

Ansible role which installs openssl and creates/imports certificates

69 commits

2 branches

12 releases

5 contributors

MIT

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

franklinkim Merge branch 'release/1.10.0'

Latest commit 01a4007 on Dec 21, 2017

defaults	feat: generate csrs	a year ago
meta	feat: replace include with import_tasks for 2.4	11 months ago
tasks	feat: replace include with import_tasks for 2.4	11 months ago
templates	added openssl configuration	4 years ago
tests	feat: generate csrs	a year ago
vars	feat: optionally install cacert roots	a year ago
.clog.toml	feat: add CHANGELOG	3 years ago
.editorconfig	chor: fix makefile indentation	3 years ago
.gitignore	feat: add CHANGELOG	3 years ago
.travis.yml	chor: update travis testing	3 years ago
CHANGELOG.md	docs: update CHANGELOG	11 months ago
LICENSE	initial commit	4 years ago
Makefile	chor: fix makefile indentation	3 years ago
README.md	docs: update README	11 months ago
Vagrantfile	chor: update vagrant boxes	a year ago

README.md

Ansible franklinkim.openssl role

build

passing

galaxy

weareinteractive.openssl

tag

v1.10.0

stars

32

franklinkim.openssl is an [Ansible](#) role which:

- installs openssl
- configures openssl
- imports ssl certificates and keys
- creates a self-signed certificate
- optionally installs the CACert root certificates

Installation

Using `ansible-galaxy` :

```
$ ansible-galaxy install franklinkim.openssl
```

Using `requirements.yml` :

```
- src: franklinkim.openssl
```

Using `git` :

https://github.com/weareinteractive/ansible-openssl

1/4

```
$ git clone https://github.com/weareinteractive/ansible-openssl.git franklinkim.openssl
```

Dependencies

- Ansible >= 2.4

Variables

Here is a list of all the default variables for this role, which are also available in `defaults/main.yml`.

```
---
# openssl_keys:
#   - name: mykey.key
#     key: "mykeycontents"
#   - name: myotherkey.key
#     cert: "myotherkeycontents"
#     mode: "0664"
#     owner: "www-data"
#     group: "www-data"
# openssl_certs:
#   - name: mycert.crt
#     cert: "mycertcontents"
#   - name: myothercert.crt
#     cert: "myothercertcontents"
#     mode: "0664"
#     owner: "www-data"
#     group: "www-data"
# openssl_self_signed:
#   - name: 'foobar.com'
#     domains: ['*.foobar.com', 'foobar.com']
#     country: 'DE'
#     state: 'Bavaria'
#     city: 'Munich'
#     organization: 'Foo Bar'
#     unit: ''
#     email: 'foo@bar.com'
#     days: 3650
# openssl_config:
#   default_bits: 2048
#   countryName_default: DE
#   stateOrProvinceName_default: Bavaria
#   localityName_default: Munich
#   organizationName_default: 'My Organization'
#   organizationalUnitName_default: 'My Organization Unit'
#   commonName_default: 'foobar.com'
# openssl_config_template: templates/openssl.cnf.j2

# keys to import
openssl_keys: []
# certificates to import
openssl_certs: []
# path to certificates
openssl_certs_path: /etc/ssl/certs
# path to keys
openssl_keys_path: /etc/ssl/private
# default key owner
openssl_default_key_owner: ssl-cert
# default key group
openssl_default_key_group: root
# default cert owner
openssl_default_cert_owner: root
# default cert group
openssl_default_cert_group: root
# self signed certificates
openssl_self_signed: []
# config variables
openssl_config: {}
# config template to install, relative to the ansible repository root
openssl_config_template:
# generate a CSR for each self signed certificate
openssl_generate_csr: no
# path to certificate signing requests
```

```

openssl_csrs_path: /etc/ssl/csrs
# should CACert certificates be downloaded and added to the keyring?
openssl_cacert_import: no
# overrides for the file checksum when the CACert root certificates are downloaded.
# must be the output of 'sha256sum <name of certificate>'
openssl_cacert_class_one_key_sha256: 'c0e0773a79dceb622ef6410577c19c1e177fb2eb9c623a49340de3c9f1de2560'
openssl_cacert_class_three_key_sha256: 'f5badaa5da1cc05b110a9492455a2c2790d00c7175dcf3a7bcb5441af71bf84f'

```

Usage

This is an example playbook:

```

---

- hosts: all
  roles:
    - franklinkim.openssl
  vars:
    openssl_keys:
      - name: foobar.com.key
        key: "-----BEGIN PRIVATE KEY-----\nMIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAKggSkgAgEAAoIBAQDhSsYh36iAShd\nNM0dSxiVX
    openssl_certs:
      - name: foobar.com.crt
        cert: "-----BEGIN CERTIFICATE-----\nMIIDuTCCAqGgAwIBAgIJA07EaRwLzPYyMA0GCSqGSIb3DQEBCwUAMHMxCzAJBgNV\nnBAYTAKRF
    openssl_self_signed:
      - { name: 'foobar.org', country: 'DE', state: 'Bavaria', city: 'Munich', organization: 'Foo Bar', unit: '', email: '' }
      - { name: '*.foobar.org', domains: ['foobar.org', '*.foobar.org'], country: 'DE', state: 'Bavaria', city: 'Munich', organization: 'Foo Bar', unit: '', email: '' }
    openssl_keys_path: /etc/my-ssl/private
    openssl_certs_path: /etc/my-ssl/certs
    openssl_default_key_owner: root
    openssl_default_key_group: root
    openssl_default_cert_owner: root
    openssl_default_cert_group: root
    openssl_config:
      default_bits: 2048
      countryName_default: DE
      stateOrProvinceName_default: Bavaria
      localityName_default: Munich
      organizationName_default: 'My Organization'
      organizationalUnitName_default: 'My Organization Unit'
      commonName_default: 'foobar.com'
    openssl_cacert_import: yes
    openssl_generate_csr: yes

```

Testing

```

$ git clone https://github.com/weareinteractive/ansible-openssl.git
$ cd ansible-openssl
$ make test

```

Contributing

In lieu of a formal style guide, take care to maintain the existing coding style. Add unit tests and examples for any new or changed functionality.

1. Fork it
2. Create your feature branch (`git checkout -b my-new-feature`)
3. Commit your changes (`git commit -am 'Add some feature'`)
4. Push to the branch (`git push origin my-new-feature`)
5. Create new Pull Request

Note: To update the `README.md` file please install and run `ansible-role` :

```

$ gem install ansible-role
$ ansible-role docgen

```

License

Copyright (c) We Are Interactive under the MIT license.