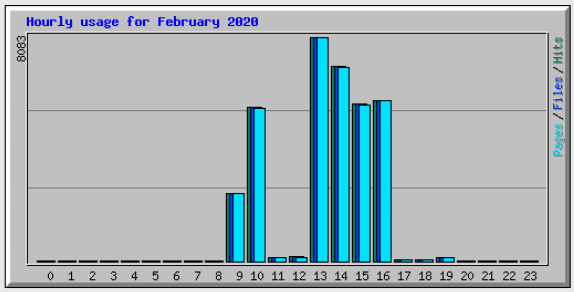
Advantages

- written in C (fast, portable)
- handles different format of log files (CLF, FTP format logs, Squid proxy server format logs etc.)
- supports multiple languages (Czech, Croatian, Portuguese)
- open source [7]

[2]What actually constitutes a page van vary from server to server, the default are files with extension .htm, .html or .cgi.

Day	Hits		Files		Pages		Visits		Sites		kB F	
24	5639	15,99%	5620	15,95%	5601	15,96%	9	12,33%	1	100,00%	7340	16,01%
25	5783	16,40%	5778	16,40%	5748	16,38%	15	20,55%	1	100,00%	7558	16,49%
26	3475	9,85%	3470	9,85%	3452	9,83%	8	10,96%	1	100,00%	4445	9,70%
27	7430	21,07%	7421	21,06%	7382	21,03%	4	5,48%	1	100,00%	9695	21,15%
28	5499	15,59%	5498	15,61%	5477	15,60%	19	26,03%	1	100,00%	7115	15,52%
29	7432	21,07%	7432	21,10%	7429	21,16%	17	23,29%	1	100,00%	9675	21,11%

(a) Statistics for 6 days in a table



(b) Statistics graphically

Figure 6: Webalizer statistics

Disadvantages

- does not differentiate between robot and human visits
- over-estimation of data
- absence of cookies to identify visitors
- 30 minute visitors window (after 30 minutes of visiting a page it will be counted as a new visit)

4.2 AWStats

AWStats is a free tool that generates graphically web, streaming, ftp or mail server statistics. It works as a CGI or from command line and shows you all possible information of your log files in few graphical web pages. [1]

As Webalizer, AWStats will also need a configuration file for each port in order to define the input log files and the output directory files of each domain. Config files are located in */etc/awstats/* folder.

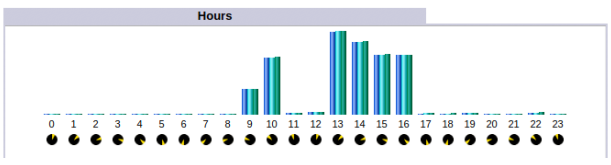
Our results

We have been collecting data from February 24. Results are displayed in Figure 7. Except number of **visit**, **pages**, **hits** and **bandwidth**, AwStats is giving us more information such as:

- visit duration,
- authenticated users,
- rush hours (pages, hits, KB for each hour and day of week),
- use of operation system (pages, hits, KB for each OS),
- HTTP errors,
- visits of robots etc.

Day	Number of visits	Pages	Hits	Bandwidth
24 Feb 2020	3	5,658	5,680	7.33 MB
25 Feb 2020	4	6,018	6,057	7.88 MB
26 Feb 2020	5	3,504	3,547	4.52 MB
27 Feb 2020	3	7,493	7,537	9.82 MB
28 Feb 2020	5	5,540	5,546	7.14 MB
29 Feb 2020	5	7,437	7,437	9.47 MB

(a) Statistics for 6 days



(b) Statistics in a graph

Figure 7: AwStats statistics

Advantages

- written in Perl
- can differentiate robot from human
- displays higher amount and better presentation of data

Disadvantages

- no advanced features (compared for example with Google Analytics)
- some bots cannot be identified
- provides a visitor window of 60 minutes, while the window is of 30 minutes in case of Webalizer

4.3 Webalizer versus Awstats

Both tools are popular for server side analysis. They are both free but we can find several differences between them.

- AWStats has nicer graphical representation.
- AWStats doesn't take too many GB of files.
- Awstats can differentiate a robot from human.
- Webalizer is a better option if you have tons of log files to analyze.
- Webalizer works in a similar way to AWStats – in that it interprets server logfiles but it doesn't try by default, to differentiate between robot and human visits.
- AWStats displays higher amount and better presentation of data as compared to Webalizer.

4.4 W3perl

W3Perl is a free software log file analyser, which can parse for example Web, FTP, DHCP or Squid log files. It support most major web log files [9, 10]. It has a lot of features as:

- real time stats,
- error statistics,
- session stats,
- scripts stats.

Configuration of W3perl

Installation is done easily throw the executable file that can be downloaded from the official web page[3].

