Vault Encryption

Roël Couwenberg - PixNyb StripeCon 2023 - Hamburg

Following the talk I've had the pleasure of preparing on Vault Encryption at StripeCon 2023, I've collected some resources to help those interested learn more about the topic.

Vault Encryption

- Vault - A tool for secrets management, encryption as a service, and privileged access management.
- Vault Transit Engine - A secrets engine for Vault that handles cryptographic functions on data in-transit.

Vault Encryption in SilverStripe

- Vault Module - A SilverStripe module that integrates Vault into the SilverStripe CMS using the Transit Engine and a custom Encrypted DB field. *The module is currently unreleased and private. Keep this link handy for when it is released.*

Other Resources

- Blind Indexes in 3 minutes: Making Encrypted Personal Data Searchable - A blog post by Joshua Kelly on how to use blind indexes to search encrypted data.

Contact

If you have any questions, feel free to contact me through the following channels:

- Instagram - @roelc.me
- Discord - pixnyb
- GitHub - @pixnyb
- Email - contact@roelc.me

SilverStripe Vault Module

This module provides a way to store sensitive data securely using the Vault service (specifically the Transit API).

Requirements

- SilverStripe 4.0+
- PHP >= 7.4, >= 8.0
- Vault Server with Transit API enabled

Installation

Install the module using composer by adding the module repository and a GitHub OAuth token with repo access to your `composer.json` file.

```
{
    "repositories": [
        {
            "type": "vcs",
            "url": "https://github.com/Violet88github/silverstripe-vault"
        }
    ],
    "config": {
        "github-oauth": {
            "github.com": "1234567890abcdef1234567890abcdef12345678" //
Github OAuth token with repo access
        }
    }
}
```

Then install the module using composer.

```
composer require violet88/silverstripe-vault
```

Configuration

Vault

The module requires transit to be enabled on the Vault server. The following policy can be used to enable transit.

```
path "transit/*" {
    capabilities = ["create", "read", "update", "delete", "list", "sudo"]
}
```

The transit engine can be enabled using the following command.

```
vault secrets enable transit
```

SilverStripe

Configuration File

The module requires a Vault server to be configured. The server can be configured in the `vault.yml` file.

```
---
name: vault
---
Violet88/VaultModule/VaultClient:
    vault_token:            # Vault Authorization Token
    vault_url:              # Vault URL
    vault_transit_path:     # Transit Path, defaults to 'transit'
```

Additionally, a default key can be configured in the `vault.yml` file.

```
Violet88/VaultModule/VaultKey:
    name: # Key Name
    type: # Key Type, e.g. aes256-gcm96
```

If no key is configured, the module will use the following defaults.

```
Violet88/VaultModule/VaultKey:
    name: 'silverstripe'
    type: 'aes256-gcm96'
```

Keys will be created automatically if they do not exist, be sure to set Vault permissions accordingly.

Environment Variables

Along with the `vault.yml` file, the module supports the following environment variables.

```
VAULT_TOKEN="s.1234567890abcdef"
VAULT_URL="https://vault.example.com"
VAULT_TRANSIT_PATH="transit"
```

Setting these environment variables will override the corresponding values set in the `vault.yml` file.

Usage

The module provides an `Encrypted` field type that automatically encrypts and decrypts data when it is saved and retrieved from the database.

```
<?php

class MyDataObject extends DataObject
{
    private static $db = [
        'MyEncryptedField' => 'Encrypted',
    ];
}
```

The datatype supports automatic casting, to use it simply pass the cast type as well as any of it's parameters.

```
<?php

class MyDataObject extends DataObject
{
    private static $db = [
        'MyEncryptedIntegerField' => 'Encrypted("Int")',
        'MyEncryptedEnumField' => 'Encrypted("Enum",
"value1,value2,value3")',
    ];
}
```

Filtering

The module provides an `EncryptedSearch` that can be used to filter data by encrypted fields. Keep in mind that the filter will only return exact matches.

```
<?php

class MyDataObject extends DataObject
{
    private static $searchable_fields = [
        'MyEncryptedField' => 'EncryptedSearch',
    ];
}
```

Tasks

The module provides tasks for encrypting and decrypting all data and rotating the default key.

```
# Encrypt all data
vendor/bin/sake dev/tasks/EncryptDBTask
```

```
# Decrypt all data
vendor/bin/sake dev/tasks/DecryptDBTask
```

```
# Rotate keys
vendor/bin/sake dev/tasks/RotateKeyTask
```