

title: Logagent Monitoring description: How to monitor Logagent runtime performance metrics

Node.js runtime performance

Containerized environment

If you are running Logagent in containers you should be running sematext/logagent. Set the MONITORING_TOKEN environmental variable to ship Logagent's Node.js runtime metrics to Sematext. Just make sure the token is for the Node.js Monitoring App.

Example:

```
docker run -it \
  -e MONITORING_TOKEN=<nodejs app token here> \
  -e INFRA_TOKEN=<infra app token here> \
  -e LOGS_TOKEN=<logs app token here> \
  -v /var/run/docker.sock:/var/run/docker.sock \
  sematext/logagent
```

If you use Sematext agent helm chart, use `logagent.config.MONITORING_TOKEN` and `logagent.config.INFRA_TOKEN` to activate Logagent metrics:

```
helm install st-agent \
--set logsToken=<logs app token here> \
--set infraToken=<infra app token here> \
--set region=US \
--set logagent.config.MONITORING_TOKEN=<nodejs app token here> \
--set logagent.config.INFRA_TOKEN=<infra app token here> \
--namespace sematext stable/sematext-agent
```

Non-containerized environment

In non-containerized environments install the logagent-nodejs-monitor plugin.

Installation

```
npm i -g @sematext/logagent-nodejs-monitor
```

Configuration

Add the following to the Logagent config file:

input:

```
  logagent-monitor:
    module: @sematext/logagent-nodejs-monitor
    MONITORING_TOKEN: <nodejs app token here>
    INFRA_TOKEN: <infra app token here>
    SPM_LOG_TO_CONSOLE: true
```

```
SPM_LOG_LEVEL: error
```

Logagent internal metrics

Logagent keeps track of several internal metrics. Every 60 seconds they are logged to the console by default. These metrics are also shipped to the Node.js monitoring App if it is specified by the `MONITORING_TOKEN`. To enable or disable this use the `printStats` setting.

```
# Global options in logagent.conf
options:
  # Logagent internal metrics are printed to console
  # if a MONITORING_TOKEN is set, the metrics are also shipped
  # to the Node.js App
  # print stats every 60 seconds
printStats: 60
```

Using the Chart Builder you can create a set of charts for Logagent monitoring. Logagent metrics have the `logagent` prefix. The following metrics are available:

- `logagent.tokens.used` - number of logs tokens used
- `logagent.logs.shipped` - number of shipped logs (multi-line)
- `logagent.http.failed` - number of failed http requests for log shipping
- `logagent.http.retranstmit` - number of retransmits from disk-buffer for log shipping
- `logagent.parser.throughput.lps` - parser throughput in lines per second
- `logagent.parser.lines.parsed` - total number of lines parsed in measurement interval