

コンテンツ管理システム PukiWiki の導入と活用事例

雨谷恵

筑波大学システム情報工学等支援室(情報システム管理班)

〒305-8573 茨城県つくば市天王台 1-1-1

概要

最近では、ほとんどの業務がネットワークにつながれた PC を利用していると思われる。そのような環境で、すばやくお互いの情報を共有する為には、ネットワーク上で共有ツールを使用するのが便利である。しかし、専門的な知識を持たないユーザーが共有ツールを利用するだけでなく、導入するには高度な知識が必要と思われるかもしれない。確かに適当なソフトの選定、インストールとその後の設定等の手間、また、組織によっては CGI の使用規制、コストの問題等で、なかなか気軽には導入に踏み切れないのではないだろうか？

おそらく多くの組織では、すでに情報共有ツールとして fml¹等のメーリングリスト(以下MLと略)プログラムを利用していると思われるが、毎日大量に送られてくるメールをひとつひとつチェックするのも大変な労力である。しかも、例えば論議の途中からそのMLのメンバーに加わった場合、今なされている論議の経緯を知るためには、過去に配信されたメールを取り寄せ読まなくてはならない。多くのMLプログラムの機能には、記事の取り寄せをすることは可能だが、その中から必要な情報を検索し取り出す事は出来ない。

それらの問題点に対し簡単にすばやく「情報共有」を実現するためのツールとして PukiWiki を実例を交えて紹介する。

1. はじめに

初めて PukiWiki を導入したのは、2004 年 4 月に筑波大学システム情報工学研究科総合研究棟 B が完成した時である。建物の完成とともに一部の研究室/実



図 1 新築ワーキンググループ TOP画面

験室を移転するため、移転対象となる同研究科コンピュータサイエンス専攻では、早急にネットワークの整備等を検討する為にワーキンググループが発足し

た。このワーキンググループの情報共有の場として PukiWiki を採用し、掲示板的な利用の他、議事録等各資料の保管を行った。

同じようなシステムは、以前にも同大学情報学類計算機運用委員会内、学生管理者グループで、グループ内の情報共有のため Tiki という PukiWiki と同じシステムを利用していた。ただし、Tiki はオブジェクト指向スクリプト言語である Ruby 言語²で書かれており、CGI モードで動かすことを前提としているため、導入には基本的な CGI の設定の知識とともに、Ruby 言語の知識が必要な場合もある。PukiWiki は PHP 言語³で書かれており PHP モジュールを Apache サーバーに組み込むことで Perl/CGI と比較して処理速度の高速化、また、サーバーの負荷を低減することができる。さらに CGI を利用しないため動作するため設置の面でも安易である。現在、これらのシステム WikiEngine を利用したプログラムは多数存在し、Wiki クローンと呼ばれている。

本稿では PukiWiki の導入方法、初期設定、また導入時の注意点を報告する。

2. PukiWiki のアウトライン

PukiWiki の「Wiki」とは自由に拡張可能な、連結されたウェブ「ページ」の集積で、情報を格納し、修正するためのハイパーリンクシステム WikiEngine のことである。この WikiEngine を利用することにより、web ブラウザを利用し編集をすることが可能であり、各ページが誰にでも容易に編集可能なデータベースとしても利用できる。また、一からシステムを構築することなく、既存の web サーバープログラムを利用することで、インストールから動作するまでとても手軽であるのが特徴のひとつである。

PukiWiki のインフラは WWW とサーバー・クライアントアプリケーションに支えられているとも言える。その基盤となる HTTP プロトコルは、クライアント・サーバー間の通信がどのように行われているかを規定しているが、Wiki は、web 上で使用されている HTTP プロトコルの GET(データ要求)処理と POST(データ提出要求)処理を行っている。

PukiWiki に使われている PHP 言語は、スクリプト言語のひとつである。大きな特徴は HTML ファイル内に直接記述することができる点である。また、CGI としても使用できるが、利用の多くは Apache 等の web サーバーに PHP モジュールを組み込まれた環境で使用されている。また、Perl/CGI と比較して処理速度の高速化、サーバー負荷の低減が可能で各種データベースとの連携に優れている。その他 PHP 言語は、XML, PDF, IMAP, LDAP 等各種機能をサポートしており、広範な web アプリケーションを容易かつ

