

Статистика в Dokuwiki

Код на этой странице описывает, как Dokuwiki может быть расширен с помощью журнала. Это может быть полезно, если файлы журнала веб-сервера находятся вне диапазона и требуется некоторая статистика об использовании Dokuwiki.

Я опишу две версии. Первая — это решение высокого уровня, которое будет регистрировать доступ к вики-страницам, а также к внутренним и внешним медиа-файлам. Вторая — для «новичков» и будет регистрировать только вики-страницы.



[примечание J.-F. Lalande] Используя информацию и код, предоставленные на этой странице, я создал плагин [logstats](#) который генерирует запись в access.log для каждого доступа к странице dokuwiki. Вы можете увидеть подробности и загрузить его на моей странице [плагина logstat](#).

Формат файла журнала

Оба решения будут использовать формат файла журнала NCSA combined или NCSA extended. Этот формат файла журнала очень популярен и часто используется на веб-серверах, таких как apache. Многие генераторы отчетов могут читать этот формат и создавать из него хорошие отчеты. Благодаря этой фантастической поддержке внешних программ Dokuwiki не нуждается в какой-либо встроенной функции отчетности.

Можно использовать следующие генераторы отчетов (это лишь некоторые примеры, список далеко не полный):

- [AWStats](#) - Хороший генератор отчетов из Франции. Заметки по настройке DokuWiki в [Приложениях](#) В ниже
- [Webalizer](#) - Еще один известный генератор отчетов
- ...

Формат файла журнала состоит из нескольких полей, объединенных в одну строку:

```
<host> <rfc931> <user> [<timestamp>] "<request>" <error> <filesize>
"<referer>" "<agent>"
```

- <host> - IP-адрес клиентского хоста (мы не делаем обратный поиск хоста)
- <rfc931> - удаленная идентификация пользователя или «-», если недоступно
- <user> - идентификатор аутентифицированного пользователя или «-», если недоступно
- <timestamp> - время в формате [01/Dec/2005:22:19:12 +0200]
- <request> - Запрошенный протокол, например GET или POST, запрошенная страница и протокол
- <error> - код ошибки от сервера, например, 200 (OK) или 404 (файл не найден)
- <filesize> - размер страницы вики (только голый текст)
- <referer> - страница, с которой пришел пользователь. Эта информация очень зависит от клиента и не всегда доступна. Функция регистрации делает все возможное, чтобы заполнить здесь полезную информацию.

- <agent> - идентификационная информация, которую клиентский браузер сообщает о себе

Окончательная статистика

Ultimate statistics будет регистрировать страницы вики, внутренние и внешние медиафайлы. Основная часть кода была помещена в *logfile.php*. Этот файл имеет размер около 4 Кбайт, и поскольку мне не разрешено загружать файлы в эту вики, я добавил исходный код в конце этой страницы (см. [Приложение А](#)).

Давайте посмотрим, что нам нужно для получения максимальной статистической работы.

1. В качестве первого шага скопируйте файл *logfile.php* в */inc*
2. Вторым шагом нам нужно сообщить Dokuwiki, что она должна заботиться о файле журнала. Это делается в функции *init_paths()* файла */inc/init.php*. Эта функция оценивает некоторые имена путей и сохраняет их в массиве *\$conf[]*. Измените функцию *init_paths()* так, чтобы она выглядела следующим образом:

```
function init_paths(){
    global $conf;

    $paths = array('datadir' => 'pages',
                  'olddir' => 'attic',
                  'mediadir' => 'media',
                  'metadir' => 'meta',
                  'cachedir' => 'cache',
                  'indexdir' => 'index',
                  'lockdir' => 'locks',
                  'tmpdir' => 'tmp',
                  'accesslog' => 'access.log');

    foreach($paths as $c => $p){
        if(empty($conf[$c])) $conf[$c] = $conf['savedir'].'/'.$p;
        $conf[$c] = init_path($conf[$c]);
        if($c != 'accesslog' && empty($conf[$c])) nice_die("The $c ('$p') does not exist, isn't accessible or writable.
                                                You should check your config and permission settings.
                                                Or maybe you want to <a href=\"install.php\">run the installer</a>?");
    }
}

[...]
```

Dokuwiki теперь проверит, существует ли файл журнала *access.log*, и только в этом случае путь сохраняется в *\$conf['accesslog']*.

