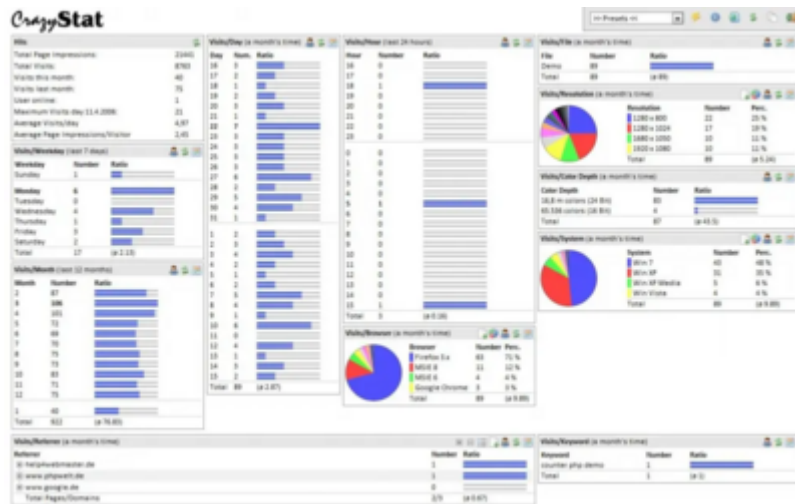


Statistik Programme



Folgende Programme sind interessant:

CrazyStat	https://www.christosoft.de/
PHP Web Stat	https://www.php-web-statistik.de/
Webalizer	http://www.webalizer.org/
HTTP-ANALYZE	http://www.http-analyze.org/
kleines Counter Script	https://www.bauser-enterprises.com/html/counter.php
Absolute Log Analyzer	http://www.bitstrike.com/analyzer/

Statistiken in PHP

Logfiles mit PHP generieren

Wenn der Besucher eine Seite aufruft, dann werden verschiedene Daten in ein „Logfile“ gespeichert. Es werden IP, Datum, die aufgerufene Datei, Status der Datei, die Größe der Datei und woher der Besucher kommt gespeichert.

So sieht ein Logeintrag aus:

```
1 09.10.2015 22:47:10 5.178.162.14 /index.php Mozilla/5.0 (Windows NT 6.1; WOW64; rv:41.0)
Gecko/20100101 Firefox/41.0
```

Das Logscript sieht wie folgt aus:

[statis.php](#)

```
<?php
$format = "csv"; //Moeglichkeiten: csv und txt

$datum_zeit = date("d.m.Y H:i:s");
$ip = $_SERVER["REMOTE_ADDR"];
$site = $_SERVER['REQUEST_URI'];
$browser = $_SERVER["HTTP_USER_AGENT"];
```

```
$monate = array(1=>"Januar", 2=>"Februar", 3=>"Maerz", 4=>"April",
5=>"Mai", 6=>"Juni", 7=>"Juli", 8=>"August", 9=>"September",
10=>"Oktober", 11=>"November", 12=>"Dezember");
$monat = date("n");
$jahr = date("y");

$dateiname="logs/log_". $monate[$monat]. "_$jahr.$format";

$header = array("Datum", "IP", "Seite", "Browser");
$infos = array($datum_zeit, $ip, $site, $browser);

if($format == "csv") {
    $eintrag= ' '.implode(' ', $infos).' ';
} else {
    $eintrag = implode("\t", $infos);
}

$write_header = !file_exists($dateiname);

$datei=fopen($dateiname,"a");

if($write_header) {
    if($format == "csv") {
        $header_line = ' '.implode(' ', $header).' ';
    } else {
        $header_line = implode("\t", $header);
    }

    fputs($datei, $header_line. "\n");
}

fputs($datei,$eintrag. "\n");
fclose($datei);
?>
```

Die Logdateien werden im Ordner logs hinterlegt, dabei ein Logfile je Monat. Bei Format könnt ihr zwischen txt und csv wählen. Bei txt werden die Logdaten in eine Textdatei geschrieben, bei csv in ein Format welches sich mittels Excel bequem öffnen lässt.

Das Script im Detail

In Zeile 5 bis 7 werden verschiedene Information abgefragt, beispielsweise die IP, die aufgerufene Seite sowie die Browser- und Betriebssystemkennung des Besuchers.

Zeile 9 bis 13 behandeln die Benennung des Logfiles. Zuerst wird ein Array mit den deutschen Monatsnamen erstellt, danach die Information zum Monat und Jahr abgefragt (siehe [Datum und Uhrzeit](#)). Die Logfiles werden im Ordner logs abgespeichert beispielsweise mit dem Namen log_November_2015.txt.

Zeile 15 und 16 beinhaltet unsere verschiedenen Informationen, die wir in den Logfile schreiben wollen und speichert dies für die leichtere Weiterverarbeitung in einem Array ab.

