
TracerouteDB Documentation

Release 0.1.0

Ryan Carter

December 09, 2016

1	TracerouteDB	3
1.1	Features	3
1.2	Examples	3
2	Installation	7
3	Usage	9
4	Contributing	11
4.1	Types of Contributions	11
4.2	Get Started!	12
4.3	Pull Request Guidelines	12
4.4	Tips	13
5	Credits	15
5.1	Development Lead	15
5.2	Contributors	15
6	History	17
7	0.1.0 (2015-01-11)	19
8	Indices and tables	21

Contents:

TracerouteDB

Experiment in full mesh historical traceroutes

Lasciate ogni speranza, voi ch'intrate.

This is an UNFINISHED project and likely won't even work depending on what bad changes have been made at any one time.

- Free software: GPLv2 license
- Documentation: <https://traceroutedb.readthedocs.org>.

1.1 Features

- TODO

1.2 Examples

find traceroutes where any hop is over a time

```
traceroutedb> SELECT * FROM traceroute WHERE traceroute_id in (SELECT DISTINCT traceroute_id FROM
```

traceroute_id	origin_ip	dest_ip	cdate	reporter
6	10.0.0.43	173.245.48.10	2015-02-02 22:15:17.632821-08:00	nm
7	10.0.0.43	8.8.8.8	2015-02-02 22:52:46.240210-08:00	nm

find full traceroute like output

```
traceroutedb> SELECT * FROM trv_trace WHERE traceroute_id = 186 ORDER BY hop_number, probe_id;
```

reporter	traceroute_id	origin_ip	dest_ip	probe_id	hop_number	host
wintermute	186	10.0.0.90	8.8.8.8	4924	1	10.0.0.1
wintermute	186	10.0.0.90	8.8.8.8	4925	1	10.0.0.1
wintermute	186	10.0.0.90	8.8.8.8	4926	1	10.0.0.1
wintermute	186	10.0.0.90	8.8.8.8	4930	2	96.120.89
wintermute	186	10.0.0.90	8.8.8.8	4931	2	96.120.89
wintermute	186	10.0.0.90	8.8.8.8	4932	2	96.120.89
wintermute	186	10.0.0.90	8.8.8.8	4927	3	162.151.3
wintermute	186	10.0.0.90	8.8.8.8	4928	3	162.151.3

```

| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4929 | 3 | 162.151.3
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4933 | 4 | 68.87.193
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4934 | 4 | 68.87.193
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4935 | 4 | 68.87.193
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4939 | 6 | 68.86.86.
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4940 | 6 | 68.86.84.
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4941 | 6 | 68.86.86.
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4936 | 7 | 66.208.22
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4937 | 7 | 66.208.22
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4938 | 7 | 66.208.22
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4945 | 8 | 216.239.4
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4946 | 8 | 216.239.4
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4947 | 8 | 216.239.4
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4942 | 9 | 216.239.4
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4943 | 9 | 216.239.5
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4944 | 9 | 216.239.5
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4921 | 10 | 8.8.8.8
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4922 | 10 | 8.8.8.8
| wintermute | 186 | 10.0.0.90 | 8.8.8.8 | 4923 | 10 | 8.8.8.8
SELECT 27

```

size of db with 730 traces

```

dev_fzt> SELECT nspname || '.' || relname AS "relation",
pg_size_pretty(pg_total_relation_size(C.oid)) AS "total_size"
FROM pg_class C
LEFT JOIN pg_namespace N ON (N.oid = C.relnamespace)
WHERE nspname NOT IN ('pg_catalog', 'information_schema')
AND C.relkind <> 'i'
AND nspname !~ '^pg_toast'
ORDER BY pg_total_relation_size(C.oid) DESC
LIMIT 20;
+-----+-----+
| relation | total_size |
+-----+-----+
| public.hop | 2320 kB |
| public.traceroute | 160 kB |
| public.probe_id_seq | 8192 bytes |
| public.traceroute_id_seq | 8192 bytes |
| public.annotation | 8192 bytes |
+-----+-----+
SELECT 5

```

Annotations are usually bad, lets find them

```

traceroutedb> SELECT DISTINCT traceroute_id from hop where (hop_kvs->'anno') IS NOT NULL;
+-----+
| traceroute_id |
+-----+
| 25 |
+-----+
SELECT 1

```

Find missing hops (though this has questionable utility)

```

traceroutedb> SELECT prev_id + 1 as missing FROM ( SELECT DISTINCT hop_number, LAG(hop_number) OVER
| missing |
|-----|
| 7 |

```



```
SELECT 1
```

Find same routers at same distance between N traces

```
traceroutedb> SELECT hop_number, ARRAY(SELECT DISTINCT UNNEST(array_agg(host))) FROM trv_trace W
| hop_number | array
|-----+-----|
|          1 | {192.168.43.1}
|          2 | {172.26.96.169}
|          3 | {172.16.157.164}
|          4 | {12.249.2.49}
|          5 | {12.83.180.82}
|          6 | {12.122.137.181}
|          7 | {12.250.31.10}
|          8 | {209.85.244.23,209.85.241.171}
|          9 | {64.233.174.43,216.239.49.123,216.239.56.127,216.239.56.123,209.85.255.255}
|         10 | {8.8.8.8}
SELECT 10
```

Aggregate times from hops

```
traceroutedb> SELECT hop_number, json_build_object(host, json_agg(cast(hop_kvs->'time' as double p
| hop_number | time
|-----:|-----|
|          1 | {"10.0.0.1" : [3.568, 3.189, 3.053]}
|          2 | {"96.120.89.253" : [18.729, 18.064, 11.431]}
|          3 | {"162.151.31.33" : [19.332, 19.331, 18.952]}
|          4 | {"68.87.193.129" : [20.54, 21.1, 19.922]}
|          5 | {"68.86.90.93" : [19.491]}
|          6 | {"68.86.87.158" : [14.088, 20.12, 29.474]}
|          7 | {"68.86.88.190" : [18.398, 18.772, 18.01]}
|          8 | {"66.208.216.38" : [20.68]}
|          8 | {"66.208.216.42" : [20.624]}
|          8 | {"66.208.216.34" : [20.613]}
|          9 | {"202.97.50.73" : [19.168, 20.45, 19.843]}
|         10 | {"202.97.51.229" : [165.616]}
|         12 | {"202.97.33.17" : [178.856, 178.248, 179.682]}
|         13 | {"61.152.86.193" : [172.432, 170.878, 173.214]}
SELECT 14
```

Installation

At the command line:

```
$ easy_install traceroutedb
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv traceroutedb  
$ pip install traceroutedb
```

Usage

To use TracerouteDB in a project:

```
import traceroutedb
```

Contributing

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

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at <https://github.com/eiginn/traceroutedb/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.

4.1.2 Fix Bugs

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

4.1.3 Implement Features

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

4.1.4 Write Documentation

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

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/eiginn/traceroutedb/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 :)

4.2 Get Started!

Ready to contribute? Here's how to set up *traceroutedb* for local development.

1. Fork the *traceroutedb* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/traceroutedb.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv traceroutedb
$ cd traceroutedb/
$ python setup.py develop
```

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 flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 traceroutedb tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

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.

4.3 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 2.6, 2.7, 3.3, and 3.4, and for PyPy. Check https://travis-ci.org/eiginn/traceroutedb/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_traceroutedb
```

Credits

5.1 Development Lead

- Ryan Carter <ryan@cloudflare.com>

5.2 Contributors

None yet. Why not be the first?

History

0.1.0 (2015-01-11)

- First release on PyPI.

Indices and tables

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