Datadog

Level 1 - Collecting your Data

The Datadog agent runs in the background and gathers all the metric data that is displayed in Datadog dashboards and configured to trigger alerts if required. It needs to be configured so it knows what data it needs to gather and for what applications. By default, basic information such as memory and cpu utilisations are captured.

These instructions assume you have the Datadog agent installed as well as the Python pip (package manager).

Add Server Tags

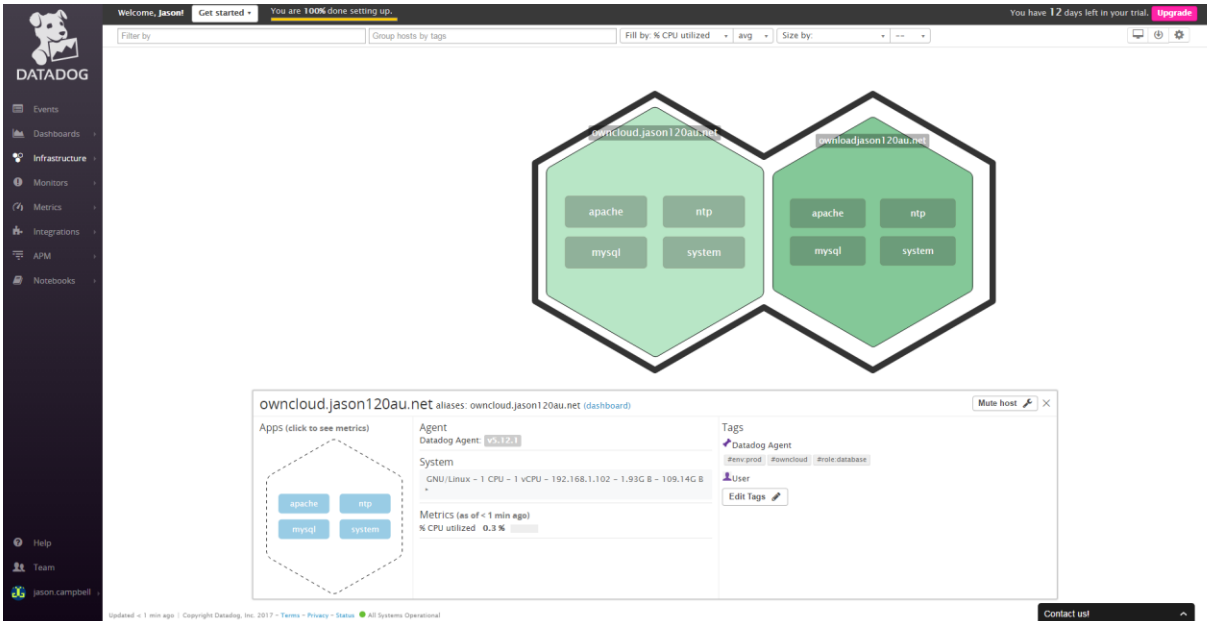
Edit the follow file /etc/dd-agent/datadog.conf find tags and uncomment the line (remove the # at the start).

```
# Set the host's tags (optional)
 tags: owncloud, env:prod, role:database
```

Save the file and restart the dd-agent.

```
/etc/init.d/datadog-agent restart
```

Hostmap



Welcome, Jason! You are 100% done setting up. You have 12 days left in your trial.

