Расширенные настройки фермы и советы

Fix Me! Большую часть старой документации фермы, которая не охвачена новой документацией фермы, следует добавить сюда.

В следующих разделах предполагается установка, аналогичная той, что описана на фермах.

CLI tools

CLI-инструменты инструменты командной строки вам нужно только задать имя животного в `animal` переменной окружения. Например, если вы хотите выполнить `bin/indexer.php` для животного «foo» вы можете использовать `animal=foo bin/indexer.php`.

Установка значений по умолчанию и ограничение опций

Вы можете добавить дополнительные файлы конфигурации для значений по умолчанию, которые должны использоваться в животных, и для ограничения определенных параметров конфигурации. Если вы добавите следующий код в конец `inc/preload.php`, основной файл конфигурации фермера будет использоваться как файл по умолчанию для всех животных, а защищенный файл фермера будет использоваться как последний защищенный файл для всех животных, что означает, что животные не смогут изменять эти параметры. Это может быть использовано для отключения определенных параметров, таких как phpok если вы не доверяете администратору животных.

```
$config_cascade['main']['default'][] = DOKU_INC.'conf/local.php';
$config_cascade['main']['protected'][] =
DOKU_INC.'conf/local.protected.php';
```

Обратите внимание, что эти изменения также влияют на фермера, если к нему есть доступ. Это сломает менеджер конфигурации в фермере, поскольку он предполагает, что все измененные значения являются значениями по умолчанию, и удалит их из конфигурации.

Аналогичные изменения можно использовать для каскада плагинов, чтобы определить набор плагинов, которые включены по умолчанию, или набор плагинов, которые должны или не должны быть включены.

Конечно, вы также можете полностью изменить каскад конфигурации, каскад конфигурации по умолчанию, который используется в фермах, вы можете найти в конце `inc/farm.php`.

Отключение доступа к фермеру

Если вы хотите отключить доступ к фермеру, вы можете добавить что-то вроде

```
if (DOKU_FARM == false) { nice_die('Access to the farmer denied'); }
```

в конце `inc/preload.php` (после включения `farm.php`).

Централизованное управление пользователями

Если некоторые или все ваши животные должны иметь общих пользователей, вы можете сделать это не только с помощью той же внешней аутентификации, но и с помощью управления пользователями DokuWiki по умолчанию, указав путь к списку пользователей в том же месте в конце `inc/preload.php`:

```
$config_cascade['plainauth.users'] = array(
    'default'   => '/your/path/to/conf/users.auth.php',
);
```

Если вы хотите сделать это только для некоторых животных, вы, конечно, можете просто добавить условие, основанное на значении `DOKU_CONF`.

Гибкие настройки на основе .htaccess

На самом деле настройки на основе .htaccess могут быть намного более гибкими, чем настройки, описанные на фермах. Для этих расширенных настроек ваш каталог фермы может и даже должен быть вне корня документов (каталога, доступного из Интернета). Если ваш каталог фермы находится внутри корня документов, вам нужно убедиться, что вы должным образом защищаете каждый каталог данных (например, используя .htaccess, как это делается в обычной настройке DokuWiki), так как в этих расширенных настройках они могут не быть автоматически защищены (зависит от используемых вами условий перезаписи).

Чтобы сделать животных доступными, как в стандартной настройке, вы можете создать второй каталог фермы внутри корня документа с файлом .htaccess и без каталогов животных. Вы также можете добавить `RewriteCond` операторы, как в примере ниже, чтобы предотвратить циклы перенаправления. Тогда ваши животные могут быть в корневом каталоге домена, т. е. вы можете иметь `example.com/animal1/` для животного `animal1` и `example.com/dokuwiki/` для фермера.

.htaccess

```
RewriteEngine On

# Prevent rewrite loops by making sure that the farmer directory won't
be matched
# If you need to enable access to other directories either add them
here or try something like
# RewriteCond %{REQUEST_FILENAME}       !-f
# RewriteCond %{REQUEST_FILENAME}       !-d
# but then also the files of your animals (if they should be in this
directory) will be accessible
```

```
# so make sure you have properly protected them

RewriteCond %{REQUEST_URI} !^/?dokuwiki/
RewriteRule ^/?([^/]+)/(.*)  /dokuwiki/$2?animal=$1 [QSA]
RewriteCond %{REQUEST_URI} !^/?dokuwiki$
# RewriteCond %{REQUEST_FILENAME}       !-f # allow files in the root
directory (like some index file)
RewriteRule ^/?([^/]+)$      /dokuwiki/?animal=$1 [QSA]
Options +FollowSymLinks
```

Более одного домена

Если вы можете использовать символические ссылки вы можете легко связать каталог в корневом каталоге документов каждого домена с фермером и указать этот каталог в правилах перезаписи.

