

RPi: Building a Raspberry Shake seismogram on the Node-Red Dashboard

Goal:

Set up a real time Seismogram on the Node-Red Dashboard.

Resources:

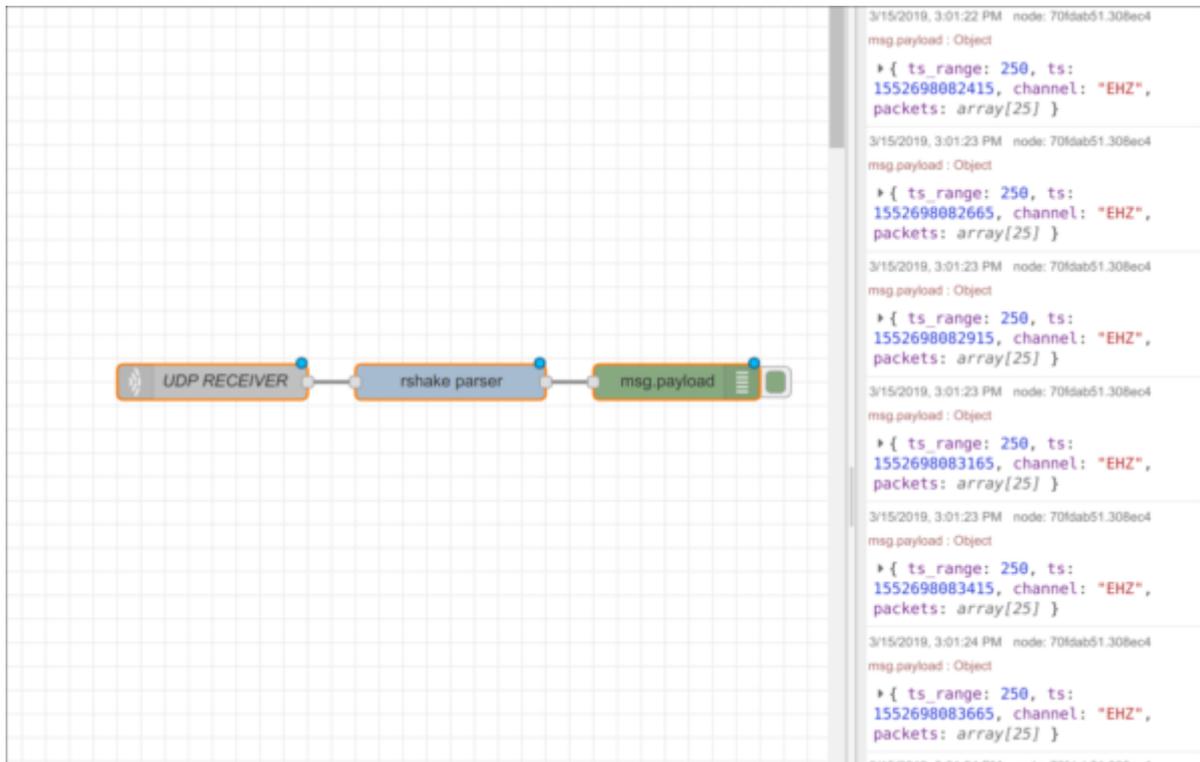
- Raspberry shake, setup following this guide: <https://t3alliance.org/rpi-setting-up-a-raspberry-shake/>
- A separate raspberry pi with node-red, running on the same network
- If you have not already connected the Raspberry Shake to Node-Red, then follow this tutorial to do so: <https://t3alliance.org/rpi-streaming-raspberry-shake-data-to-node-red/>

Introduction:

After following the first tutorial on connecting Shake to Node Red, now we can start to use Node-Red to process and visualize the seismic data that is being streamed through the UDP port.

Getting Started:

Before starting this tutorial, make sure you have already completed the [“Streaming Raspberry Shake Data to Node-Red”](#) tutorial. Your flow should look like this:



The screenshot shows a Node-RED flow on a grid background. The flow consists of three nodes connected in a line: a grey 'UDP RECEIVER' node, a blue 'rshake parser' node, and a green 'msg payload' node. To the right of the flow is a console window displaying a series of log messages. Each message is a JSON object with the following structure:

```
{ ts_range: 250, ts: 1552698082415, channel: "EHZ", packets: array[25] }
```

 The console shows five such messages, each with a different 'ts' value (1552698082415, 1552698082665, 1552698082915, 1552698083165, 1552698083415) and a timestamp of 3/15/2019, 3:01:23 PM.

Make sure that there is UDP data being received by checking the debug tab. As you can see, the data coming through the Rshake Parser is still in object form, consisting of multiple types of information. For building the siemogram graph, we just want the numerical data.

First, we have to add a switch node that discriminates the the channel of the data. If we are using a Shake 1D, it will be the channel EHZ. So, we must add a switch node after the parser, and format it to be listening for the property “msg.payload.channel” and underneath we can add the rule as: == “string” EHZ. This separates the data that we want.

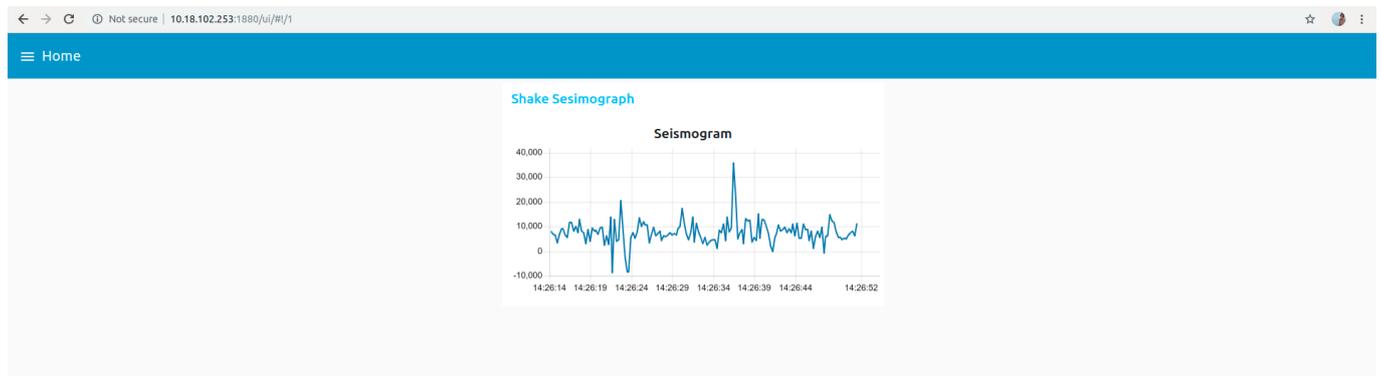
Connect the ‘format plot values’ node to the switch node. Now, to finish we connect a Dashboard chart. Your flow should look similar to this:



Then, configure the chart in the way that you would like, and Deploy!

Now, open your local Node-Red Dashboard! You can access this page by entering your Node-Red address /ui. For example: `http://10.18.102.253:1880/ui`

Now, bang the table! Jump up and down! See what happens to the graph!



To continue working with the Shake Data in Node-Red, go to the next Tutorial!

Making

a

Shakemeter: <https://t3alliance.org/rpi-building-a-shakemeter-with-raspberry-shake-and-node-red/>