kea_config

Release 4.6.1

Gene C

Dec 19, 2023

CONTENTS:

CHAPTER

ONE

KEA-CONFIG

1.1 Overview

What is kea?

kea is a modern dhcp server from ISC (<https://www.isc.org/kea>) which supercedes its older dhcp software.

kea offers a nice feature set including the ability to have a hot standby server to pick up in case the primary is unavailable. Its power also lurks behind a complicated configuration suite that, at least for me, is not terribly human friendly.

Most notably each server requires it's own separate config and keeping them all synchronized can be a bit of a chore and which naturally is prone to human error.

What is kea-config?

kea-config provides a tool which takes a single configuration file as its input and it then generates the native kea configuration files needed from that single source of truth. By using a single configuration we can be assured that the configs for the primary, standby and backup servers are consistent with one other.

It also provides the convenience of doing the DNS lookups for any host reservations, meaning the reservation is specified using hostname only not IP as expected by kea.

At the moment kea-config supports kea-dhcp4 and its companion control agent.

Contents

1.2 Installation

Available on

- Github
- Archlinux AUR

On Arch you can build using the PKGBUILD provided in packaging directory or from the AUR package.

Listing 1: Manual Install

```
rm -f dist/*
/usr/bin/python -m build --wheel --no-isolation
root_dest="/"
./scripts/do-install $root_dest
```

When running as non-root then set root_dest a user writable directory This will install the executable */usr/bin/kea-config* along with a sample config in */usr/share/kea_config*

1.3 kea_config application

kea-config is written in python and that is its sole dependency, hence python must be installed.

You can install it or run it out of the cloned repo (src/kea_config/kea-config.py)

1.3.1 Using kea-config

Once installed, to use it, copy the sample config file in the *configs* dir, modify for your own setup and simply run it:

```
kea-config -c <your.conf>
```

This will generate pairs of files, one kea config and one control agent config for each of primary, standby and backup - or whatever subset you used in the conf file.

e.g. it will create kea configs in the <conf_dir> which is defined config being used:

```
kea-ctrl-agent-primary.conf
kea-dhcp4-primary.conf
```

and similarly for standby and/or backup if requested. Each pair of files is to be used on the corresponding server. e.g The 2 primary files are used on the kea-dhcp4 primary server.

One simple way to manage these is to copy the entire <conf_dir> to each kea server /etc/kea then use sym links for kea config - linking to appropriate primary, standby or backup.

e.g. /etc/kea on primary would have

```
kea-dhcp4.conf -> <conf_dir>/kea-dhcp4-primary.conf
kea-ctrl-agent.conf -> <conf_dir>/kea-ctrl-agent-primary.conf
```

1.3.2 Summary of config variable

Comments begin with '#' and are ignored. The conf file in standard TOML format and as usual sections are denoted by square brackets. e.g.:

```
some_variable = 'xxx'
[section_1]
    a_variable = 'hi'
    a_list = ['1', 'two', 'three']
```

See the sample config for additional details. We summarize the main pieces here:

- *title*

  For human use only - not used by kea-config.

- *conf_dir*

  Directory where generated kea configs reside. What I do is rsync this directory to /etc/kea/ on each kea server. Each server then has a soft link to its own specific config. For example on my primary server I have

```
ln -s <conf_dir>/kea-ctrl-agent-primary.conf kea-ctrl-agent.conf
ln -s <conf_dir>//kea-dhcp4-primary.conf kea-dhcp4.conf
```

And similarly for standby and backup.

- *server_types*

  The list of servers used - should contain at least 'primary'. e.g. server_types = ['primary', 'standby', 'backup']

- [*global_options*]

  This section has some common dhcp information shared with dhcp clients:

  - domain-name-servers - list of DNS server IPs
  - domain-name - what is sounds like
  - domain-search - list of (sub)domains to search (if any)
  - ntp-servers - list of local ntp server IPs (if any)

- *[server.primary]*

  Provides the information needed for the primary server interface, hostname, port, auth_user and auth_password

- *[server.standby] [server.backup]*

  Same format as primary server section. Optional and only used if turned on in *server_types* list.

- *[net]*

  This section describes the standard dhcp information including host IP reservations.

  - dns_net

    internal domain, used to lookup IP for host reservations.

  - pools

    list of IP ranges to use

  - subnet

    what it sounds like

  - max-valid-lifetime

    as usual in seconds

  - *[net.option-data]*

    sub section with:

    * *broadcast-address*
    * *routers*

      default gateway / route

* *ntp-servers*

  A list

  - *[net.reserved.XXX]*

    host XXX hardware-address = “mac address”

    Will reserve the IP for XXX based on dns lookup of XXX. Have as many of these as needed.

