⇐ Вернуться назад

Пример создания вики-фермы

Когда я создал свою первую вики-ферму, это заняло около 5 часов. Следующая, как я ожидаю, будет значительно меньше часа. Поэтому я подумал, что другим будет полезно, если я опубликую свои заметки администратора в публичном месте. Обратите внимание, что мы не рекомендуем это как лучший способ что-то делать, а скорее как пример, который работает для нас.

Эта страница предназначена для дополнения официальной документации farms а не для ее замены. Пожалуйста, прочитайте хотя бы то объяснение перед этим.

Что мы пытаемся сделать

Этот сервер находится в специализированном исследовательском центре, который является частью физического факультета крупного университета. В настоящее время у нас есть 3 профессора, один штатный сотрудник (я) и от 15 до 30 аспирантов и студентов, участвующих в исследовании.

Мы хотим полупубличную вики для центра и частные вики для каждого профессора, чтобы управлять проектами и публиковать результаты перед тем, как они станут общедоступными. Вероятно, мы захотим очень частные вики для работы над грантовыми предложениями или сбора данных для публикации

Среда

Наш веб-сервер работает под управлением Scientific Linux 6.3 (SL6), похожего на CentOS 6.3. Мы используем Apache 2.2.

Наш ИТ-отдел выдал нам один внешний фиксированный IP-адрес и одну запись DNS . Возможно, можно получить дополнительные записи CNAME, которые являются псевдонимами для IP-адреса, чтобы позволить одному серверу размещать несколько сайтов (см. wikipedia's description. Поэтому мы решили использовать метод .htaccess, своего рода.

Мы используем PHP v5.3.3 с Zend Engine v2.3.0.

Обзор процесса

Apache, PHP уже настроен и работает с доверенным сертификатом SSL. Шаги, которые мы собираемся рассмотреть, следующие:

- Установите последнюю версию Dokuwiki из tarball. (только обзор)
- Настройте базовую вики (только обзор)

- Установить ферму
- Заблокируйте его для ограниченного доступа

С точки зрения браузера пользователь не может отличить ферму вики от нескольких установок полных или родительских вики. Мы могли бы установить несколько копий всего tarball dokuwiki в отдельные каталоги на веб-сервере, но это значительно усложнило бы текущие задачи по обслуживанию.

Причина создания фермы заключается в том, что только родительская вики должна быть обновлена до новой версии, а плагины должны быть установлены только один раз. Веб-сервер Apache также более эффективен с одной копией программы, но это вторично.

Последняя версия DokuWiki

В целом, я большой поклонник использования предварительно упакованного ПО для каждой ОС . Однако в этом случае пакетное ПО было 2-летней давности, не содержало модуля фермы и разбросало файлы php в /usr/share/dokuwiki, /etc/dokuwiki и бог знает где еще. Теперь, когда у нас есть рабочая установка, мы, вероятно, могли бы работать с пакетом, но это было запутанно.

Руководство по установке довольно простое и ему легко следовать, если вы решите пойти по этому пути.

Настройте базовую вики

В настройке базовой вики нет ничего особенного. В документации это называется «фермер» и является начальной настройкой работающей Dokuwiki, как объясняется в руководстве по установке.

Этот wiki должен быть запущен и запущен, прежде чем мы займемся фермой. Он содержит весь код (файлы PHP), который управляет wiki. Другие wiki в ферме имеют только столько PHP-кода, чтобы отделить их друг от друга.

Установить ферму

Обзор терминологии

The farm metaphor maybe taking things too far but we'll be consistent and try to avoid the common computer documentation foible of mixing our metaphors.

- **Ферма**: Полная установка с одним родительским вики (фермер) и несколькими дочерними вики (животные)
- **Фермер**: Родительская вики, которая содержит большинство файлов PHP, управляющих Dokuwiki. Это также полноценная вики сама по себе. Что важно в настройке, так это то, что мы можем получить доступ как к отдельным модулям, используемым животным (см.

