## About

tnascert-deploy is a tool used to deploy TLS certificates to one or more TrueNAS systems running version 25.04, Fangtooth, or later. It is written in Go and when compiled for your target system, there are no other dependencies than the binary itself, **tnascert-deploy**.

The tool connects to the JSON-RPC 2.0 WebSocket API endpoint in order to deploy the certificates and private key for use as the TrueNAS UI certificate, FTPS service certificate, or as Docker App TLS certificates.
**tnascert-deploy** utilizes an INI configuration file where multiple TrueNAS systems may be configured in separate sections of the file. The user of the tool specifies one or more TrueNAS systems by their section name on the commandline defined in the configuration file in order to deploy certificates.

The tool may be utilized as part of an ACME, Automated Certificate Management Environment to deploy new or renewal certficates to TrueNAS 25.04 systems. The command line usage is as follows:

```
Usage: tnascert-deploy [-hv] [-c value] config_section ...
config_section`

-c, --config="full path to the configuration file [tnas-
cert.ini]".
-h, --help print usage information and exit.
-v, --version print version information and exit
```

Example to deploy certficates to two TrueNAS machines nas01 and nas02:

```
$ tnascert-deploy -c /etc/tnas-cert.ini nas01 nas02
```

## Getting Started

Precompiled releases of ‘tnascert-deploy’ are available for FreeBSD, Debian Linux, MacOS, or Windows 11. See the Releases section of this repository. The current Release is 1.3.

To build and test on any system with Go installed, clone this repository and run unit tests using:

```
'go test ./...' or 'make test' if you have make installed.
```

build ‘tnascert-deploy’ using:

```
'go build' or 'make' if you have make installed
```

copy ‘tnascert-deploy’ for use either as a command line tool or as part of your ACME deployment scripts and create an INI configuration file that lists all your TrueNAS systems.

```
tnascert-deploy -c /usr/local/etc/tnas-cert.ini nas01 nas02
```

### Configuration file

The default configuration file is named **tnas-cert.ini** and it is searched for in your current working directory if the **-c filename** option is not used. By using the ***-c filename*** option, you may specify the full path to the configuration file and use any filename that you like.

The configuration file uses the INI format that lists section names in square brackets followed by named value pairs separated by an equal sign. The ***deploy_default*** section name if defined, will be used if no other section name is listed on the commandline. The following shows an example configuration file with three TrueNAS systems configured. In the example there are 3 sections defined, ***default_deploy***, ***nas02***, and ***nas03***. If no section is listed on the tnascert-deploy commandline, the ***default_deploy*** configuration will be loaded and certificates will be deployed to the TrueNAS host defined in that section. Each individual NAS configuration can be loaded by listing only that desired section on the commandline. All 3 sections can be loaded and have certificates deployed in turn by listing all 3 sections on the commandline.

```
[default_deploy]
api_key = 1-
ZFhoN97YrxqWg5GIR3XjhPNuaO7NKAwDBbwCashgTCi0z4Mfy9sYo8e8g4WPMCO2
private_key_path = test_files/privkey.pem
cert_basename = letsencrypt
full_chain_path = test_files/fullchain.pem
connect_host = nas01.mydomain.com
protocol = wss
tls_skip_verify = false
delete_old_certs = true
add_as_ui_certificate = true
add_as_ftp_certificate = true
timeoutSeconds = 10
debug = false

# sample production config
[nas02]
api_key = 1-
ZFhoN97YrxqWg5GIR3XjhPNuaO7NKAwDBbwCashgTCi0z4Mfy9sYo8e8g4WPMCO2
private_key_path = test_files/privkey.pem
cert_basename = letsencrypt
full_chain_path = test_files/fullchain.pem
connect_host = nas02.mydomain.com
protocol = wss
tls_skip_verify = false
delete_old_certs = true
add_as_ui_certificate = false
add_as_ftp_certificate = false
timeoutSeconds = 10
debug = false

# sample production config
[nas03]
api_key = 2-
AFhoB89YqxrWg5GIR3XjhPFUao7NKAwDBbWcAshgTCi0z47fM9sYo8e8g4wpMCO2
```

```
cert_basename = letsencrypt
private_key_path = test_files/privkey.pem
full_chain_path = test_files/fullchain.pem
connect_host = nas03.mydomain.com
protocol = wss
tls_skip_verify = true
delete_old_certs = true
add_as_ui_certificate = false
add_as_ftp_certificate = true
timeoutSeconds = 10
debug = false
```

### Configuration File settings

In order to authenticate with a TrueNAS system, the user must either use the TrueNAS UI to generate and copy an **api_key** or use an admin **username** and **password** in the configuration file. The **api_key** is preferred and if all three are defined in the configuration file, only the **api_key** will be used. Do not include the **api_key** if you wish to use the **username** and **password**.

The following configuration settings are used for each NAS section:

```
+ api_key                string  - TrueNAS 64 byte API Key for
login the
                                   preferred login method).
+ username               string  - TrueNAS username with admin
privileges
                                   (API key is preferred for
login)
+ password               string  - TrueNAS password for user with
admin
                                   privileges, (API key is
preferred for
                                   login)
+ cert_basename          string  - basename for cert naming in
TrueNAS
+ connect_host           string  - TrueNAS DNS Fully Qualified
Domain
                                   Name, FQDN, or IP address
+ delete_old_certs       bool    - whether to remove old
certificates,
                                   default is false
+ full_chain_path        string  - path to full_chain.pem
+ port                   uint64  - TrueNAS API endpoint port
+ protocol               string  - websocket protocol 'ws' or
'wss' wss'
                                   is default
+ private_key_path       string  - path to private_key.pem
+ tls_skip_verify        bool    - strict SSL cert verification of
the
                                   endpoint, false by default
+ add_as_ui_certificate  bool    - install as the active UI
certificate
```

```
                                    if true
+ add_as_ftp_certificate bool    - install as the active FTP
service
                                    certificate if true
+ add_as_app_certificate bool    - install as the active APP
service
                                    certificate if true
+ timeoutSeconds         int64   - the number of seconds after
which the
                                    truenas client calls fail
+ debug                  bool    - debug logging if true
```

## Notes

This tool uses the TrueNAS Scale JSON-RPC 2.0 API and the TrueNAS client API module. Only ***TrueNAS 25.04** systems or later are supported.

## See Also

- TrueNAS api_client_golang
- TrueNAS websocket API documentaion

## Contact

- John J. Rushford
- jrushford@apache.org