
pyArchOps/tinc Documentation

Release 0.0.3

Azul

Jun 02, 2019

Contents:

1	pyArchOps/tinc	1
1.1	Features	1
1.2	Installation	1
1.3	Usage	1
1.4	Development	5
1.5	Credits	5
2	Installation	7
2.1	Stable release	7
2.2	Latest tinc release	7
2.3	From sources	7
3	Usage	9
4	pyarchops_tinc	11
4.1	pyarchops_tinc package	11
5	Contributing	13
5.1	Types of Contributions	13
5.2	Get Started!	14
5.3	Pull Request Guidelines	14
5.4	Deploying	15
6	Credits	17
6.1	Development Lead	17
6.2	Contributors	17
7	History	19
7.1	0.0.1 (2018-12-11)	19
8	Indices and tables	21
	Python Module Index	23
	Index	25

CHAPTER 1

pyArchOps/tinc

tinc

- Free software: MIT license
- Documentation: <https://pyarchops-tinc.readthedocs.io>.

1.1 Features

- tinc

1.2 Instalation

```
$ pip install pyarchops-tinc
```

1.3 Usage

```
import os
import textwrap
import pyarchops_tinc

tinc_network_name = 'core-vpn'
public_key = textwrap.dedent('''
    -----BEGIN RSA PUBLIC KEY-----
```

(continues on next page)

(continued from previous page)

```

MIIBCgKCAQEAwBLTc+75h13ZyLWlvup0OmbhZWxohLMMFCUBCLSMxZxZdMvyzBnW
+JpOQuvnasAeTLtEDWSID0AB/EG68Sesr58Js88ORUw3VrjObiG15/iLtAm6hiN
BboTqd8jgWrlyC3LfNSKJk82qQzHJPlCO9Gc5HcqVWrIrqrJL2kwjOU66U/iRxJu
dyOrz0sBkVgfwDBqNS96L0zFQCqk70w9KyOJqe4JNJUtBas6lbwgChDU4/B3BDW5
PYJy2Pp8MSs2n1lhrUkXxRnj+Vl5wLQLdwog1XAGu2J8pIckPg/aB7mB/fSlFihU
bnFlRlgHr1h8gyNYztbGWKMrQ4Bz2831PQIDAQAB
-----END RSA PUBLIC KEY-----

'''
private_key = textwrap.dedent('''
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAwBLTc+75h13ZyLWlvup0OmbhZWxohLMMFCUBCLSMxZxZdMvy
zBnW+JpOQuvnasAeTLtEDWSID0AB/EG68Sesr58Js88ORUw3VrjObiG15/iLtAm
6hiNBboTqd8jgWrlyC3LfNSKJk82qQzHJPlCO9Gc5HcqVWrIrqrJL2kwjOU66U/i
RxJudyOrz0sBkVgfwDBqNS96L0zFQCqk70w9KyOJqe4JNJUtBas6lbwgChDU4/B3
BDW5PYJy2Pp8MSs2n1lhrUkXxRnj+Vl5wLQLdwog1XAGu2J8pIckPg/aB7mB/fSl
FihUbnFlRlgHr1h8gyNYztbGWKMrQ4Bz2831PQIDAQABAOIBAQCUP1TWM/GX7m
yCmGuYV7TxAPMxprYeTlrdR7rQklo9Ac5pIQvSxYwFRUQVPDCsmzkLyNZ+wwgPvI
LJPANUKTsOzUrrS0UGD8cR9kPvaWtAqNX6n8syKNQTVd7pc6HrQKDbAMz0N65sqS
ExNkUNArStsMTnXePrDx9cxerYIOi8duRVH/VEBiluVh5+m7ggiYzjWxOy1lExd3
m5tqv0PicI6UTm6sUIar3pdypt0ditHBJN4apSIxv9yW+M+Uw4JEB1XL0Kq7o9M0
NZRwdtL9qlxBC8/lu9K/nd55OvBo0cuHR6ZIQioBIsr5A0LJ6mc0xfwllEQjUD7E
Z8pfW8blAoGBAP9iylxVvC3HWgXuTvDGPCbIdhw/AM4gR1Tz1eXGKCqyc9mTQR44
MKCF8nrzVvQU+j5VVGLwQhZIMp7VyDLQMw0uZdj+mfSs2qwFPClpbsa2hFco2xbs
+Ejr3MQpbDYlodo9hAiFcg1AsRsI8MnlngXFgEupMS+WSHQCMTZ5XcnAoGBAMCJ
D9ofWghlrMA1M5CSdukrcU26ScCq2b9WyhBTAd8v8SuOxLjVxsHjbhxZro9rEj0q
Qb4AiDB88ksLGpiBg36UKHUwpiq5vlhkb27r+EwpCB5CFelOBVIwMwv+kyLHep1
wdHEzndx9cWibmlHwIaX7RM2qOGuA//3art7Ag77AoGAf4/54hsU7ozXw4Sg05XY
78pLbJpvrYXj+2P8IFRVndaDFVd/PDf22gdt8cngUfs0djQrAqsC55xSZJIF+JOU
HG5jgvrRLay1YR0QR6PvqCP8gIiwvofJEKt3Tygdm/U9eAQoEhWNvV7118okc8RU
tlOpsxd4R6mIXeJKrwdjpBsCgYApOFqB+5cZCTloTOWS+0wZ3ZZw1Alab4B0vouJ
uglJBGdzF0GABUvWjE0ImS2A9jby06+Nbr4msawJQjMXdeEx50lWEie0VbySA9Xz
mAnHuI2LzLxoWi5rqA4eEnlgkEIB+vF59i0E4doHeVbJRIz6bhpNtuw8fddWsVy
TAepawKBgQDHOiiGIKsWR4Qrb0zTCQGLddRYE9/UG76ntTLA2ocOglLNZJtKFOGz
0DbxS6ow4enTcRMhvPf3Kn0AEdR9BlcnKgDZLZUqe0Plz1lQ4LZ7BXjHJgypzEEF
esViwAMKfer7C5GZwGMMaBVlzK/THREA1IPcOKwXopWgVvIbKus3VA==
-----END RSA PRIVATE KEY-----

''')

tinc_up_file = textwrap.dedent(f'''
#!/bin/sh
# see: https://www.tinc-vpn.org/pipermail/tinc/2017-January/004729.html
macfile=/etc/tinc/{tinc_network_name}/address
if [ -f $macfile ]; then
    ip link set tinc.{tinc_network_name} address `cat $macfile`
else
    cat /sys/class/net/tinc.{tinc_network_name}/address >$macfile
fi

# https://bugs.launchpad.net/ubuntu/+source/isc-dhcp/+bug/1006937
dhclient -4 -nw -v tinc.{tinc_network_name} -cf /etc/tinc/{tinc_network_name}/
->dhclient.conf -r
dhclient -4 -nw -v tinc.{tinc_network_name} -cf /etc/tinc/{tinc_network_name}/
->dhclient.conf
''')

tinc_down_file = textwrap.dedent(f'''
#!/bin/sh