Пользовательские пути

Если вы не хотите, чтобы все животные были в одном каталоге, вы можете просто создать файл .htaccess для каждого животного. В этом файле .htaccess вы можете указать имя животного.

.htaccess

```
RewriteEngine On

# Prevent rewrite loops by making sure that the farmer directory won't
be matched
# If you need to enable access to other directories either add them
here or try something like
# RewriteCond %{REQUEST_FILENAME}       !-f
# RewriteCond %{REQUEST_FILENAME}       !-d
# but then also the files of your animal (if they should be in this
directory) will be accessible
# so make sure you have properly protected them
RewriteCond %{REQUEST_URI} !^/?dokuwiki/
RewriteRule ^/?(.*) /dokuwiki/$1?animal=ANIMAL [QSA]
RewriteRule ^/?$ /dokuwiki/?animal=ANIMAL [QSA]

Options +FollowSymLinks
```

Операторы RewriteCond предотвращают циклы перенаправления, когда ваш каталог фермера находится внутри каталога животных. Таким образом, вы также можете поместить животное в корневой каталог домена.

Запуск фермы за обратным прокси-сервером

Это происходит, когда вы настраиваете свою ферму на внутреннем сервере (например http://11.22.33.44) и представляете свои вики-ресурсы миру через фронтенд-сервер Apache (например, http://www.example.com).

Сначала настройте ферму на сервере в соответствии со стандартной документацией.

В моем случае я решил поместить все файлы DokuWiki в одно место, чтобы отделить их от других сайтов, работающих на внутреннем сервере:

`/var/www/wikis/` – Корень моей установки DokuWiki.

Итак, моя ферма имеет следующую конфигурацию:

- `/var/www/wikis/dokuwiki`– движок DokuWiki, он должен быть доступен из Интернета
- `/var/www/wikis/farm`– каталог ферм DokuWiki, содержащий всех животных
- `/var/www/wikis/farm/bear`– моя первая вики о медведях

На внутреннем сервере моя вики о медведях (после полной настройки) будет доступна по следующему адресу http://11.22.33.44/wikis/farm/bear

Однако на сервере frontend я хочу представить свою вики по следующему адресу (обратите внимание, что я решил убрать «s» из «wikis» в этом URL): http://www.example.com/wiki/bear

Используя тип настройки .htaccess, я настроил его `/var/www/wikis/farm/.htaccess` с помощью следующего кода:

.htaccess

```
RewriteEngine On

RewriteRule ^/?([^/]+)/(.*)  /wikis/dokuwiki/$2?animal=$1 [QSA]
RewriteRule ^/?([^/]+)$      /wikis/dokuwiki/?animal=$1 [QSA]
Options +FollowSymLinks
```

Затем `/var/www/wikis/farm/bear/conf`я редактирую **local.protected.php** следующим образом (примечание: здесь используется адрес интерфейса):

local.protected.php

```
<?php
/**
 * These settings are "protected" and cannot be overwritten
 * by the configuration manager, but need to be edited manually.
 */

$conf['savedir'] = DOKU_CONF.'../data';
$conf['updatecheck'] = 0;
// if you're using the .htaccess base setup, set this to your animal's
```

```
base directory:
$conf['basedir'] = '/wiki/bear/';
```

Поэтому я использую Apache's mod_proxy и настраиваю конфигурацию в **httpd.conf** на сервере front-end:

httpd.conf

```
    <Location /wiki>
      ProxyPass http://11.22.33.44/wikis/farm
      ProxyPassReverse http://11.22.33.44/wikis/farm
    </Location>
```

Это означает, что все животные на моей ферме всегда будут проходить обратный прокси-сервер с желаемого адреса интерфейса.

Это работает очень хорошо, пока вы не выполните определенные действия (например, вход в систему, выход из системы, сохранение изменений и т. д.). Поскольку движок DokuWiki выполняет перенаправление в этих точках, браузеру конечного пользователя отправляется заголовок HTTP 302, содержащий следующее:

`Location: http://11.22.33.44/wiki/bear/doku.php?id=blahblahblah`

Конечно, на внутреннем сервере по этому адресу нет страницы, и в результате в браузере пользователя появляется сообщение об ошибке HTTP 404 «Страница не найдена».

Средство — использовать Apache's mod_headers. Мы изменяем наш блок кода в **httpd.conf** следующим образом:

httpd.conf

```
    <Location /wiki>
      ProxyPass http://11.22.33.44/wikis/farm
      ProxyPassReverse http://11.22.33.44/wikis/farm
      Header edit Location ^http://11.22.33.44 http://www.example.com
    </Location>
```

The 'Location' response header is now re-written to the correct address and everything works fine.

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

Permanent link:
https://wwoss.ru/doku.php?id=wiki:farms:advanced

Last update: **2025/01/15 21:55**