Aspects of PmWiki security are found on the following pages:

Pages distributed in a PmWiki release:

- Passwords General use of passwords and login
- Passwords Admin More password options for the administrator
- AuthUser Authorization system that uses usernames and passwords
- Url Approvals Require approval of Url links
- Site Analyzer
- Blocklist Blocking IP addresses, phrases, and expressions to counteract spam and vandalism.
- Notify Allows a site administrator to configure PmWiki to send email messages whenever pages are changed on the wiki site
- Security variables variables crucial for site security

Cookbook Pages

- Cookbook index for Security recipes
- Cookbook:HtpasswdForm Form based management of users and passwords using .htpasswd/.htgroup files
- Cookbook:Secure attachments Protecting uploaded attachments
- Cookbook:Web server security Making the server more secure with .htaccess
- Cookbook:Farm security Making Farm installations secure
- Cookbook:DeObMail Hide e-mail address
- Cookbook:Protect email Obfusticate email addresses
- Cookbook:Audit images Check to see what images have been uploaded to your wiki.
- Cookbook:Private groups Create and secure private groups on a public wiki
- Cookbook:Only one login Only allow 1 login at the same time for a username
- Cookbook:Session guard Protects against Session Theft

How do I report a possible security vulnerability of PmWiki?

Pm wrote about this in a post to pmwiki-users from September 2006. In a nutshell he differentiates two cases:

1. The possible vulnerability isn't already known publicly: In this case please contact Pm by private mail.
2. The possible vulnerability is already known publicly: In this case feel free to discuss the vulnerability in public (e.g. on pmwiki-users).

See his post mentioned above for details and rationals.

What about the botnet security advisory at http://isc.sans.org/diary.php?storyid=1672?

Sites that are running with PHP's *register_globals* setting set to "On" and versions of PmWiki prior to 2.1.21 may be vulnerable to a botnet exploit that is taking advantage of a bug in PHP. The vulnerability can be closed by turning *register_globals* off, upgrading to PmWiki 2.1.21 or later, or upgrading to PHP versions 4.4.3 or 5.1.4.
In addition, there is a test at PmWiki:SiteAnalyzer that can be used to determine if your site is vulnerable.

Wiki Vandalism and Spam

 Copyright: 2024

Assumptions

you are using a Blocklist and Url approvals.

You don't want to resort to password protecting the entire wiki, that's not the point after all.

Ideally these protections will be invoked in `config.php`

How do I stop pages being deleted, eg password protect a page from deletion?

Use Cookbook:DeleteAction and password protect the page deletion action by adding `$DefaultPasswords['delete'] = '*';` to `config.php` or password protect the action with `$HandleAuth['delete'] = 'edit';`

or `$HandleAuth['delete'] = 'admin';` to require the edit or admin password respectively.

How do I stop pages being replaced with an empty (all spaces) page?

Add `block: /^\s*$/` to your blocklist.

how do I stop pages being completely replaced by an inane comment such as *excellent site*, *great information*, where the content cannot be blocked?

Try using the newer automatic blocklists that pull information and IP addresses about known wiki defacers.

(OR) Try using Cookbook:Captchas or Cookbook:Captcha (note these are different).

(OR) Set an edit password, but make it publicly available on the Site.AuthForm template.

How do I password protect all common pages in all groups such as recent changes, search, group header, group footer, and so on?

Insert the following lines into your local/config.php file. Editing these pages then requires the admin password.

```
## Require admin password to edit RecentChanges (etc.) pages.
if ($action=='edit'
    && preg_match('/\\.(Search|Group(Header|Footer)|(All)?RecentChanges)$/', $pagename))
  { $DefaultPasswords['edit'] = crypt('secret phrase'); }
```

Note that all GroupAttributes pages are protected by the attr password.

Alternative: you can require 'admin' authentication for these pages:

```
## Require admin password to edit RecentChanges (etc.) pages.
if ($action=='edit'
    && preg_match('(Search|Group(Header|Footer)|(All)?RecentChanges)', $pagename))
  { $HandleAuth['edit'] = 'admin'; }
```

How do I password protect the creation of new groups?

 Copyright: 2024

See Cookbook:Limit Wiki Groups

How do I password protect the creation of new pages?

See Cookbook:Limit new pages in Wiki Groups

How do I take a whitelist approach where users from known or trusted IP addresses can edit, and others require a password?

Put these lines to local/config.php:

```
## Allow passwordless editing from own turf, pass for others.
if ($action=='edit'
 && !preg_match("/^90\\.68\\./", $_SERVER['REMOTE_ADDR']) )
 { $DefaultPasswords['edit'] = crypt('foobar'); }
```

Replace 90.68. with the preferred network prefix and foobar with the default password for others.

How do I password protect page actions?

See Passwords for setting in config.php

```
$HandleAuth['pageactionname'] = 'pageactionname'; # along with :
$DefaultPasswords['pageactionname'] = crypt('secret phrase');
```

or

```
$HandleAuth['pageactionname'] = 'anotherpageactionname';
```

How to make a rule that allows only authors to edit their own wiki page in Profiles group?

Add this to your *local/config.php*

```
$name = PageVar($pagename, '$Name');
$group = PageVar($pagename, '$Group');
if($group=='Profiles') $DefaultPasswords['edit'] = 'id:'.$name;
```

How do I moderate all postings?

Enable PmWiki.Drafts

- Set $EnableDrafts, this relabels the "Save" button to "Publish" and a "Save draft" button appears.
- Set $EnablePublish, this adds a new "publish" authorization level to distinguish editing from publishing.

How do I make a read only wiki?

In config.php set an "edit" password.

How do I restrict access to uploaded attachments?

See

- instructions for denying public access to the uploads directory
- see Cookbook:Secure attachments

 Copyright: 2024

Copyright: Peter Bowers - 24/02/2011