CONFIGURATION SERVICE

PROJECT GOAL

Create an API accessible Service that provides that ability to retrieve and store configuration data.

PROJECT REQUIREMENTS

zendesk.atlassian.net/wiki/display/ops/Configuration+S

- Service is API accessible
- A Ruby library to interface with the API
- Data storage technology must:
  - Be abstracted out and not directly accessed
  - Handle replication of data

- The API:
  - Is well documented
  - Requires an API token to use
  - Supports namespacing / separation of data
  - Supports API tokens:
    - To be matched with namespaces
    - That are read, write or read/write
  - Writes data as key -> value pairs
  - Returns data in JSON format

IMPLEMENTATION

- Ruby library
- Command line tool (Rubygem)
- A token is required for access
- ACL's applied to restrict token access
- Supports Hashicorp Consul & Hashicorp Vault
- Consistent adapter interface

RUBY LIBRARY

Basic usage

```
include Zendesk::ConfigurationService

cs = Adapters::Consul.new(
  address: "http://127.0.0.1:8500",
  token:   "2701E633-9EB5-4E38-B190-0CC809CDFC64"
)

cs.read('people/credentials')
=> {"username"=>"admin", "password"=>"woo-ahhh"}

credentials = { username: "admin", password: "whoah" }
cs.write('people/credentials', credentials)
=> true

cs.read('people/credentials')
=> {"username"=>"admin", "password"=>"whoah"}
```

Read/write outside of permitted space

```
include Zendesk::ConfigurationService

cs = Adapters::Consul.new(
  address: "http://127.0.0.1:8500",
  token:   "2701E633-9EB5-4E38-B190-0CC809CDFC64"
)

cs.read('embeddable/credentials')
Zendesk::ConfigurationService::Exceptions::NotFound: Zendesk::C
--snip--

cs.write('embeddable/credentials', {})
Zendesk::ConfigurationService::Exceptions::Forbidden: Zendesk::
--snip--
```

Service unavailable

```
include Zendesk::ConfigurationService

cs = Adapters::Consul.new(
  address: "http://127.0.0.1:8500",
  token:   "2701E633-9EB5-4E38-B190-0CC809CDFC64"
)

cs.read('people/credentials')
Zendesk::ConfigurationService::Exceptions::Unavailable: Zendesk

cs.available?
=> false

cs.status
=> {:available=>false}
```

COMMAND LINE TOOL

ZENDESK-CONFIGURATION-SERVICE

```
$ zendesk-configuration-service --help

Toggle(s)
    --debug                         Enable debugging (default: fal
    --warn                          Enable warnings (default: true
    --verbose                       Enable verbose (default: true)

Option(s)
    --config <value>                Configuration file path (defau

Subcommand(s)
    status                          Show the status
    get, read                       Read a configuration item
    set, write                      Write a configuration item
    del, delete, remove, rm         Delete a configuration item
    watch                           Watch a key for changes
    update-file                     Update a file from a key
```

CONFIG.YML

```
---
consul:
  address: "http://127.0.0.1:8500"
  token:   "2701E633-9EB5-4E38-B190-0CC809CDFC64"

hashicorp_vault:
  address: "http://127.0.0.1:8200"
  token:   "105286C0-63E7-493E-924C-42AA04651F13"
```

ENVIRONMENT VARIABLE

```
$ echo $ZENDESK_CONFIGURATION_SERVICE_CLASS
consul
```

DEMO

UPDATE-FILE

THANKS 😃