¹ <http://www.fml.org/>

² <http://www.ruby-lang.org/ja/>

³ <http://www.php.net/>

柔軟に作成可能な点が魅力で、最近急速に普及している言語のひとつである。

3. 動作環境の確認

PukiWiki は日本語マルチバイト対応の PHP (PHP 4.1.0 以降) と web サーバーを利用できる環境なら基本的にどの OS 上でも動作する。ただし、PukiWiki の設置は public_html 等、web でアクセスできる場所におく必要がある。web サーバーに php モジュールが組み込まれているかを確認するには、その web サーバーの管理者に問い合わせるか、もしくは自分で以下のサンプルスクリプトを適当なファイル名(例では info.php)で作成し、このファイルをブラウザで表示して確認できる。ブラウザで表示して図 2 のように表示されれば、PukiWiki は設置可能である。

サンプルスクリプト(info.php) の内容

```
<html>
<head>
<title>PHP TEST</title>
</head>
<body>
<?php phpinfo(); ?>
</body>
</html>
```



図 2 サンプルスクリプト(info.php) 表示例

このサンプルスクリプトで使用されている phpinfo 関数は、PHP の状態に関する情報を出力する。出力される情報は PHP コンパイルオプションと拡張機能、PHP のバージョン、サーバー情報と環境(モジュールとしてコンパイルされた場合)、PHP の環境、OS バージョン情報、パス、構成オプションのマスター およびローカルの値、HTTP ヘッダ、PHP License 等である。

4. インストール手順

インストールの為に必要なパッケージはPukiWiki 公式サイト⁴からダウンロードすることができる。2005/01/25 現在のPukiWikiの最新バージョンは 1.4 系(XHTML 1.1) はPukiWiki1.4.4、1.3 系(HTML 4.01 transitional)は PukiWiki1.3.7 である。

⁴ <http://www.pukiwiki.org/>

以降、インストール手順を説明するにあたり、Linux 環境で、PukiWiki ファイル群を設置した場所を /home/amagai/public_html/pukiwiki

URL を <http://localhost/~amagai/pukiwiki/> として進める。

公式サイトから提供されるパッケージは ZIP 形式と tar 形式があるので好みの形式をダウンロードする。tar 形式パッケージ(pukiwiki-1.4.4.tar.gz)の場合、以下のように p オプションをつけると、設定されたアクセス権で展開されるので、すぐに web 上で利用できる。

```
%tar pzxvf pukiwiki-1.4.4.tar.gz
%mv -Rf pukiWiki-1.4.4 ~amagai/public_html/pukiwiki
```

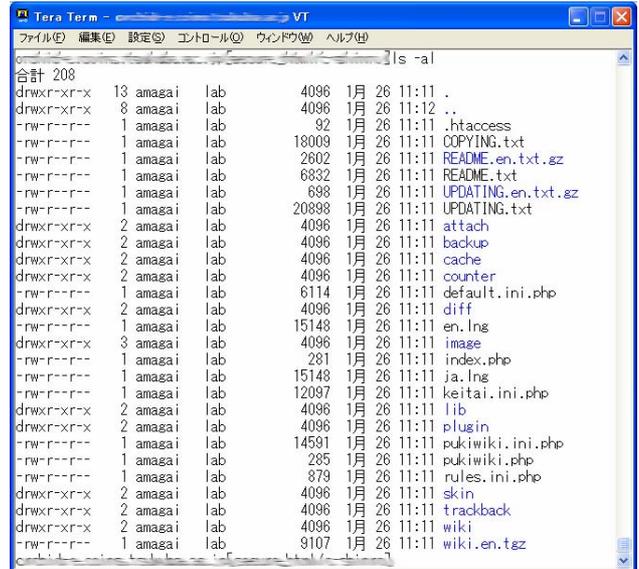


図 3 展開された PukiWiki パッケージ

なお、正しいアクセス権を確認するにはパッケージ内にある README.txt ファイルを参考にされたい。これら展開されたファイル群のアクセス権が正しく設定されていれば、web ブラウザで <http://localhost/~amagai/pukiwiki/index.php> へアクセスをすると、図 4 のように PukiWiki のフロントページが表示されるはずである。これで基本的な設置は完了である。ただし実際に PukiWiki を運用する際は、



