

GitHub Load Balancer Director and supporting tooling.

148 commits

4 branches

0 releases

7 contributors

View license

Branch: master ▾

New pull request

Create new file

Upload files

Find File

Clone or download ▾

theojulienne	Merge pull request #49 from lmb/icmp-pmtu ...	Latest commit 07b2c37 on Feb 15
docs	Implement configuration of flow hash fields.	8 months ago
script	Merge pull request #49 from lmb/icmp-pmtu	3 months ago
src	Merge pull request #49 from lmb/icmp-pmtu	3 months ago
.gitignore	Refactor packet parsing and flow hash calculation.	8 months ago
CODE_OF_CONDUCT.md	Add CONTRIBUTING and CODE_OF_CONDUCT, stub LICENSE.md.	10 months ago
CONTRIBUTING.md	Document apt release instructions.	9 months ago
LICENSE.md	Add reference to MIT license as well.	10 months ago
Makefile	Add build of glb-director package with default config/service scripts.	10 months ago
README.md	Add packages/quick start guide.	9 months ago
Vagrantfile	Merge pull request #49 from lmb/icmp-pmtu	3 months ago
requirements.txt	Pin versions and downgrade Scapy.	3 months ago

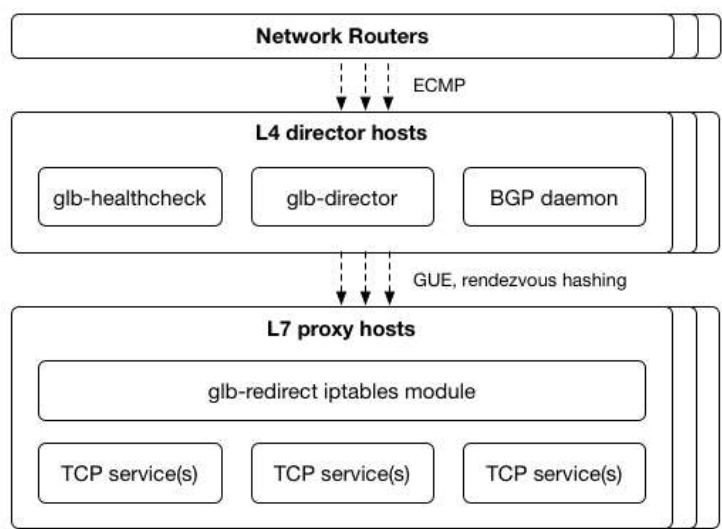
README.md

GitHub Load Balancer Director

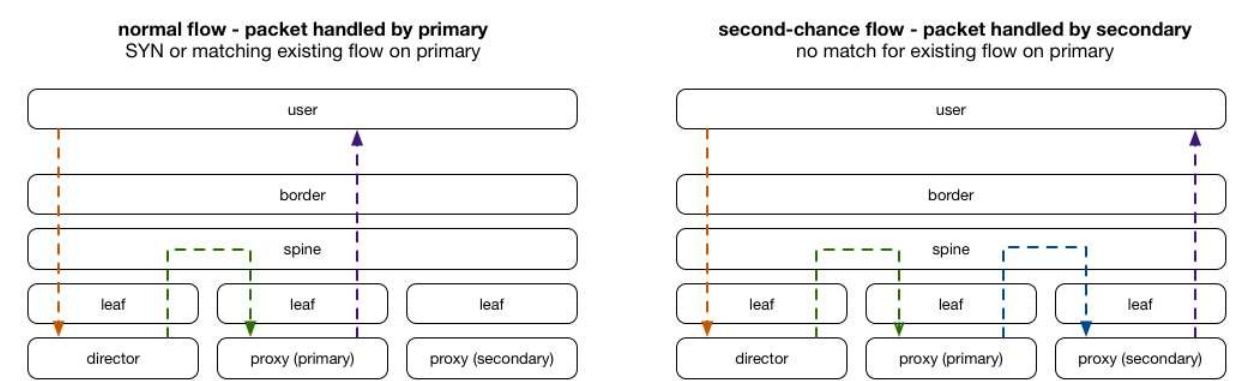
The GitHub Load Balancer (GLB) Director is a set of components that provide a scalable set of stateless [Layer 4](#) load balancer servers capable of line rate packet processing in bare metal datacenter environments, and is used in production to serve all traffic from GitHub's datacenters.

Design

GLB Director is designed to be used in datacenter environments where multiple servers can announce the same IP address via BGP and have network routers shard traffic amongst those servers using [ECMP routing](#). While ECMP shards connections per-flow using consistent hashing, addition or removal of nodes will generally cause some disruption to traffic as state isn't stored for each flow. A split L4/L7 design is typically used to allow the L4 servers to redistribute these flows back to a consistent server in a flow-aware manner. GLB Director implements the L4 (director) tier of a split L4/L7 load balancer design.



Traditional solutions such as [LVS](#) have stored flow state on each director node and then shared flow state between nodes. GLB Director instead receives these flows and uses a [derivative of rendezvous hashing](#) to hash flows to a pair of servers with a pre-determined order, and [leverages the state already stored](#) on those servers to allow flows to complete after a server begins draining.



GLB Director only processes packets on ingress, and encapsulates them inside an [extended Generic UDP Encapsulation](#) packet. Egress packets from proxy layer servers are sent directly to clients using Direct Server Return.

Getting started

GLB Director has a number of components that work together with other infrastructure components to create a complete load balancer. We've created an [example Vagrant setup/guide](#) which will create a local instance of GLB with all required components. The [docs](#) directory also contains additional documentation on the design and constraints. For details about the packages provided and how to install them, see the [packages and quick start guide](#).

Contributing

Please check out our [contributing guidelines](#).

License

Components in this repository are licensed under BSD 3-Clause except where required to be GPL v2 depending on their dependencies and usage, see the [license documentation](#) for detailed information.

Authors

GLB Director has been an ongoing project designed, authored, reviewed and supported by various members of GitHub's Production Engineering organisation, including:

- [@joewilliams](#)
- [@nautalice](#)
- [@ross](#)
- [@theojulienne](#)