```

(continues on next page)

(continued from previous page)

```

    dhclient -4 -nw -v tinc.{tinc_network_name} -cf /etc/tinc/{tinc_network_name}/
↪dhclient.conf -r
    '')

fix_route_file = textwrap.dedent(f'''
    #!/usr/bin/env bash

    sleep 15
    netstat -rnv | grep {tinc_network_name} | grep 0.0.0.0 >/dev/null 2>&1

    if [ $? = 0 ]; then
        route del -net {tinc_network_name} netmask 24 gateway 0.0.0.0
        route add -net {tinc_network_name} netmask 24 gateway `ifconfig tinc.{tinc_
↪network_name}| grep inet | awk '{{ print $2 }}'`
    fi
    '')

dhclient_conf_file = textwrap.dedent(f'''
    option rfc3442-classless-static-routes code 121 = array of unsigned integer 8;

    # https://bugs.launchpad.net/ubuntu/+source/isc-dhcp/+bug/1006937
    send host-name "my-host-name";
    #send dhcp-requested-address 10.16.254.23;

    request subnet-mask, broadcast-address, time-offset, routers,
        domain-name, domain-search, host-name,
        netbios-name-servers, netbios-scope, interface-mtu,
        rfc3442-classless-static-routes, ntp-servers;

    timeout 300;
    ''')

connect_to_hosts = {
    'core_network_01': textwrap.dedent('''
        Name=core_network_01
        Address=core01.example.com
        Port=655
        Compression=0
        Subnet=10.16.254.1/32

        -----BEGIN RSA PUBLIC KEY-----
        MIIBCgKCAQEAt9PKpazBuZMEH4mwX+yc0yq5sHT3D6fLJ+VHeJdSr6nsTWbthLGS
        ti+jt2bBniaUgkzbc1vRrWtM0sfRwhFbkBOXmXsSFQY19Yft7IqxG6nJ7JLiDMPL
        V1bK8xa9t2oXOdATNrTehq6oy8BeLls+BOKXJ9T6ZR6T4Hu4KN2tcf78ZMRCyrx1
        E10/D3YAEErnZj1KN44Agf2GJXbCfayicywvojz3Otyiu/4VMtjubgUitbHS8ZiM
        yomH9ayDdAdqyFTEQopH4zQ7GYPH2syziJsD9yRoc2CnMALb6q5rDGeL0foCeIZq
        TAFi2ceSo0lGQwdiFehiogZ2ng5P9/Iq5QIDAQAB
        -----END RSA PUBLIC KEY-----
    '''),

    'core_network_02': textwrap.dedent('''
        Name=core_network_02
        Address=core02.example.com
        Port=655
        Compression=0
        Subnet=10.16.254.2/32