A possible way to configure w3perl is by accessing an administration interface which allows building configuration files from a web interface (*http://127.0.0.1/w3perl/admin/*) and create a new configuration for each port (in Figure 8). Manually , it necessary to create a configuration file for each domain and save it to */usr/lib/cgi-bin/w3perl/*. We chose this way because we were having a lot of fatal errors during the configuration at the admin page of W3perl. Finally, we got statistics window for each site as we see for port 8080 in Figure 9.

[3] `http://www.w3perl.com/`

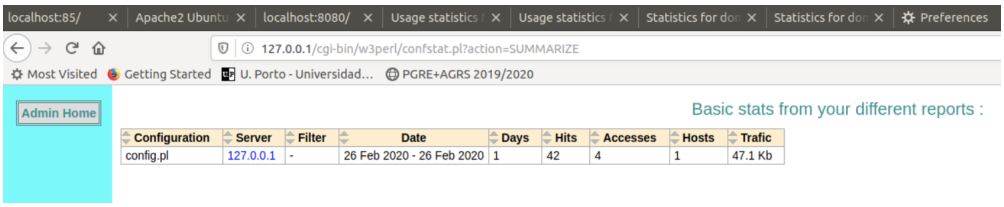


Figure 8: Creating a configuration on admin page of W3perl

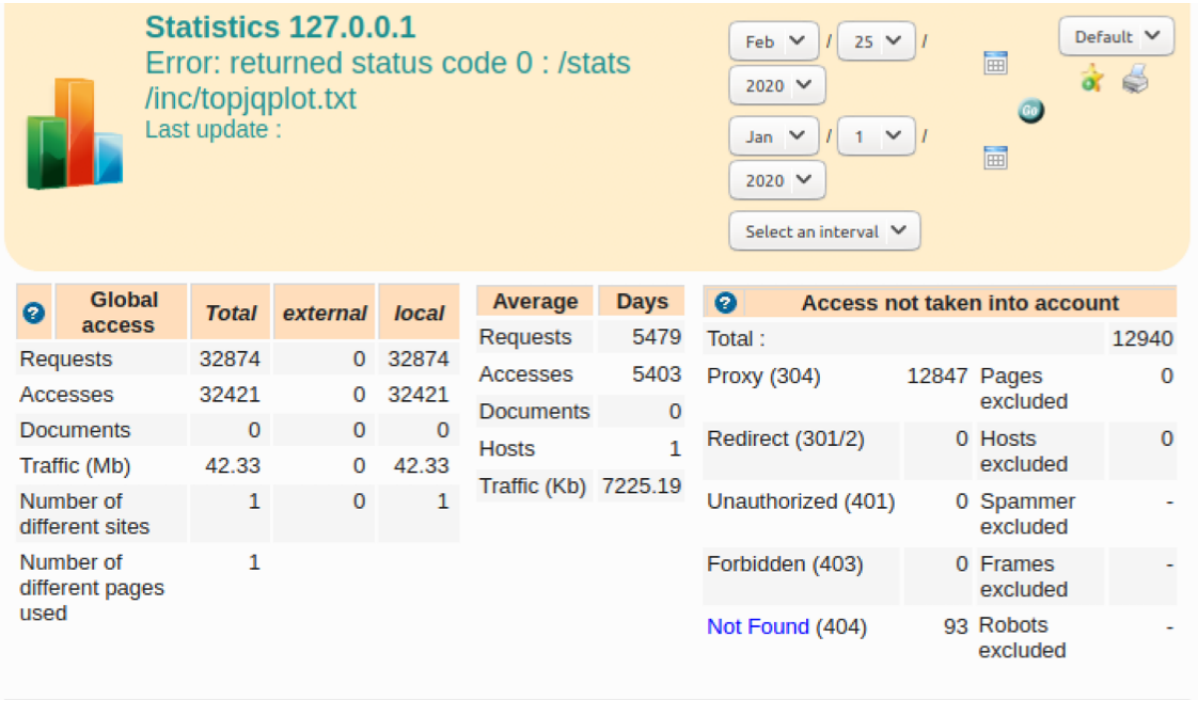


Figure 9: W3perl interface

Comparison of W3perl with Webalizer and AWStats

All these tools serve to analyse log files but we were able to find few differences.

- AWStats and W3Perl are written in Perl and they are still active, but are slower then Webalizer.
- AWStats statistics are more user friendly, but it produces less statistics then W3Perl.
- W3Perl is a better option if you want granular statistics and you don't have too many GB of log files.
- W3perl has an administration interface allow to configuration/launch your stats from a web interface. [6]

4.5 Crontab

Crontab is a Unix program that is used to view or edit the table of commands to be run by cron on a given schedule. [8]

Firstly, it is necessary to configure the cron file using the command **crontab -e**. In the configuration file it is necessary to enter all the commands we want to execute and also schedule the time they should run.