Filter by | Group hosts by tags | Fill by: % CPU utilized | avg | Size by: | --

owncloud.jason120au.net aliases: owncloud.jason120au.net (dashboard)

Apps (click to see metrics)

Agent
Datadog Agent: v5.12.1

System
GNU/Linux – 1 CPU – 1 vCPU – 192.168.1.102 – 1.93G B – 109.14G B

Metrics (as of < 1 min ago)
% CPU utilized 0.3 %

Tags
Datadog Agent
#env:prod #owncloud #role:database
User

Updated < 1 min ago | Copyright Datadog, Inc. 2017 – Terms – Privacy – Status ● All Systems Operational

Installing MySQL (MariaDB)

Tip: In order to use the latest version of Maria-DB setup your system to use the Maria-DB repository. Create a file named MariaDB.repo in /etc/yum.repos.d containing the following

```
# MariaDB 10.1 CentOS repository list - created 2017-04-02
09:30 UTC

# http://downloads.mariadb.org/mariadb/repositories/

[mariadb]

name = MariaDB

baseurl = http://yum.mariadb.org/10.1/centos7-amd64

gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB

gpgcheck=1
```

Install MySQL if using Centos 7 by default will use MariaDB which is a fork of MySQL. Commands to install the database are as follows:

```
sudo yum install mariadb mariadb-server
```

More detailed instructions are here

MariaDB may install by default to a setup which will not be set to listen on port 3306 for client connections. To fix this open the /etc/my.cnf file comment out(ie put a # infront) skip-networking and restart the database(systemctl mariadb restart). This will ensure that the Datadog agent can connect successfully to the database.

The next step will be to ensure that Python has the relevant libraries and dependencies so the Datadog agent can communicate metric data. Installing the Python MySQL library is required by running the following command.

```
sudo yum install MySQL-python
```

Next run the following command so python MySQL libraries are configured.

```
pip install pymysql
```

Restart the Datadog agent by running the following command

```
/etc/init.d/datadog-agent restart
```

Check that the Datadog agent is gathering MySQL data correctly by running the following command.

```
/etc/init.d/datadog-agent info
```

Search for MySQL in the output. The following should be shown if it is setup correctly.

```
mysql (5.12.1)

    --------------

      - instance #0 [OK]

      - Collected 65 metrics, 0 events & 1 service check

      - Dependencies:
          - pymysql: 0.6.6.None
```

Setting up a Random Number Generator Metric

To setup a custom metric you will need to create a simple Python application which just generates a random number and stores the result in a metric so the data agent can pick up the data. Source code for the Python application should be placed in /etc/dd-agent/check.d and /etc/dd-agent/conf.d will contain information about how the agent should use the check. Source code is in (test.support.random.py) and the configuration must be named the same however will have the yaml extension. To check if the metrics are reporting correctly run the following command.

```
sudo -u dd-agent dd-agent check test.support.random
```

If successful the following will be displayed.

```
test.support.random (5.12.1)

    ----------------------------

      - instance #0 [OK]

      - Collected 1 metric, 0 events & 0 service checks
```

Restart the Datadog agent for the changes to take effect.

```
/etc/init.d/datadog-agent restart
```