3. Следующий шаг — добавить функцию в *inc/template.php* для сохранения согласованности API шаблона . Но сначала нам нужно включить *inc/logfile.php*.

```

if(!defined('DOKU_INC')) die('meh.');
require_once(DOKU_INC.'inc/logfile.php');

[...]

/** 
 * зарегистрировать эту страницу в файле журнала
 *
 * @author Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
 *
*/
function tpl_logfile(){
    global $ID;

    logPageAccess(cleanID($ID));
}

```

4. Обновите ваш любимый шаблон для вызова `tpl_logfile()`. Лучшее место — строка сразу после вызова `tpl_indexerWebBug()`. Скорее всего, это находится в конце файла `main.php` вашего шаблона:

```

[...]

</div>
<div class="no"><?php /* provide DokuWiki housekeeping */
tpl_indexerWebBug(?)?></div>
<div class="no"><?php /* do the logging stuff */ tpl_logfile()?></div> <-
ADD THIS LINE
</body>
</html>

```

5. Чтобы иметь возможность регистрировать медиафайлы, нам нужно изменить `/lib/exe/fetch.php`. Сначала нам нужно снова включить `inc/logfile.php`. Добавьте оператор `include` ниже всех остальных, которые уже есть в `lib/exe/fetch.php`. Функция `logMediaAccess()` создает запись в журнале и должна быть вставлена после проверки источника медиафайла. Хорошее место — рядом с строкой 60 строка с комментарием «проверка существования файла», непосредственно перед проверкой существования (см. комментарии ниже).

```

require_once(DOKU_INC.'inc/logfile.php');

[...]

//регистрировать доступ к мультимедиа
logMediaAccess( $MEDIA , $FILE ) ;

//проверка существования файла

```

6. Вот и все. Теперь ведение журнала должно работать. Остался только один шаг — создать пустой файл журнала `/data/access.log`. Процедуры файла журнала будут записывать только в уже существующий файл журнала. Если файл `/data/access.log` не существует, ничего не

будет сделано.

Базовая статистика

Базовая статистика означает, что только доступ к странице вики вызовет запись в файл журнала. Если вам также нужна регистрация медиафайлов, прочтите главу [Ultimate Statistics](#).

Для запуска базовой статистики необходимо всего несколько шагов. Первым шагом будет указание DokuWiki, где находится или должен находиться файл журнала. Добавьте следующую строку в ваш *local.php* :

```
$conf['logfile'] = './data/access.log'; //location of log file
```

В качестве второго шага добавьте следующую функцию в *inc/template.php*:

```
/**  
 * Эта функция записывает информацию о доступе к текущей странице в журнал  
 * файл. Он использует комбинированный формат файла журнала, который также  
 используется  
 * веб-сервер apache. Целая куча доступных анализаторов логов может быть  
 * используется для визуализации журнала.  
 */  
  
function tpl_logfile(){  
    global $ID;  
    global $conf;  
  
    $exists = false;  
    $page = cleanID($ID);  
  
    resolve_pageid('', $page, $exists);  
    $page = str_replace(':', '/', $page);  
    $page = utf8_encodeFN($page);  
  
    $host      = $_SERVER['REMOTE_ADDR'];  
    $user      = isset($_SERVER['REMOTE_USER']) ? $_SERVER['REMOTE_USER'] :  
    "-" ;  
    $timestamp = date("d/M/Y:H:i:s O");  
    $method    = isset($_SERVER['REQUEST_METHOD']) ?  
    $_SERVER['REQUEST_METHOD'] : "";  
    $protocol  = isset($_SERVER['SERVER_PROTOCOL']) ?  
    $_SERVER['SERVER_PROTOCOL'] : "";  
    $filesize  = @filesize(wikiFN($ID));  
    $status    = $exists ? "200 $filesize" : "404 0";  
    $agent     = isset($_SERVER['HTTP_USER_AGENT']) ?  
    $_SERVER['HTTP_USER_AGENT'] : "";  
    $referer   = $_SERVER['PHP_SELF'];
```

```

$logline = "$host - $user $timestamp \"$method $page $protocol\" $status
\"$referer\" \"$agent\"\n";
io_saveFile($conf['logfile'], $logline, true);
}

```

В качестве третьего и последнего шага добавьте следующую строку в *main.php* вашего любимого шаблона. Стока сразу за вызовом функции индексатора будет вполне подходящей:

```
<?php tpl_logfile() ?>
```

Вот и все. Побродите немного по своей вики, а затем посмотрите на файл журнала. Он должен содержать строку для каждой вызванной страницы.