図 4 PukiWiki 初期画面

PukiWiki 設定ファイル pukiwiki.ini.php の設定と後述 7. を参考に、セキュリティ対策を行う必要がある。

5. 初期設定時に注意する点

pukiwiki-1.4.4 からは、配布パッケージを展開した時、各ディレクトリ内にあらかじめ「.htaccess」ファイルが存在している。.htaccess ファイルとは、web サーバプログラム Apache の挙動を決定する設定ファイルの一つであるが、使用する web サーバの運用方針によっては設置を許可しない設定にしている場合もある。この場合、.htaccess ファイルがあると "Internal Server Error" を起こすことがある。その場合は pukiwiki/.htaccess、および pukiwiki ディレクトリ以下にあるすべての .htaccess ファイルを削除することでエラーを回避できる。また、"Runtime Error" が表示される場合は、ファイルのアクセス権や、それらを収めているディレクトリのアクセス権が適切でない可能性があるため、README.txt ファイルで確認をする。

6. 基本的な機能の紹介

基本的なメニューは図 5 にあるようにページの上部にメニューが表示される。[新規] ページの作成と [編集] について簡単に説明する。

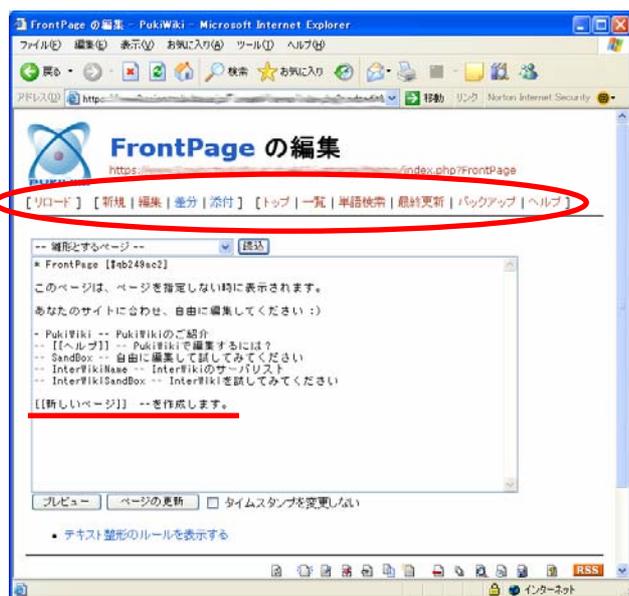


図 5 PukiWiki の編集画面

新規ページを作成する方法として、すでに作成済みのページの [編集] 画面において、図 5 のとおり、既存のページ (図 5 では『FrontPage』) に BracketName ([[]]) で囲まれた文字) を入力する方法がある。図 5 では『[[新しいページ]] --をを作成します。』と入力されている。この状態で [ページの更新] をすると、図 6 のとおり『FrontPage』に『新しいページ --をを作成します。』の後ろに ? というリンクがつく。そこをクリックすると『新しいページ』が作成され、編集画面になる。次に図 7 のとおり「新たに作成されました」と入力し、[ページの更新] すると『新しいページ』に文章が追加され図 8 のとおり更新される。またメニュー [新規] から新規ページを作成できる。なお、PukiWiki に用意された基本メニューはリスト 1 のとおりである