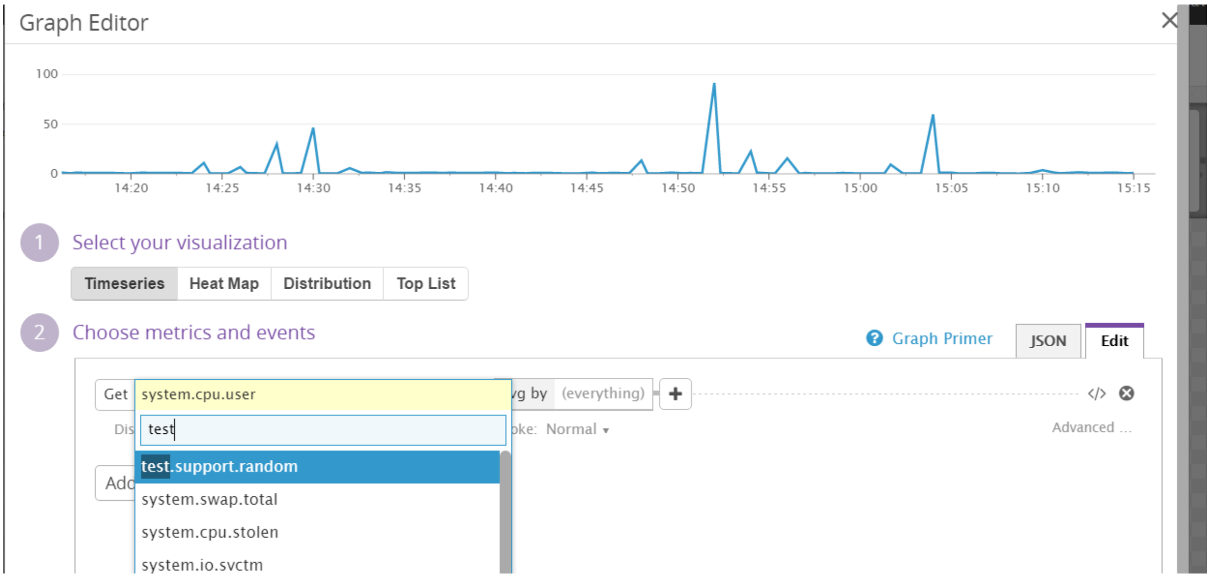
Setting up MySQL Monitoring

In order to setup MySQL monitoring on Centos you will need to setup a new MySQL user with specific permissions for the monitoring. Specific information is available here.

Level 2 - Visualizing your Data

Database dashboard

To setup a dashboard go to the Datadog management page(datadoghq.com and select logon from the right hand side) and click on Dashboards and click New Dashboard. Select whether you want a Timeboard or a Screenboard. Timeboards a best for viewing metrics based on time for example how many MySQL connections have been made over the course of a day. The Screenboard dashboard is used primarily to manage status changes with systems. To add the random number generator as a metric drag and drop the graph widget to the dashboard. Select the test.support.random metric from the list as shown below.



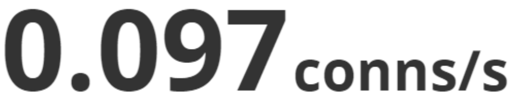
Get system.cpu.user

test

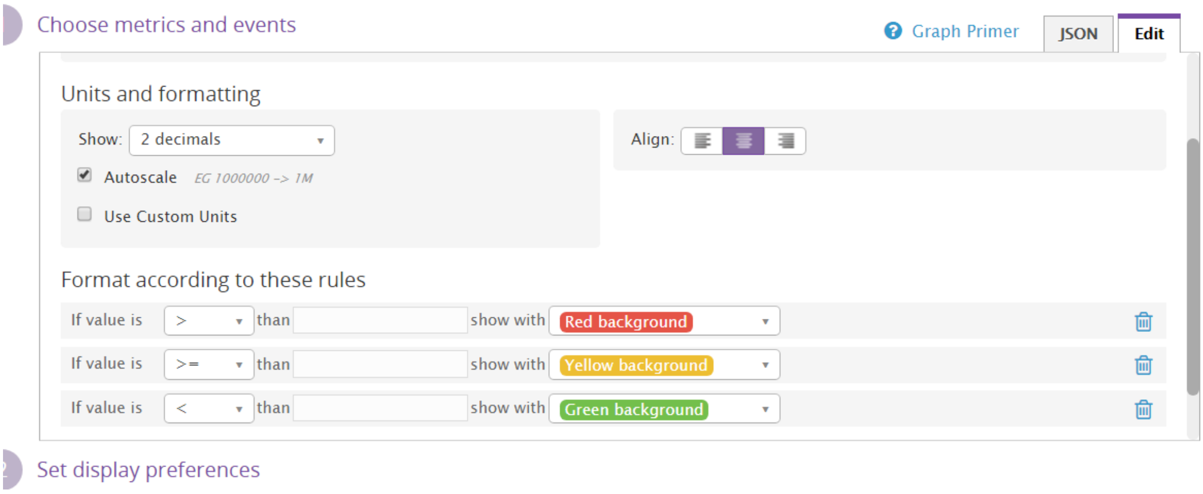
- test.support.random
- system.swap.total
- system.cpu.stolen
- system.io.svctm

Add a marker to the graph so when the metric is above 0.80 a yellow line is displayed on the graph and add another marker when the graph is above 0.90. Click the box to add a label to the graph.