```

(continues on next page)

(continued from previous page)

```

-----BEGIN RSA PUBLIC KEY-----
MIIBCgKCAQEApoRHeJrFbe6H9Wqi25PEHPLMr1+scReE1BFNs9U5UME4PY+AGF92
Qczpy70bH30quowy406zmglDQenIVWdpMpN2odm1V9OAz4vyk/AZzWK/wVDrIqqy
OUFQnocdWwONGkfLShM5DHiyi5FcDS0oORWrh6LobxSQdBOZTKfgy2F6xJYUWAq7
7jmqbJb40/Cd4BvyJekzkU71Y1TWfQnX7hvRD9S3pCpknwqtFnD8MYE+zv9p/uNC
OsrCncN4Ur3/pcAqRRRfLuZqNvXTf1+Hqk6jTPM0s1UCV5LtcEcbB4xzV8boDipm
//8326DMTzUJntyqdhRxEYRjGuvI/Ri6uwIDAQAB
-----END RSA PUBLIC KEY-----

'''),

'core_network_03': textwrap.dedent('''
    Name=core_network_03
    Address=core03.example.com
    Port=655
    Compression=0
    Subnet=10.16.254.3/32

    -----BEGIN RSA PUBLIC KEY-----
    MIIBCgKCAQEA81/Mkzf+qwxQ+Py8O5lyOWUmN84aAd5Z9d1XrCQ3iuHm7g5J0K4f
    U3JHMEOn0RU2RRUbUXiK0L8LeHCU7TiuBwz8+nOy/HCeZCiGUTuBkGKiIWSVVGY/
    SOHvAIUHhCCPMmgIuTBDyWdthDGVPr++klsRXsK5ODrPclqJzF5f5AKUaEfcfOw
    XdBn08fOu3S8XSUoOepSS8mPQv2D0LB7hPXWlAV8tu3R9ibx2oR22c9zBWjgG5y
    tMhI3vLvnnzf2+0bNdY1ekc6G5wcCFuZb8qxt8+88Ls5Ek5jUc9Z1aqpe7x6MaYP
    KafbBTGK6BXmaAhyiQG4aALH5U2+Z17BOWIDAQAB
    -----END RSA PUBLIC KEY-----

'''),
}

tinc_conf_file = textwrap.dedent(f'''
    Name = laptop
    DeviceType = tap

    Device = /dev/net/tun
    Interface = tinc.{tinc_network_name}
    AddressFamily = ipv4
    LocalDiscovery = yes
    Mode=switch
    ConnectTo = core_network_01
    ConnectTo = core_network_02
    ConnectTo = core_network_03
    Cipher=aes-256-cbc
    ProcessPriority = high
''')

config = {
    'tinc_network_name': tinc_network_name,
    'public_key': public_key,
    'private_key': private_key,
    'tinc_up_file': tinc_up_file,
    'tinc_down_file': tinc_down_file,
    'fix_route_file': fix_route_file,
    'tinc_conf': tinc_conf_file,
    'dhclient_conf': dhclient_conf_file,
    'connect_to_hosts': connect_to_hosts,
}

```

(continues on next page)

(continued from previous page)

```
api = Api(
    '127.0.0.1:22',
    connection='smart',
    remote_user='root',
    private_key_file=os.getenv('HOME') + '/.ssh/id_rsa',
    become=True,
    become_user='root',
    sudo=True,
    ssh_extra_args='-o StrictHostKeyChecking=no'
)

result, logs = pyarchops_tinc.apply(api, config=config)

print(result)
print(logs)
```

1.4 Development

Install requirements:

```
$ sudo pacman -S tmux python-virtualenv python-pip libjpeg-turbo gcc make vim git tk_
↪tcl
```

Git clone this repository

```
$ git clone https://github.com/pyarchops/tinc.git pyarchops.tinc
$ cd pyarchops.tinc
```