ниже), так и к основным модулям, используемым всеми вики.
- **Директория фермы**: Не путать с фермой, это корневой каталог всех дочерних вики. Это просто каталог в веб-пространстве.
- **Животное **: Помните, что разница между дочерней вики (животное) и родительской вики (фермер) заключается в том, какие файлы PHP хранятся в их каталоге. Каждое животное содержит минимальное количество файлов .php, которые используются для настройки dokuwiki, а также каталоги, содержащие все данные для этой вики, в то время как у фермера есть полная установка dokuwiki, которая используется всеми животными, а также его собственные каталоги для данных. Таким образом, только родительской вики нужно установить плагины, и только родительскую вики нужно обновить до новой версии.

Что видит браузер

Наше веб-пространство организовано следующим образом:

- `http://whatsamatta.edu` - домашняя страница
  - `http://whatsamatta.edu/wiki` - Родительская вики с материалами для потребления авторизованными пользователями, **фермером**.
  - `http://whatsamatta.edu/einstein` - Первая частная вики-статья профессора, посвященная **животным** (дочерняя вики-статья).
  - `http://whatsamatta.edu/keppler` - Частная вики второго профессора, еще одно **животное**.

Все остальное `http://whatsamatta.edu/somethingElse` не зависит от Dokuwiki.

Как организованы наши каталоги

Организация и расположение файлов конфигурации веб-сервера и контента являются, мягко говоря, гибкими. Соглашения значительно различаются, особенно если вы находитесь в общей или размещенной среде. Это описание нашей системы, не путайте его с рекомендацией того, как выглядит или должна выглядеть ваша система.

Конфигурация веб-сервера для этих вики содержится в /etc/httpd/conf.d/dokuwiki.conf. Мы создали этот файл.

Веб-контент находится в /var/www/html, а соответствующие каталоги:

```
├── wiki                <--The farmer (parent wiki)
│   ├── bin
│   ├── conf
│   ├── data                <-- These folders are what Dokuwiki uses
│   ├── inc
│   └── lib
└── wikis                   <--The farm (folder with child wikis)
    ├── _animal             <-- template for new child wiki
    ├── kepler              <-- animal (child wiki) #1
    ├── Kepler -> kepler    <--symbolic link for common errors
    ├── einstein            <-- animal #2
```

```
    └── Einstein -> einstein  <--symbolic link
```

Каталог _animal — это шаблон, доступный на странице фермы этот шаблон животных. Он принадлежит пользователю root с привилегиями 600, поэтому он недоступен для веб-сервера. Чтобы подготовить новое животное (дочернюю вики), введите:

```
cp -r _animal newton
chown -R apache:apache newton
ln -s newton Newton
```

Символические ссылки — это просто моя любимая мозоль. Я никогда не знаю, как люди будут писать имя с заглавной буквы, это делает newton и Newton похожими.

Для нашей системы каждый файл и каталог, показанные здесь (и их подкаталоги), принадлежат apache:apache и имеют разрешения 640 для файлов и 750 для каталогов. Обратите внимание, что имена пользователей и групп также значительно различаются в зависимости от установки.

Самое интересное: конфигурация

Следующие файлы изменены или созданы:

В каждом животном (дочерняя вики): (/var/www/html/wikis/einstein, /var/www/html/wikis/kepler...)

Редактировать `conf/local.protected.php` uncomment и установить `$conf['basedir']`

Например, измените:

```
  //$conf['basedir'] = '/farm/animal/';
```

на

```
  $conf['basedir'] = '/wikis/einstein/';
```

Я считаю, что завершающий символ / необходим, поэтому будьте осторожны.

Мы также изменили `conf/local.php` чтобы установить заголовок вики, чтобы помочь отладке, но это можно сделать через веб-интерфейс после того, как все заработает. Если вы хотите сделать это, измените:

```
  $conf['title'] = 'Animal Wiki Title';
```

на

```
  $conf['title'] = 'Albert\'s Wiki';
```

В каталоге фермера (/var/www/html/wiki)

Чтобы активировать ферму, перейдите в каталог inc в каталоге farmer (родительская вики).

