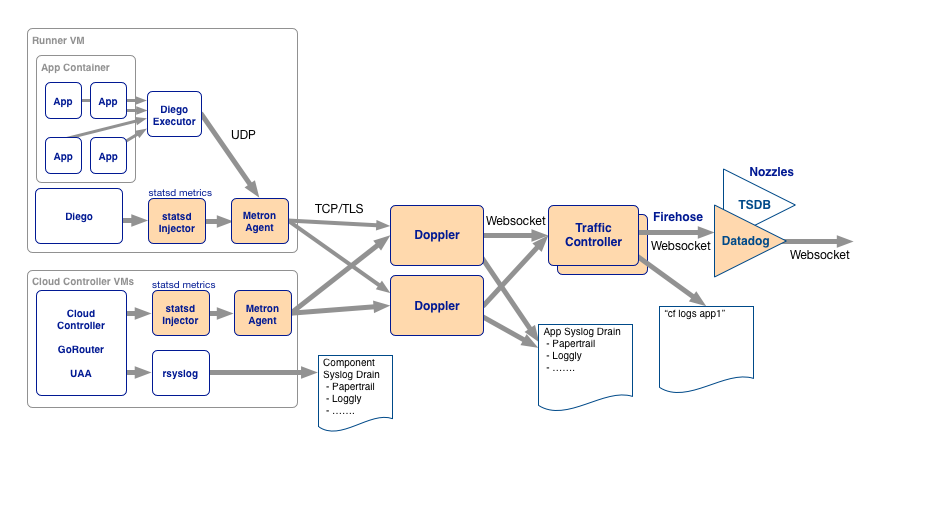
**Overview of the Loggregator System Using Loggregator**

The main use cases are as follows:

* App developers can tail their application logs or dump the recent logs from the Cloud Foundry Command Line Interface (cf CLI), or stream these to a third party log archive and analysis service.
* Operators and administrators can access the **Loggregator**[**Firehose**](http://docs.pivotal.io/pivotalcf/1-9/loggregator/architecture.html#firehose), the combined stream of logs from all apps, and the metrics data from Cloud Foundry components.
* Operators can deploy “nozzles” to the Firehose. A nozzle is a component that listens to the Firehose for specified events and metrics, and streams this data to external services.

# Loggregator Components

Overview of Loggregator components

## Source

Sources are logging agents that run on the Cloud Foundry components.

## Metron

Metron agents are co-located with sources. They collect logs and forward them to the Doppler servers.

## Doppler

Dopplers gather logs from the Metron agents, store them in temporary buffers, and forward them to the Traffic Controller or to third party syslog drains.

## Traffic Controller

Handles client requests for logs. Gathers and collates messages from all Doppler servers, and provides external API and message translation (as needed for legacy APIs). Exposes the Firehose.

## Firehose

The Firehose is a WebSocket endpoint which streams all the event data coming from an Elastic Runtime deployment. The data stream includes logs, HTTP events and container metrics from all applications, and metrics from all Elastic Runtime system components. Logs from system components such as Cloud Controller are not included in the firehose and are typically accessed via rsyslog configuration.

Because the data coming from the Firehose may contain sensitive information, such as customer information in the application logs, the Firehose is only accessible by users who have the right permissions.

The Traffic Controller serves the Firehose over WebSocket at the /firehose endpoint. The events coming out of the Firehose are formatted as protobuf messages conforming to the [dropsonde protocol](https://github.com/cloudfoundry/dropsonde-protocol).

The address of the traffic controller can be discovered by hitting the info endpoint on the API and getting the value of the doppler\_logging\_endpoint.

Example output for a BOSH Lite CF environment:

$ cf curl /v2/info | jq .doppler\_logging\_endpoint

wss://doppler.192.0.2.34.xip.io:443

### Difference Between Logs and Metrics

The Firehose carries both logs and metrics, which differ as follows:

* **Metrics**
  + Report a measurement from or status of a VM at its timestamp time
  + Follow [dropsonde](https://github.com/cloudfoundry/dropsonde-protocol) or [statsd](https://github.com/etsy/statsd) protocol
  + Can route to a monitoring platform to trigger alerts
* **Logs**
  + Report events detected, actions taken, errors, or any other messages the operator or developer wanted to generate
  + Follow the syslog standard
  + Are not used to trigger alert

Nozzles

Nozzles are programs which consume data from the Loggregator Firehose. Nozzles can be configured to select, buffer, and transform data, and forward it to other applications and services. For example:

* The [Datadog nozzle](https://github.com/cloudfoundry-incubator/datadog-firehose-nozzle) publishes metrics coming from the Firehose to Datadog.
* The [Syslog nozzle](https://github.com/cloudfoundry-community/firehose-to-syslog) filters out log messages coming from the Firehose and sends it to a syslog server.

See our [Nozzle Tutorial](http://docs.pivotal.io/pivotalcf/1-9/loggregator/nozzle-tutorial.html).