Zeile 18 bis 22 überprüft das Format. Falls csv ausgewählt wurde, werden die Einträge im Format „Datum“;„IP“;„Seite“;„Browser“ abgespeichert. Im Falle von Text-Dateien im Format „Datum IP Seite Browser“. Dies geht mittels der Funktion implode() .

Zeile 24 überprüft, ob die Datei bereits existiert. Falls nicht, wird in Zeile 28-36 ein Header in die Datei geschrieben. Der Header beinhaltet die Information, welche Daten in den verschiedenen Spalten stehen.

Die eigentlichen Daten zum Aufruf werden in Zeile 38 in die Datei geschrieben.

Autor: [Nils Reimers](#)

Einfaches PHP Script

[download](#)

```
<?
foreach (getallheaders() as $name => $value) {
    echo "$name: $value<br>";
}
?>
```

Ausgabe:

```
Host: www.dl2fbo.de
X-Real-IP: 188.107.167.58
X-Forwarded-For: 188.107.167.58
Connection: close
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:76.0)
Gecko/20100101 Firefox/76.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate, br
Cookie: PHPSESSID=21d38c033027fae3be5bcbe4ab6d7be6
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

PHP Function

[download](#)

```
if (!function_exists('getallheaders')) {
```

```
/**
 * Get all HTTP header key/values as an associative array for the
 * current request.
 *
 * @return string[string] The HTTP header key/value pairs.
 */
function getallheaders()
{
    $headers = array();

    $copy_server = array(
        'CONTENT_TYPE'    => 'Content-Type',
        'CONTENT_LENGTH' => 'Content-Length',
        'CONTENT_MD5'     => 'Content-Md5',
    );

    foreach ($_SERVER as $key => $value) {
        if (substr($key, 0, 5) === 'HTTP_') {
            $key = substr($key, 5);
            if (!isset($copy_server[$key]) ||
!isset($_SERVER[$key])) {
                $key = str_replace(' ', '-',
ucwords(strtolower(str_replace('_', ' ', $key))));
                $headers[$key] = $value;
            }
        } elseif (isset($copy_server[$key])) {
            $headers[$copy_server[$key]] = $value;
        }
    }

    if (!isset($headers['Authorization'])) {
        if (isset($_SERVER['REDIRECT_HTTP_AUTHORIZATION'])) {
            $headers['Authorization'] =
$_SERVER['REDIRECT_HTTP_AUTHORIZATION'];
        } elseif (isset($_SERVER['PHP_AUTH_USER'])) {
            $basic_pass = isset($_SERVER['PHP_AUTH_PW']) ?
$_SERVER['PHP_AUTH_PW'] : '';
            $headers['Authorization'] = 'Basic ' .
base64_encode($_SERVER['PHP_AUTH_USER'] . ':' . $basic_pass);
        } elseif (isset($_SERVER['PHP_AUTH_DIGEST'])) {
            $headers['Authorization'] =
$_SERVER['PHP_AUTH_DIGEST'];
        }
    }

    return $headers;
}
}
```

Ausgabe

```
echo "<pre>"; print_r( getallheaders() ); echo "</pre>"; if
(!function_exists('getallheaders')) { /** * Get all HTTP header key/values
as an associative array for the current request. * * @return string[string]
The HTTP header key/value pairs. */ function getallheaders() { $headers =
array(); $copy_server = array( 'CONTENT_TYPE' => 'Content-Type',
'CONTENT_LENGTH' => 'Content-Length', 'CONTENT_MD5' => 'Content-Md5', );
foreach ($_SERVER as $key => $value) { if (substr($key, 0, 5) === 'HTTP_') {
$key = substr($key, 5); if (!isset($copy_server[$key]) ||
!isset($_SERVER[$key])) { $key = str_replace(' ', '-',
ucwords(strtolower(str_replace('_', ' ', $key)))); $headers[$key] = $value; }
} elseif (isset($copy_server[$key])) { $headers[$copy_server[$key]] = $value;
} } if (!isset($headers['Authorization'])) { if
(isset($_SERVER['REDIRECT_HTTP_AUTHORIZATION'])) { $headers['Authorization']
= $_SERVER['REDIRECT_HTTP_AUTHORIZATION']; } elseif
(isset($_SERVER['PHP_AUTH_USER'])) { $basic_pass =
isset($_SERVER['PHP_AUTH_PW']) ? $_SERVER['PHP_AUTH_PW'] : '';
$headers['Authorization'] = 'Basic ' .
base64_encode($_SERVER['PHP_AUTH_USER'] . ':' . $basic_pass); } elseif
(isset($_SERVER['PHP_AUTH_DIGEST'])) { $headers['Authorization'] =
$_SERVER['PHP_AUTH_DIGEST']; } } return $headers; } }
```

From:

<https://wiki.hennweb.de/> - **HennWeb**

Permanent link:

<https://wiki.hennweb.de/doku.php?id=programmieren:statistiken>Last update: **27/05/2020 11:39**