2. See the *Makefile*, to get started simply execute:

```
$ make up
```

1.5 Credits

- TODO

2.1 Stable release

The pyArchOps software should be installed through the main repository, .. _pyArchOps : <https://github.com/pyarchops/pyarchops.git>

```
$ pip install pyarchops
```

2.2 Latest tinc release

To install pyArchOps/tinc, run this command in your terminal:

```
$ pip install pyarchops_tinc
```

This is the preferred method to install pyArchOps/tinc, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

2.3 From sources

The sources for pyArchOps/tinc can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/pyarchops/tinc
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/pyarchops/tinc/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

CHAPTER 3

Usage

To use pyArchOps/tinc in a project:

```
import pyarchops_tinc
```


4.1 pyarchops_tinc package

4.1.1 Submodules

4.1.2 pyarchops_tinc.tinc module

Main module.

`pyarchops_tinc.tinc.apply` (*api: suitable.api.Api, config: dict, quiet: bool = False*) → Tuple
installs tinc

`pyarchops_tinc.tinc.create_tinc_network_folder` (*api: suitable.api.Api, config: dict*) → dict
creates /etc/tinc/<network>

`pyarchops_tinc.tinc.deploy_configuration` (*api: suitable.api.Api, config: dict, quiet: bool = False*) → dict
deploys tinc configuration

`pyarchops_tinc.tinc.deploy_hosts_files` (*api: suitable.api.Api, config: dict*) → dict
deploys tinc /hosts/* files

`pyarchops_tinc.tinc.deploy_interface_files` (*api: suitable.api.Api, config: dict*) → dict
deploys tinc interface files

`pyarchops_tinc.tinc.deploy_tinc_conf` (*api: suitable.api.Api, config: dict*) → dict
deploys tinc.conf

`pyarchops_tinc.tinc.deploy_tinc_key_files` (*api: suitable.api.Api, config: dict*) → dict
deploys tinc id_rsa files

`pyarchops_tinc.tinc.enable_service` (*api: suitable.api.Api, config: dict, quiet: bool = False*) → dict
enables tinc services

`pyarchops_tinc.tinc.install_pkgs` (*api: suitable.api.Api, quiet: bool = False*) → dict
installs tinc requirements

4.1.3 Module contents

`__init__.py` for `pyarchops_tinc`

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

5.1 Types of Contributions

5.1.1 Report Bugs

Report bugs at <https://github.com/pyArchOps/pyarchops/tinc/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

5.1.4 Write Documentation

pyArchOps/tinc could always use more documentation, whether as part of the official pyArchOps/tinc docs, in docstrings, or even on the web in blog posts, articles, and such.

5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/pyArchOps/tinc/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

5.2 Get Started!

Ready to contribute? Here's how to set up *pyarchops/tinc* for local development.

1. Fork the *pyarchops/tinc* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/pyarchops/tinc.git
```

3. set a development environment:

```
$ sudo pacman -Sy tmux libffi pkgconf base-devel make $ cd pyarchops/tinc/ $ make up
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass pylint and the tests.
6. Commit your changes and push your branch to GitHub:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.

3. The pull request should work for Python 3.7. Check <https://gitlab.com/pyarchops/tinc/pipelines> and make sure that the tests pass for all supported Python versions.

5.4 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bumpversion patch # possible: major / minor / patch
$ git push
$ git push --tags
```


6.1 Development Lead

- Azul <pyarchops@azulinho.com>

6.2 Contributors

None yet. Why not be the first?

7.1 0.0.1 (2018-12-11)

- First release on PyPI.

CHAPTER 8

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`pyarchops_tinc`, [12](#)

`pyarchops_tinc.tinc`, [11](#)

A

`apply()` (*in module `pyarchops_tinc.tinc`*), 11

C

`create_tinc_network_folder()` (*in module `pyarchops_tinc.tinc`*), 11

D

`deploy_configuration()` (*in module `pyarchops_tinc.tinc`*), 11

`deploy_hosts_files()` (*in module `pyarchops_tinc.tinc`*), 11

`deploy_interface_files()` (*in module `pyarchops_tinc.tinc`*), 11

`deploy_tinc_conf()` (*in module `pyarchops_tinc.tinc`*), 11

`deploy_tinc_key_files()` (*in module `pyarchops_tinc.tinc`*), 11

E

`enable_service()` (*in module `pyarchops_tinc.tinc`*), 11

I

`install_pkgs()` (*in module `pyarchops_tinc.tinc`*), 11

P

`pyarchops_tinc` (*module*), 12

`pyarchops_tinc.tinc` (*module*), 11