図 6 新しいページへのリンク

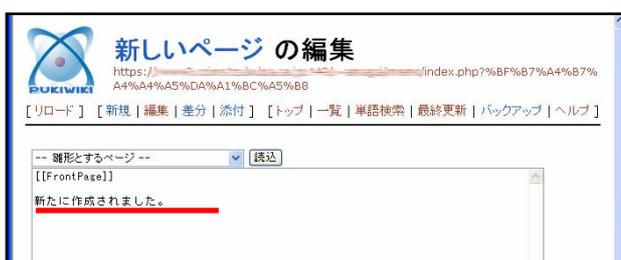


図 7 新しいページの編集画面

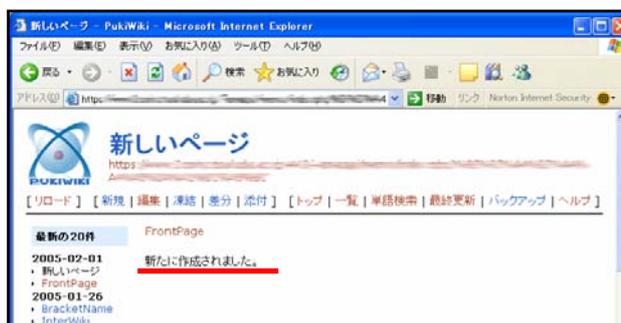


図 8 作成した新しいページ

- [新規] 新しいページを作成する
- [編集] ページを編集する
- [凍結] ページを pukiwiki.ini.php で設定したパスワードで書き込み不可にする
- [凍結解除] ページを pukiwiki.ini.php で設定したパスワードで書き込み許可にする
- [差分] ページの更新の差分
- [添付] ページにファイルを添付するプラグイン
- [一覧] ページの一覧
- [単語検索] 全てのページから単語を検索
- [最終更新] 各ページの最終更新一覧
- [バックアップ] 過去のバックアップ一覧
- [ヘルプ] PukiWiki の使い方等

リスト 1 PukiWiki の基本メニュー

7. アクセス制限とセキュリティ対策

PukiWiki はその性質上、誰でもどこからでもアクセスできるため、無断で消去、改竄される可能性がある。定期的にバックアップを取る機能があり、完全ではないが復旧することもできるが、それだけでは不十分である。よって初期状態のまま PukiWiki

の運用は危険である。そこで、別途セキュリティ対策が必要となる。PukiWiki 自体が提供している機能とともに Apache で提供されている機能も利用する。

7.1 PukiWiki の「凍結機能」

先にも書いたとおり、PukiWiki はアクセスしてきたすべてのユーザーに、ページ編集や新規ページの作成を許可している。そこで、PukiWiki の「凍結」機能を利用し、予め設定したパスワードで、書き込みを制限し、セキュリティを強化する。

PukiWiki の凍結機能は予め MD5 という方式で暗号化したパスワードを設定ファイル `pukiwiki.ini.php` に書き込むことで実現する。このファイル内の `$adminpass` 以降の (シングルコーテーション) で囲われた間に MD5 で暗号化された値を以下のように記入すればよい。

`pukiwiki.ini.php` のパスワード設定部分

```
// 管理者パスワード
// 以下は md5('pass') の出力結果です
$adminpass = '329435e5e66be809a656af105f42401e';
```

MD5 での暗号化は、Linux 上では、以下のように簡単に計算させる事ができる。

```
% echo -n 'pass' | md5sum
329435e5e66be809a656af105f42401e -
```

この設定により、パスワードを知るユーザのみが変更作業を行うことができ、改竄されにくくなる。

7.2 Apache を利用したアクセス制限

webサーバApache⁵を利用し、`.htaccess`ファイルを活用することにより、より多彩なアクセス制限を行うことが可能になる。例えば、特定のホスト (IPアドレス、ドメイン名等) からのアクセスを制限、IDとパスワードによるアクセス制限 (Basic認証) 等が実現可能である。例えば、Basic認証を利用する場合は `pukiwiki/.htaccess` に以下の行を追加する。

```
AuthUserFile /home/amagai/.htpasswd
AuthGroupFile /dev/null
AuthName "Please enter your ID and password"
AuthType Basic
require valid-user
```

1行目の `AuthUserFile` では認証に使われる ID とパスワードが書かれているファイル `.htpasswd` の場所を指定する。通常この `.htpasswd` ファイルは web でアクセスできない場所 (例 `/home/amagai/.htpasswd`) に設置する。この `.htpasswd` の中身は以下のとおり「ID: 暗号化されたパスワード」の形式で記述されている。パスワードの暗号化をする場合、PHP 言語の `crypt` 関数を使う事で暗号化が可能だが、フリーウェア等で提供されている CGI (perl) や PHP で書かれた専用のプログラムを利用しても暗号化することが出来る。

`/home/amagai/.htpasswd` の例

```
amagai:fjeio29f32Ein
user1:342bieEezjVie
```

この設定により

`http://localhost/~amagai/pukiwiki/index.php` にアクセスした場合、図 10 のような Basic 認証が現れ、`.htpasswd` に登録されたユーザーが任意のパスワードでのみ利用することが可能となる。



図 10 認証画面

7.3 .htaccess 設定の応用例

7.1 と 7.2 で説明した機能を利用する場合の問題点は、PukiWiki の凍結機能を利用した場合、万が一設定したパスワードが流出した時、共通パスワードのため、誰が使用したかの特定が難しい。また、`.htpasswd` と `.htaccess` を方法の場合は `.htpasswd` を作成する際、DES によるパスワードの暗号化に多少専門知識が必要となる。