You can similarly add MySQL metrics to the dashboard by selecting any of the MySQL metrics from the list. For example to display the current connection count to the MySQL database you can drag and drop the “Query value” widget to the dashboard and select the mysql.net.connections. Scroll down to *format according to these rules*. Type values in the if value is > sections as shown below under the specified colours.



0.097 conns/s



Units and formatting

Show: 2 decimals

☑ Autoscale *EG 1000000 -> 1M*

☐ Use Custom Units

Format according to these rules

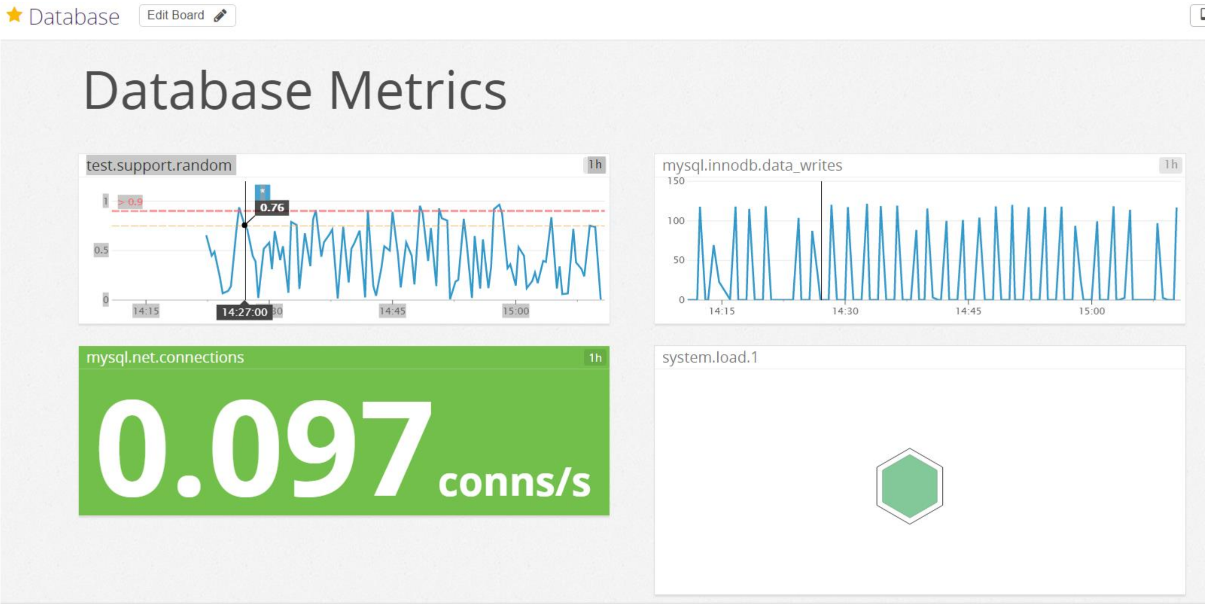
If value is > than show with Red background

If value is >= than show with Yellow background

If value is < than show with Green background

Find below the database dashboard with random number generator graph and other database metrics. You can setup the dashboard so it is public accessible and have done so and the below is the URL and screenshot. I have setup the test.support.random metric so if it is above 90% it will be displayed clearly on the graph with a red line. If it is above 80% a yellow line is shown on the graph.

https://p.datadoghq.com/sb/44ef2b50b-23d0fd613d



Question: what is the difference between a timeboard and a screenboard?