Приложение A

Исходный код *logfile.php*

[logfile.php](#)

```

<?php
/**
 * Функции ведения журнала DokuWiki
 *
 * @license GPL 2 (http://www.gnu.org/licenses/gpl.html)
 * @автор Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
 */

if(!defined('DOKU_INC'))
define('DOKU_INC', realpath(dirname(__FILE__).'../../').'/');
require_once(DOKU_CONF.'dokuwiki.php');

/**
 * украсить идентификатор страницы вики для журнала
 *
 * Идентификатор страницы вики будет преобразован в строку имени файла
 * коды utf8 будут закодированы.
 *
 * @param $id идентификатор вики-страницы
 *
 * @автор Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
 */
function prepareID($path){
    $path = cleanID($path);
    $path = str_replace(':', '/', $path);
    $path = utf8_encodeFN($path);
    return $path;
}

/**

```

```
* проверяет, существует ли файл, и возвращает соответствующий веб-адрес
* статус сервера
*
* @param $file полный путь к файлу для проверки
*
* @автор Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
*/
function getStatus($file){
    if(@file_exists($file)){
        $size = @filesize($file);
        return "200 $size";
    }else
        return "404 0";
}

/**
* регистрирует доступ к вики-странице
*
* @param $id идентификатор страницы вики, включая пространство имен
*
* @автор Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
*/
function logPageAccess($id){
    global $ACT;

    if ($ACT == 'show'){
        $page = prepareID($id);

        $crumbs = breadcrumbs(); // получить последние посещенные
страницы
        $crumbs = array_keys($crumbs); // получить необработанные
идентификаторы страниц
        array_pop($crumbs); // пропустить текущую страницу
        $referer = array_pop($crumbs); // получить предыдущую страницу
текущей страницы
        $referer = ($referer) ? prepareID($referer) : '';

        logAccess($page,getStatus(wikiFN($id)), $referer);
    }
}

/**
* регистрирует доступ к медиафайлу (внутренний или внешний)
*
* @param $media url или путь к медиа в dokuwiki
* @param $file полный путь к медиафайлу
*
* @автор Маттиас Гримм <matthiasgrimm@users.sourceforge.net>
*/
function logMediaAccess($media,$file){
    if(!preg_match('#^(https?|ftp)://#i', $media))
```

```
$media = prepareID($media);

logAccess($media, getStatus($file));
}

/**
* создает запись в файле журнала и записывает ее в журнал
*
* Эта функция записывает информацию о доступе к текущей странице в журнал
* файл. Он использует комбинированный формат файла журнала, который также
используется
* веб-сервер apache. Целая куча доступных анализаторов логов может быть
* используется для визуализации журнала.
*
* @param $page имя вызванной страницы
* @param $status HTTP-код статуса, за которым следует размер файла
* @param $referer предшественник $page (какая страница ссылается на $page)
* Если это поле пустое, функция пытается получить
* реферер с веб-сервера (HTTP_REFERER)
*
* @author Маттиас Гримм <matthias.grimmt@users.sourceforge.net>
*
* комбинированный формат файла журнала:
* <хост> <rfc931> <пользователь> [<метка времени>] "<запрос>" <ошибка>
<размер файла>
* "<реферер>" "<агент>" \n
*
* <host> IP-адрес клиентского хоста (мы не делаем обратный поиск хоста)
* <rfc931> удаленная идентификация пользователя или «-», если недоступно
* <пользователь> идентификатор пользователя или «-», если недоступно
* <timestamp> время в формате [01/Dec/2005:22:19:12 +0200]
* <request> Запрошенный протокол, например GET или POST, запрошенная
страница
* и протокол
* <error> код ошибки от сервера, например, 200 (OK) или 404 (файл
* не найдено)
* <filesize> размер страницы вики (только голый текст)
* <referer> страница, которая вызвала эту. У нас нет этой информации
* и заполнил имя скрипта dokuwiki.
* <agent> идентифицирующая информация, которую сообщает клиентский браузер
* о себе
*/
function logAccess($page,$status,$referer=''){
    global $conf;

    if (!empty($conf['accesslog'])){
        $host      = $_SERVER['REMOTE_ADDR'];
        $user      = isset($_SERVER['REMOTE_USER']) ?
$_SERVER['REMOTE_USER'] : "-";
        $timestamp = date("[d/M/Y:H:i:s O]");
        $method    = isset($_SERVER['REQUEST_METHOD']) ?