1.4 Discussion and Next Steps

This version is for kea-dhcp4 (IPv4).

Not all kea options are supported by kea-config. For example the high availibilty component of kea allows for either hot-standby or load balancing. At moment we only support hot standby. Hot standby has one server at a time actively serving clients, whereas in load balancing case both servers are servicing clients at same time.

To create a version for kea-dhcp6, for example where a firewall is responsible for passing prefix delegation to the internal hosts, one needs an IPV6 internet connection; I am unable to work on this at the moment.

While kea-config is distro agnostic, I do provide an Archlinux package available on the AUR.

CHAPTER

TWO

APPENDIX

2.1 Dependencies

- Run time
- python
- Building Package:
  - git
  - poetry (aka python-poetry)
  - wheel (aka python-wheel)
  - build (aka python-build)
  - installer (aka python-installer)
  - rsync
- Optional for building docs:
  - sphinx
  - texlive-latexextra (archlinux packaguing of texlive tools)

2.2 Philosophy

We follow the *live at head commit* philosophy. This means we recommend using the latest commit on git master branch. We also provide git tags.

This approach is also taken by Google[1][2].

[1] https://github.com/google/googletest
[2] https://abseil.io/about/philosophy#upgrade-support

2.3 License

Created by Gene C. and licensed under the terms of the MIT license.

- SPDX-License-Identifier: MIT
- Copyright (c) 2022-2023 Gene C

CHAPTER

THREE

CHANGELOG

[4.6.1] —— 2023-12-19

- Update depends array in PKGBUILD
- update Docs/Changelog.rst

[4.6.0] —— 2023-11-26

- Switch python backend build to hatch (was poetry)
- update Docs/Changelog.rst

[4.5.1] —— 2023-09-27

- fix links in README
- update Docs/Changelog.rst

[4.5.0] —— 2023-09-27

- Reorganize docs and move to rst Now simple to build html and pdf docs using sphinx
- update CHANGELOG.md

[4.4.3] —— 2023-05-18

- install: switch from pip to python installer package. This adds optimized bytecode
- update CHANGELOG.md

[4.4.2] —— 2023-05-17

- Simplify Arch PKGBUILD and more closely follow arch guidelines
- update CHANGELOG.md

[4.4.1] —— 2023-01-06

- Add SPDX licensing lines
- update CHANGELOG.md

[4.4.0] —— 2022-12-14

- Use poetry to build wheel in PKGBUILD Installer now uses pip install Update readme build to use poetry
- update CHANGELOG.md

[4.3.2] —— 2022-11-06

- remove un-needed comments
- update CHANGELOG.md

[4.3.1] —— 2022-11-05

- small tweak to readme and sample config
- tweak readme
- aur package now available
- update CHANGELOG

[4.3.0] —— 2022-11-05

- kea_config - Manage kea dhcp4 configs from single source config

CHAPTER

FOUR

MIT LICENSE

CHAPTER

FIVE

HOW TO HELP WITH THIS PROJECT

Thank you for your interest in improving this project. This project is open-source under the MIT license.

5.1 Important resources

- Git Repo

5.2 Reporting Bugs or feature requests

Please report bugs on the issue tracker in the git repo. To make the report as useful as possible, please include

- operating system used
- version of python
- explanation of the problem or enhancement request.

5.3 Code Changes

If you make code changes, please update the documentation if it's appropriate.

CHAPTER

SIX

CONTRIBUTOR COVENANT CODE OF CONDUCT

6.1 Our Pledge

In the interest of fostering an open and welcoming environment, we as contributors and maintainers pledge to making participation in our project and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

6.2 Our Standards

Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

6.3 Our Responsibilities

Maintainers are responsible for clarifying the standards of acceptable behavior and are expected to take appropriate and fair corrective action in response to any instances of unacceptable behavior.

Maintainers have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, or to ban temporarily or permanently any contributor for other behaviors that they deem inappropriate, threatening, offensive, or harmful.

6.4 Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project or its community. Examples of representing a project or community include using an official project e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event. Representation of a project may be further defined and clarified by project maintainers.

6.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by contacting the project team at <arch@sapience.com>. All complaints will be reviewed and investigated and will result in a response that is deemed necessary and appropriate to the circumstances. The Code of Conduct Committee is obligated to maintain confidentiality with regard to the reporter of an incident. Further details of specific enforcement policies may be posted separately.

6.6 Attribution

This Code of Conduct is adapted from the Contributor Covenant, version 1.4, available at https://www.contributor-covenant.org/version/1/4/code-of-conduct.html

6.7 Interpretation

The interpretation of this document is at the discretion of the project team.

CHAPTER

SEVEN

INDICES AND TABLES

- genindex
- modindex
- search