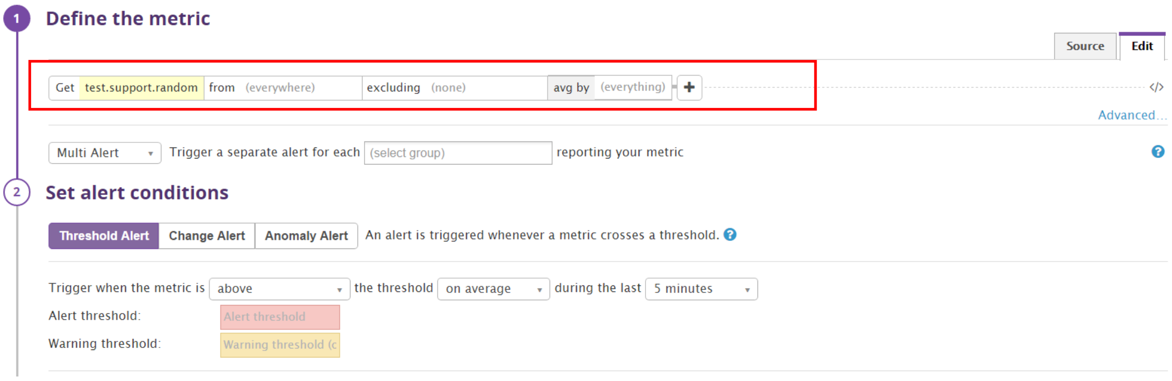
A timeboard Is good to share time based metrics for example how many connections your database has received over a period of time, whereby a screenboard is good for status boards and sharing data with multiple parties.

Level 3 - Alerting on your Data

Setting up a Monitor

1. Login to the Datadog control manager at datadoghq.com and clicking logon on the right-hand side of the window
2. Click Monitors | New Monitor on the left-hand menu
3. Select the Metric monitor type
4. Under *1. Define the metric*, click “select a metric”, type the metric you want to monitor in the case it’s test.support.random. (Tip: you can just type the first few letters of the metric you are looking for and then select the autocomplete item).

5. In the from section select “env prod” or a tag you have created previously. Leave everything else as is and select multi alert and trigger a separate alert for each and select host from the drop down menu.



1 Define the metric

Get test.support.random from (everywhere) excluding (none) avg by (everything) +

Multi Alert Trigger a separate alert for each (select group) reporting your metric

2 Set alert conditions

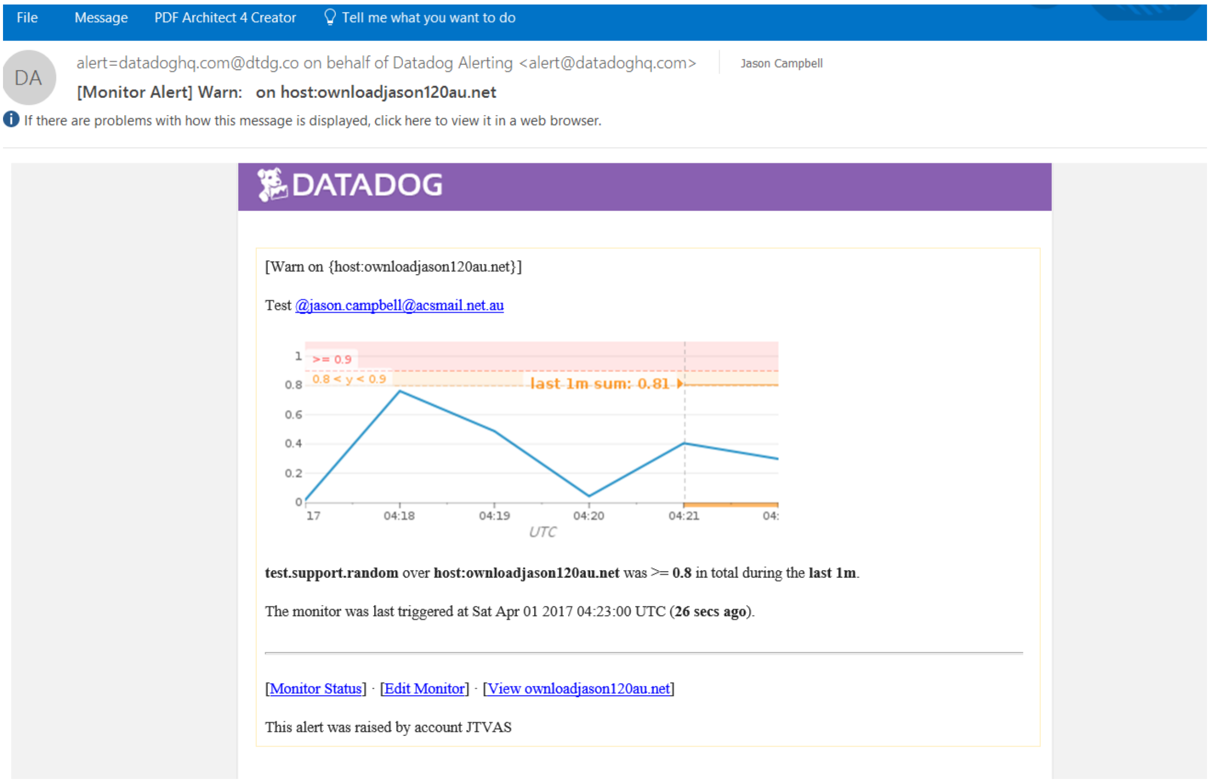
An alert is triggered whenever a metric crosses a threshold.