同大学情報学類⁶では、NISのパスワードを、Apache に組み込まれた認証モジュールが読めるように変換するプログラムを開発し、その出力ファイル (`webpass.coins`) を各ユーザが `.htaccess` に利用できるサービスを提供している。このサービスにより別途 web アクセス用のパスワードファイル `.htpasswd` を設定することなく、ユーザはシステムから提供される暗号化された NIS のパスワード情報を安全に設定することが可能となる。情報学類で提供している `.htaccess` のサンプルは以下のとおりであるが、このように記述した `.htaccess` ファイルを用意することで、NIS 情報を利用した認証が可能となる。

COINS の全ユーザにアクセスを許可する

```
AuthType Basic
AuthDBUserFile /etc/webpass.coins
AuthName "Please input your coins's Account
& Password"
Require valid-user
```

⁵ <http://www.apache.org/>

⁶ <http://www.coins.tsukuba.ac.jp>

COINS のユーザ user1,user2 にアクセスを許可する

```
AuthType Basic
AuthDBUserFile /etc/webpass.coins
AuthName "Please input your coins's Account
& Password"
Require user user1 user2
```

COINS のグループ ugrad(情報学類生全員)に属するユーザにアクセスを許可する

```
AuthType Basic
AuthDBGroupFile /etc/webpass.coins
AuthDBUserFile /etc/webpass.coins
AuthName "Please input your coins's Account
& Password"
Require group ugrad
```

ここで指定されている webpass.coins は cron を利用し一定時間ごとに、開発したスクリプトによりログインアカウント、パスワード、グループ等の情報を取得している。

7.4 SSL の導入

アクセス制限を行っても、ネットワーク上で流れているデータは平文のままなので、常に脅威に晒されている。大切なデータを「悪意を持った第三者」から見られても平気なように、データを変換する方法(暗号化)を行うことが必要である。その暗号化されたデータのやり取りで、最も一般的な方式は



図 11



図 12 SSL 証明書

SSL(Secure Socket Layer)通信と呼ばれる通信方式である。この SSL 通信とアクセス制限を併用して利用することにより、よりセキュアな運用が可能となる。

SSL 通信を利用しているかどうかは、ブラウザで確認することができる。SSL で通信している場合、図 11 のとおりブラウザ画面右下のとおり鍵マークが表示される。この鍵マークをクリックすると図 12 のようにその web サーバーが使っている証明書の情報を見ることができ、安全性を確認できる。

まとめ

PukiWiki が最も力を発揮するのはコラボレーション(共同作業)に利用される場合である。例えば、会議時に PC とプロジェクトを用いて、リアルタイムで議事録等の編集したり過去の議事を確認しつつ進行できる。追加資料等ファイルのアップデートも容易である。

PukiWiki の設置は、特別な知識や管理者権限が必要ではなく、手軽で便利な情報共有ツールとして、エンドユーザのコンピュータリテラシの向上とともに、ますます広がっていくと予想される。また、PukiWiki には標準添付のプラグインの他、多数の自作プラグインが公開されているので、利用者それぞれがニーズにあったカスタマイズをすることで、より威力を発揮するのでぜひ一度利用されたい。

ただし、PukiWiki は web サーバーに設置して利用することが前提となっている為、そのような作業に慣れていないユーザにとっては敷居が高いかもしれない。最近、ローカル PC 上で Wiki の機能が使える「ひとり Wiki」⁷ というフリーソフトが提供されているので、web サーバに導入する事に不安を感じるユーザーには、まず「ひとり Wiki」で Wiki の使用感を試されてはいかがだろうか。

参考文献

- [1] 『Wiki Way コラボレーションツール Wiki』サポートページ <http://www.yamdas.org/wikiway/index.html>
- [2] PukiWiki の公式サイト <http://www.PukiWiki.org/>
- [3] PukiWiki 開発サイト <http://PukiWiki.org/dev/>
- [4] PHP 本家(英語) <http://www.php.net/>
- [5] coins のアカウントを用い、個人認証とアクセス制御により、学類関係者向けのページを公開する方法 <http://www.coins.tsukuba.ac.jp/ce/web-auth/>

⁷ 窓の杜の「ひとり Wiki」紹介文

<http://www.forest.impress.co.jp/article/2005/01/06/hitoriwiki.html>