Overview

The client for "SPM for Node.js" is an open-source node.js monitoring agent that collects node.js processes' metrics and sends them to SPM. It is available as npm package that can be added to the JavaScript source code like any other node.js module (see Installation & Configuration section below)

The following metrics are collected and sent to SPM:

- **Operating System**
  - CPU usage
  - Memory usage
- **Process Memory Usage**
- **Number of Worker processes** (when using "cluster" package for master/worker processes)
- **Event Loop**
  - Latencies (fastest, slowest, average)
- **Garbage Collection (GC)**
  - Counters for full GC
  - Counters for incremental GC
  - Time spend for GC
  - Difference in heap used after each GC cycle
- **HTTP Server stats**
  - Request count
  - Request rate
  - Content-Length
  - Error rates (total, 3xx, 4xx, 5xx)
- **Custom Metrics**
  - To track custom metrics like the number of concurrent users, the number of items placed in a shopping cart, or any other kind of business transaction or KPI
    we provide a Custom Metrics API and a node.js client for it: spm-metrics-js

Supported Platforms

- node.js >= 0.10

Installation and Configuration

1. Create an SPM App of type "Node.js" in SPM
2. Click the "**Install Monitor**" button and follow the customized instructions for the created SPM App (basically how to install the NPM package, configure the SPM App Token, and add require statement to your source code to load 'spm-agent-nodejs')

Troubleshooting and "How To"

Generate diagnostics file for Sematext Support

If you are not seeing some or all node.js metrics, you can create a "diagnostics dump" and contact us via chat or email. To create the diagnostics dump just run the following in your application directory:

```
node ./node_modules/spm-agent-nodejs/bin/spm-client-diagnostics.js
```

The output of this script points to the ZIP file and shows the Sematext Support email address to which the ZIP file should be sent.

Using SPM for Node.js behind Firewalls / Proxy servers

By default data is transmitted to SPM via HTTPS. If no direct connection is possible, a proxy server can be used by setting the environment variable HTTPS_PROXY=https://your-proxy.

Installation of native modules on Windows

The native modules are automatically compiled during "npm install" (using node-gyp). On Windows the required build tools like python or C++ compilers are typically not installed by default.
In this case please check https://github.com/TooTallNate/node-gyp for details about the required compiler and build tools.

How to configure spm-agent-nodejs for my app using PM2 process manager

Install spm-agent-nodejs as global module:

```
sudo npm i -g spm-agent-nodejs
```

Check the location (full path) for spm-agent-nodejs using

```
sudo npm root -g
```

The result of the command above is typicall `/usr/local/lib/node_modules` or `/usr/lib/node_modules`. Remember the path to use it in the following step "interpreter_args" in the pm2 configuration file.

If you use PM2 to start your node.js process, then use in your pm2 application config file following environment section:

```
{
  "interpreter_args": "-r /usr/local/lib/node_modules/spm-agent-nodejs"
   "env": {
      "SPM_TOKEN": "YOUR_SPM_TOKEN",
      "spmagent_dbDir": "./spmdb",
      "spmagent_logger__dir": "./spmlogs",
      "spmagent_logger__silent" = false,
      "spmagent_logger__level": "error"
}
```

Upgrading to a new node.js version

If you switch the node.js version (e.g. from 0.12 to 4.4), the spm-agent-nodejs package will need to be installed again (due to the fact that included native modules may change from version to version). After the version change please run a fresh "npm install" if you added spm-agent-nodejs to the dependencies in your package.json - or at the very least run "npm install spm-agent-nodejs"

Upgrading to the latest version of spm-agent-nodejs

To use the latest version of spm-agent-nodejs we recommend you install/upgrade using:

```
npm install spm-agent-nodejs@latest
```

To add the dependency to your package.json simply use:

```
npm install spm-agent-nodejs@latest --save
```

Integration

- Instructions: https://apps.sematext.com/ui/howto/Node.js/overview

Metrics

Metric Name	Key	Agg	Type	Description
heap used	nodejs.heap.used	Avg	Long	

Metric Name	Key	Agg	Type	Description
full gc	nodejs.gc.full	Sum	Long	
inc gc	nodejs.gc.inc	Sum	Long	
total released	nodejs.gc.heap.diff	Sum	Double	
heap total	nodejs.heap.size	Avg	Long	
total duration	nodejs.gc.time	Sum	Double	
memory rss	nodejs.memory.rss	Avg	Long	
workers count	nodejs.workers	Avg	Long	
5xx count	nodejs.errors.5xx	Sum	Long	
request count	nodejs.requests	Sum	Long	
4xx count	nodejs.errors.4xx	Sum	Long	
total res. size	nodejs.response.size.total	Sum	Long	
total req. size	nodejs.requests.size.total	Sum	Long	
min response latency	nodejs.responses.latency.min	Min	Long	
3xx count	nodejs.errors.3xx	Sum	Long	
max response latency	nodejs.responses.latency.max	Max	Long	
error count	nodejs.errors	Sum	Long	
min latency	nodejs.eventloop.latency.min	Min	Long	
max latency	nodejs.eventloop.latency.max	Max	Long	