Скопируйте файл `preload.php.dist` в `preload.php`. Обратите внимание, если вы решили работать со старой версией Dokuwiki, вам придется загрузить farm.php и preload.php.dist (см. страницу документации по фермам).

Раскомментируйте определение DOKU_FARMDIR и укажите путь. Изменить:

```
  //if(!defined('DOKU_FARMDIR')) define('DOKU_FARMDIR', '/var/www/farm');
```

в

```
  if(!defined('DOKU_FARMDIR')) define('DOKU_FARMDIR',
'/var/www/html/wikis');
```

Раскомментируйте модуль фермы:

```
  //include(fullpath(dirname(__FILE__)).'/farm.php');
```

Конфигурация Apache

Во время установки Dokuwiki мы создали файл конфигурации /etc/httpd/conf.d/dokuwiki.conf с несколькими директивами <Directory>, которые управляют доступом к различным каталогам. (Я ищу ссылку на инструкции о том, как это сделать).

Теперь нам нужно определить псевдонимы, необходимые для сокращения URL-адресов, которые будут вводить наши пользователи, и добавить правила перезаписи.

Псевдонимы

В текущем виде пользователь может ввести URL-адрес, который выглядит так, `http://wattsamatta.edu/wikis/einstein` чтобы попасть на нужное животное на ферме (дочерняя вики), однако тот факт, что мы решили назвать наши вики-сайты фермы, не имеет значения, поэтому, чтобы сократить его до, `http://wattsamata.edu/enistein` мы включаем директиву псевдонима для каждой дочерней вики:

```
Псевдоним /Эйнштейн /var/www/html/wikis/kepler
Псевдоним /Кеплер /var/www/html/wikis/kepler
```

Переписать правила

В документации ферм представлены два различных метода адресации ферм: виртуальные хосты и .htaccess. При использовании метода виртуального хоста (который мы не используем) пользователь вводит URL-адрес вида `http://einstein.whatsamatta.edu`. При

использовании метода .htaccess (описанного ниже) этот URL-адрес будет выглядеть так : . Выбор обычно определяется тем, к каким инструментам у вас есть доступ. Мы выбрали метод .htaccess, потому что нам нелегко добавлять записи CNAME на DNS-`http://whatsamatta.edu/einstein`. сервер университета.

Это единственная действительно, действительно таинственная и непонятная часть. То, что мы хотим, чтобы наши пользователи делали, это входили `http://whatsamatta.edu/einstein`, а то, что нужно dokuwiki, больше похоже на то, `http://whatsamatta.edu/wiki/doku.php?animal=einstein` за исключением, конечно, когда это не домашняя страница вики, тогда uri намного сложнее.

Это делается с помощью правил перезаписи, которые Apache использует для преобразования фактического URL в тот, который нам нужен. Это регулярное выражение (что само по себе плохо) с дополнительными флагами. По большей части вы можете скопировать эти правила с небольшими, очевидными изменениями. Если вы хотите лучше их понять, я нашел этот шведский сайт с хорошим способом поиграться с правилами mod_rewrite, см. http://martinmelin.se/rewrite-rule-tester/ (так скажем, Мартин).

Инструкции фермы делают это с помощью файла .htaccess, но мы предпочитаем помещать их в директиву <Directory>. Оба варианта работают, Apache рекомендует директиву из соображений производительности, но метод .htaccess работает для людей без привилегий root. Добавьте следующее в файл dokuwiki.conf:

```
<Directory /var/www/html/wikis>
      AllowOverride All
      Options +FollowSymLinks

      RewriteEngine On

      RewriteRule ^/?([^/]+)/(.*)  /wiki/$2?animal=$1 [QSA]
      RewriteRule ^/?([^/]+)$      /wiki/?animal=$1 [QSA]

</Directory>
```

На этом этапе, если никто из нас ничего не упустил, перезапуск веб-сервера создаст рабочую вики-ферму. Введите:

```
service httpd restart
```

затем попробуйте сделать это через браузер.

Удачи.

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

Permanent link:
https://wwoss.ru/doku.php?id=wiki:farms:example01

Last update: **2025/01/15 21:28**