The configuration of Crontab is shown in Figure 10 where we can see the commands we want to run and the time. In this case, Crontab is configured to do the log rotate and restart the Apache server every day and execute AWStats, Webalizer and W3perl commands every 5 minutes in order to collect all new data and update the latest statistical data.

```
*/5 * * * * /usr/lib/cgi-bin/awstats.pl -config=domain.com -update > /dev/null
*/5 * * * * /usr/lib/cgi-bin/awstats.pl -config=domain2.com -update > /dev/null
*/5 * * * * webalizer -c /etc/webalizer/webalizer2.conf
*/5 * * * * webalizer -c /etc/webalizer/webalizer90.conf
*/5 * * * * /usr/lib/cgi-bin/w3perl/cron-w3perl.pl -c /usr/lib/cgi-bin/w3perl/c
onfig85.pl -a
*/5 * * * * /usr/lib/cgi-bin/w3perl/cron-w3perl.pl -c /usr/lib/cgi-bin/w3perl/c
onfig90.pl -a
0 0 * * * logrotate -vf /etc/logrotate.conf
0 0 * * * sudo systemctl restart apache2
```

Figure 10: Crontab configuration

4.6 Comparison

We have been collecting data for a week from Webalizer and Awstats to be able to compare them. W3perl was configured later therefore we have only data for four days. Comparison of requests of each tool is located in Figure 11.

In both domains it is possible to verify that the three traffic control tools present results as expected. There is a maximum difference of 200 requests between each tool over the week (in both sites). This difference can be caused by different heuristics that these tools are using.

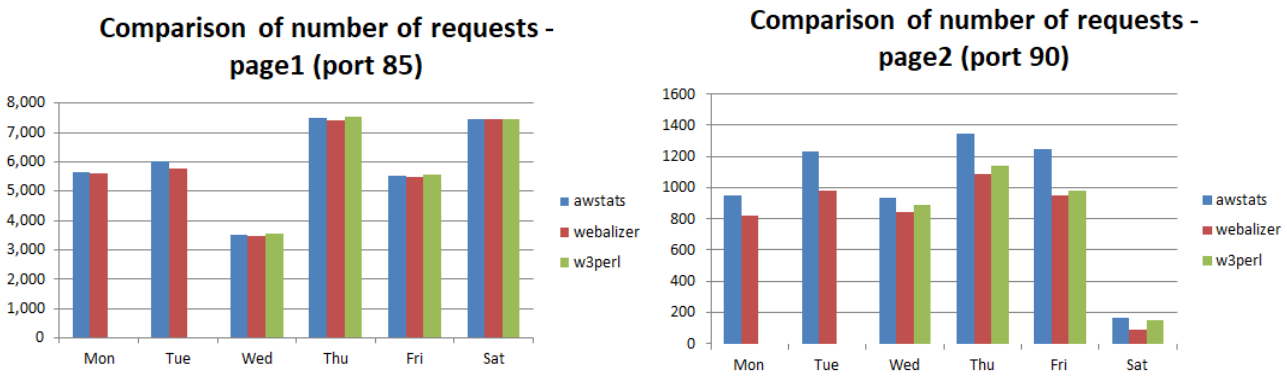


Figure 11: Requests of first and second page measured by three different tools.

5 Conclusion

To conclude, we learnt how to work with powerful tools like Crontab and the others web tools. In addition, we managed to configure Apache server and Squid as a proxy server. We achieved to get results from Webalizer, AWStats and W3perl and compare this tools. In our opinion, AWStats is better than Webalizer because it has more features. W3perl is good that has nice graphical interface but on the other hand is very hard to configure.

References

[1] *AWStats. [Online, visited 3.2.2020]. URL* https://awstats.sourceforge.io/

[2] *Squid. [Online, last change 9.5.2013]. URL* www.squid-cache.org

[3] *Apache. [Online, visited 3.2.2020]. URL* https://httpd.apache.org/

[4] *Apache HTTP Server. [Online, last change 19.2.2020]. URL* https://en.wikipedia.org/wiki/Apache_HTTP_Server

[5] *Webalizer VS. AWStats. [Online, published 27.11.2017]. URL* https://www.buycpanel.com/webalizer-vs-awstats/

[6] *W3perl. [Online, last change 14.02.2011]. URL* https://serverfault.com/questions/234245/the-most-powerful-open-source-web-analytics-tools

[7] *Webalizer. [Online, last change 11.1.2019]. URL* https://en.wikipedia.org/wiki/Webalizer

[8] *Crontab. [Online, last change 16.11.2019]. URL* https://www.computerhope.com/unix/ucrontab.htm

[9] *W3perl. [Online, last change 19.4.2019]. URL* https://en.wikipedia.org/wiki/W3Perl

[10] *W3perl. [Online, visited 3.4.2020]. URL* http://www.w3perl.com