Trigger when the metric is above the threshold on average during the last 5 minutes

Alert threshold: Alert threshold

Warning threshold: Warning threshold (c

6. 2. *Set alert conditions*, in *alert threshold* type 0.90, set *during the last* to 15 minutes and keep the other settings as per the defaults
7. For warning threshold type 0.80 and keep other settings as they are
8. 3. *Say what’s happening* Type the information you would like the alert to say when it is triggered. Tip use message template variables so you can specify a different subject if the alert is on error or recovering from it. For example, to display text only when the alert is a recovery use the following syntax.
`{{#is_alert_recovery}} Recovered alert {{/is_alert_recovery}}`
9. 4. Notify the Team, click in the input box and a drop-down menu will appear with a list of people the alert can be sent to. You can also type in an email. Alerts look like this.



alert=datadoghq.com@dtdg.co on behalf of Datadog Alerting <alert@datadoghq.com>   Jason Campbell

[Monitor Alert] Warn: on host:ownloadjason120au.net

If there are problems with how this message is displayed, click here to view it in a web browser.

[Warn on {host:ownloadjason120au.net}]

Test @jason.campbell@acsmail.net.au

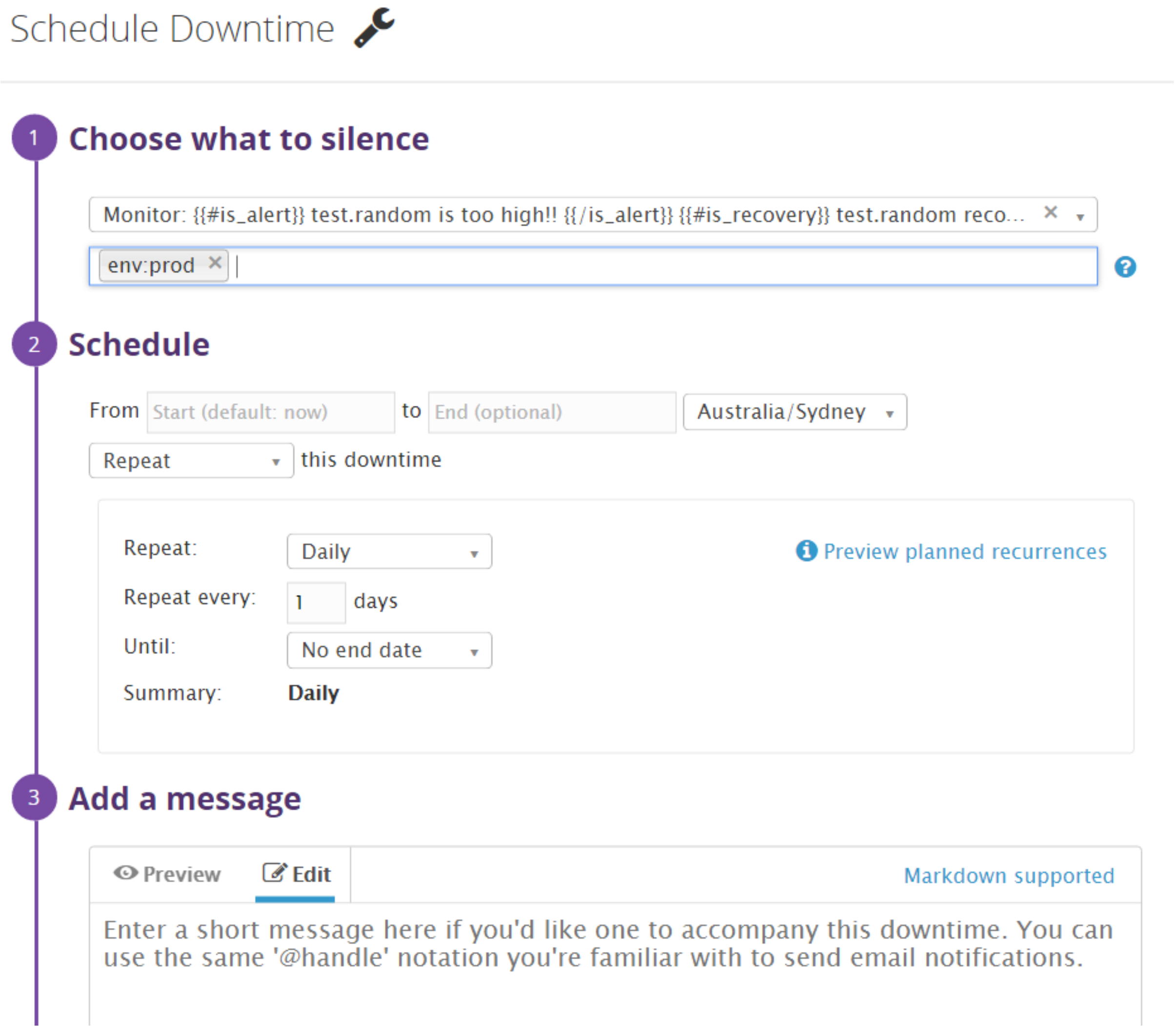
test.support.random over **host:ownloadjason120au.net** was >= **0.8** in total during the **last 1m**.

The monitor was last triggered at Sat Apr 01 2017 04:23:00 UTC (**26 secs ago**).

[Monitor Status] · [Edit Monitor] · [View ownloadjason120au.net]

This alert was raised by account JTVAS

10. If you only want the alert to be sent during business hours (and silenced at other times) you can select a downtime period. To do this click “Manage Downtime” at the top of the page and click schedule downtime. Select the monitor you just created, select the scope of servers for example *env prod* select the times you would like the downtime to occur for example between 6pm and 9am, select the timezone, ie Australia/Sydney and select repeat daily. See screenshot on next page.



Schedule Downtime

1 Choose what to silence

Monitor: {{#is_alert}} test.random is too high!! {{/is_alert}} {{#is_recovery}} test.random reco...

env:prod

2 Schedule

From Start (default: now) to End (optional) Australia/Sydney

Repeat this downtime

Repeat: Daily

Repeat every: 1 days

Until: No end date

Summary: **Daily**

Preview planned recurrences

3 Add a message

Preview Edit Markdown supported

Enter a short message here if you'd like one to accompany this downtime. You can use the same '@handle' notation you're familiar with to send email notifications.