Wish to encrypt contents of wiki pages in database | Tiki Wiki CMS Groupware :: Development
Wish to encrypt contents of wiki pages in database

Status
● Open

Subject
Wish to encrypt contents of wiki pages in database

Category
- Feature request
- Documentation (or Advocacy)

Resolution status
New

Submitted by
Alan

Lastmod by
Alan, Jean-Marc Libs, luciash d' being ?

Rating
★★½★★★★½½★★★★½½★★★½½★★★★½★★ (1)

Description
I would like to be able to encrypt the text of pages before writing them to the database, and decrypt for searches or display to the user.
It would be cool if the same function could be applied to the text of tags, categories, articles, or other features, but being able to use it for wiki articles is my main concern.

Update 31 May 2018: I found out how to encrypt entire databases through the MariaDB documentation. https://mariadb.com/kb/en/library/encryption-key-management/

With this in mind, I would like to propose a page be created in Tiki documentation that briefly overviews encryption options for Tiki installations (For example, by linking to that MariaDB article if one is using MariaDB). Perhaps a process to automate the commands required to encrypt a database could be added at some future point (?), that users could opt into when installing Tiki?

Workaround
I encrypted my entire database using MariaDB's documentation on database encryption as a guide. https://mariadb.com/kb/en/library/encryption-key-management/

Importance
3

Easy to solve?
7

Priority
21

Demonstrate Bug (Tiki 19+)

Please demonstrate your bug on show2.tiki.org

Version: trunk ▼

Ticket ID
6671

Created
Friday 25 May, 2018 20:15:06 GMT-0000
by Alan

LastModif
Friday 08 June, 2018 09:13:37 GMT-0000

Comments

Jean-Marc Libs 07 Jun 18 12:11 GMT-0000

This was categorized in "security". That's usually for security flaws reports and so it was restricted access to members of the security team.
In this case, it's a general discussion, so I removed the *security* category.

Alan 07 Jun 18 22:17 GMT-0000

Thank you for recategorizing.

Jean-Marc Libs 07 Jun 18 12:14 GMT-0000

I feel encrypting the whole database at the mariadb level is the only way of not losing access to searching.

Anyone who writes a documentation page on this topic should mention that using elastic search (ES) indexing creates a duplicate of most of the data in ES. So, obviously in this case encrypting the database is not enough.

luciash d' being ? 08 Jun 18 09:11 GMT-0000

Just thinking out loud... would a PluginEncrypt do the job?

Marc Laporte 08 Jun 18 13:04 GMT-0000

I support development of a way to encrypt snippets of text in Tiki. I envisage something like https://www.mailvelope.com/ for many data types in Tiki, especially wiki pages, part of wiki pages (PluginEncrypt), files, and tracker fields.

Some ideas:

- https://suite.tiki.org/KeePass

drsassafras 09 Jun 18 02:42 GMT-0000

It looks like the keypass encryption and encrypting an entire database through MariaDB have 2 very different use cases. User level encryption means that that user is the only one who can access the information and that even a sysadmin would not be able to decrypt it. (or maybe only with a sysadmin password could it be decrypted, depending on how it's set up) While encrypting an entire database through MariaDB does not protect user-level data at all. So a sysadmin could easily access all information stored, or user information could be leaked through a configuration error, or any kind of hack, or to anyone with file read access to the live system. The MariaDB encryption would only protect against threats that already have file access (to the stored database) but at the same time can't get there hands on the key, either through the MariaDB configuration or through the stored key & password, so the server would need to have permissions set up to only allow partial read access and the attacker would obviously need to be unable to bypass this.

The problem with the MariaDB encryption is that it's hard to imagine that someone who already has read access to the encrypted database would not also be able to read the MariaDB configuration. (why not just set permissions so they don't have read access to the database)The best use case I can think of is backups, where if one has an off-site backup it's not being stored in plain text. So at that point who ever is in charge of the off-site backups can't gain access at that point. Assuming the key is not backed up with the data. But perhaps encrypting the backups would be a better solution for this use case, as it means you won't take a performance hit on your website... Although I guess if it's a low volume site this doesn't really matter.

In any case, there may be a few use cases, but seems to provide little to no protection under most common cases of unwanted data access.

Marc, your proposed solution could provide protection from, while; just about everything! Down side is that it's not implemented and that indexing the data could be tricky. Although if the index was encrypted under the same key as the data, it perhaps could be done. Performance would also take much less of a hit because only data that needs to be encrypted is.

Just a few thoughts.

Marc Laporte 10 Jun 18 13:18 GMT-0000

Encrypted backups are a good idea but they should be checked and tested on a schedule.

Attachments

	filename	created	hits	comment	version	filetype	

No attachments for this item