New Relic Plugin for Unix

System-Level Monitoring for AIX, Linux, Mac OS X & Solaris/SunOS

Table Of Contents

Requirements

- A New Relic account. Sign up for a free account here
- A Unix server that you want to monitor
- Java JRE/JDK v1.6 or later
- Network access to New Relic (proxies are supported, see details below)
- For Dashboard Installation: `curl` or `wget` installed

Supported OSes

- AIX 7
- Linux - All sorts, including on ARM processors (such as Raspberry Pi) and z/Linux
- OS X 10.9 ('Mavericks') and above
- Solaris 10 and 11

Installation & Usage Overview

1. Download the latest version of the agent.
2. Gunzip & untar on Unix server that you want to monitor
3. Set account ID, keys and other settings in `config/plugin.json`
   - Click here for plugin.json config details
4. OPTIONAL: Configure `pluginctl.sh` to have the correct paths to Java and your plugin location
   - Set `PLUGIN_JAVA` to the location of Java on your server (including the "java" filename)
   - Set `PLUGIN_PATH` to the location of the Unix Plugin
5. Run `./pluginctl.sh start` from command-line
6. Check logs (in `logs` directory by default) for errors
7. Login to New Relic UI and find your data in Insights.
   - In the data explorer, look for custom event types that start with "unixMonitor:"
   - Possible event types (for out-of-the-box commands): unixMonitor:Disk, unixMonitor:DiskIO, unixMonitor:NetworkIO, unixMonitor:Process, unixMonitor:Stats, unixMonitor:Vmstat

---

`plugin.json` configuration

Note: A full example of the possible fields in plugin.json can be found in plugin-fullexample.json

Global settings

- `OS` (default: `auto`): Used to determine which commands to run and how to parse them. Leave set to `auto` to have the plugin figure that out (which normally works).
- `account_id`: New Relic account ID - the 6- or 7- digit number in the URL when you're logged into the account of your choosing.
- `insights_insert_key` (under `insights`): You must create an Insights Insert key, as described here.
- `name` (under `agents`): Otherwise, sets the hostname and agentName to whatever is set here. If set to `auto`, the plugin will use that server's hostname.

Proxy settings

If using a proxy, the optional `proxy` object should be added to the `global` object in `plugin.json`, if its not there already. The possible fields are: `proxy_host`, `proxy_port`, `proxy_username` and `proxy_password`, which are self-explanatory. The only field in this object that is required is `proxy_host`.

Dashboard deployment

Enabling at plugin startup

The plugin can check for and deploy the latest dashboards to your account when it starts up. This requires the `dashboards` object in `plugin.json` to be set up properly:

- `admin_api_key`: Admin API key, as described here.
- `integration_guid`: Default is `UNIX.Infra.Monitor`.
- `dashboard_install`: Default is `command_line`.

`integration_guid` and `dashboard_install` are required, but must be left to their default values. **DO NOT CHANGE THESE UNLESS OTHERWISE INSTRUCTED.**

Disabling deployment

If you don't want the dashboard deployment to run at startup, leave `admin_api_key` blank or remove it entirely.

Other configurations

Deploying Dashboards from separate server/desktop

If you want to initiate the dashboard install from a standalone machine (i.e. a tools server or your own mac, linux or cygwin laptop/desktop), you will need the following:

- `pluginctl.sh`
- `config/plugin.json` (including path) with the dashboard object filled out as described above.

To install, run `./pluginctl.sh installDashboards`.

Fix for using the WebSphere JDK

If you are using the JDK that is packaged with WebSphere and see an exception in the logs like below, it is due to attempting to use the WebSphere SSL Factory instead of the IBM JSSE packages.

```
ERROR com.newrelic.metrics.publish.binding.Request - An error occurred communicating with the New Relic service
java.net.SocketException: java.lang.ClassNotFoundException: Cannot find the specified class com.ibm.websphere.ssl.protocol.SSLSocketFactory
```

If so, uncomment the following line in `pluginctl.sh` and restart the plugin.

```
# USE_IBM_JSSE=true
```