```

```

$_SERVER['REQUEST_METHOD'] : "";
$protocol = isset($_SERVER['SERVER_PROTOCOL']) ? 
$_SERVER['SERVER_PROTOCOL'] : "";
$agent = isset($_SERVER['HTTP_USER_AGENT']) ? 
$_SERVER['HTTP_USER_AGENT'] : "";

if (empty($referer)){
    if(isset($_SERVER['HTTP_REFERER'])){
        $cnt =
preg_match('/\?id=((\w+:\*)+)/i', $_SERVER['HTTP_REFERER'], $match);
        if($cnt == 1)
            $referer = prepareID($match[1]);
    }
}

$logline = "$host - $user $timestamp \"$method $page $protocol\""
$status "\"$referer\" \"$agent\"\n";
io_saveFile($conf['accesslog'], $logline, true);
}
}

//Настройка VIM: например: et ts=2 enc=utf-8 :

```

Приложение В – Конфигурация AWStats

Если вы решили использовать AWStats для обработки своих журналов, то вот несколько советов по параметрам конфигурации, которые дадут вам больше контроля над тем, как отображаются доступы к DokuWiki.

- По умолчанию AWStats не отслеживает параметры после «?» в URL . Поэтому все ваши обращения к вики будут отображаться как обращения к одной странице «doku.php». Если вы хотите увидеть, к каким страницам вики осуществляется доступ, вы можете включить отслеживание параметров.

```

URLWithQuery=1 # Set this to "1" to enable tracking of URL parameters
URLWithQueryWithOnlyFollowingParameters="id media" # Use this to limit which
parameters you are interested in.

```

- AWStats позволяет вам создавать настраиваемые отчеты. Вот 3 дополнительных настраиваемых отчета, которые показывают 10 лучших:
 - просмотренные страницы вики в порядке убывания популярности
 - Файлы wiki-медиа, загруженные в порядке убывания популярности
 - поиски, выполненные с использованием поля поиска вики

ПРИМЕЧАНИЕ: Вам придется отредактировать следующие строки в соответствии с вашей установкой:

- ExtraSectionCondition* - укажите URL- путь к файлам doku.php и fetch.php
- ExtraSectionFirstColumnFormat* — как указано выше

- MaxNbOfExtra* — установите количество строк, которое вы хотите включить в отчет

ПРИМЕЧАНИЕ: Результаты имеют следующие ограничения:

- DokuWiki иногда переключается на метод POST вместо метода GET. Файлы журналов не содержат параметров для запросов POST и поэтому не будут учитываться в этих отчетах.
- Параметр «id» перегружен — он не всегда представляет собой запрос конкретной страницы.
- Если пользователь выполняет поиск, не нажимая <RETURN> или «Поиск», то поиск выполняется с использованием AJAX и не отображается в этих отчетах.

```
ExtraSectionName1="Wiki Pages"
ExtraSectionCodeFilter1="200 304"
ExtraSectionCondition1="URL, \/doku.php"
ExtraSectionFirstColumnTitle1="Page"
ExtraSectionFirstColumnValues1="QUERY_STRING,id=( [^&]+)"
ExtraSectionFirstColumnFormat1="%s"
ExtraSectionStatTypes1=PHBL
ExtraSectionAddAverageRow1=0
ExtraSectionAddSumRow1=1
MaxNbOfExtra1=10
MinHitExtra1=1

ExtraSectionName2="Wiki Media Downloads"
ExtraSectionCodeFilter2="200 304"
ExtraSectionCondition2="URL, \/fetch.php"
ExtraSectionFirstColumnTitle2="Document"
ExtraSectionFirstColumnValues2="QUERY_STRING,media=( [^&]+)"
ExtraSectionFirstColumnFormat2="%s"
ExtraSectionStatTypes2=PHBL
ExtraSectionAddAverageRow2=0
ExtraSectionAddSumRow2=1
MaxNbOfExtra2=10
MinHitExtra2=1

ExtraSectionName3="Wiki Searches"
ExtraSectionCodeFilter3="200 304"
ExtraSectionCondition3="QUERY_STRING,do=search&id=( [^&]+)"
ExtraSectionFirstColumnTitle3="Search terms"
ExtraSectionFirstColumnValues3="QUERY_STRING,id=( [^&]+)"
ExtraSectionFirstColumnFormat3="%s"
ExtraSectionStatTypes3=PHBL
ExtraSectionAddAverageRow3=0
```

Обсуждение

Можно ли сохранять результаты поиска в файле журнала?

Согласен, что было бы очень полезно иметь возможность записывать информацию в файл журнала, а не проверять логи Apache. Dopple 25/08/2009

From:

<https://wwoss.ru/> - **worldwide open-source software**



Permanent link:

<https://wwoss.ru/doku.php?id=wiki:tips:logging>

Last update: **2025/06/22 18:14**