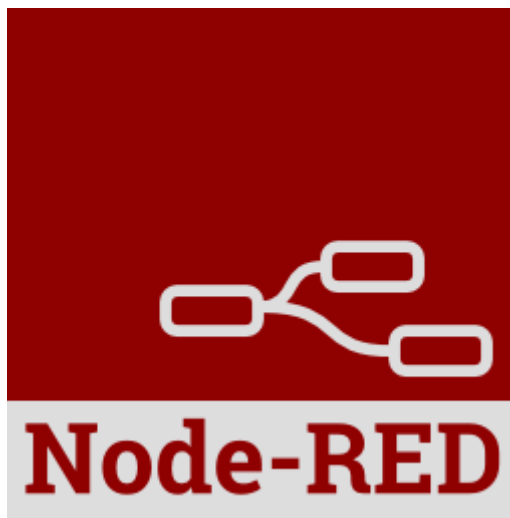


Node-RED dashboard User Manual

Getting started

<https://nodered.org/>



Node-RED is a visual wiring tool for the Internet of Things. A project of the JS Foundation (<https://js.foundation/>).

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

The example below is based on the tutorial <http://noderedguide.com/lecture-7-node-red-dashboard-part2/>

The examples described here are available at the following link: <https://iot-app.snap4city.org/nodered/nr10/ui>

1. Getting Started by creating a first basic dashboard

In this example we will create a simple flow that sends a random number between 0 and 99 to a simple chart. For that you'll need an inject node to repeatedly fire every few seconds, a function node to generate the random number and one of the node-red-dashboard nodes – in this case the chart node.

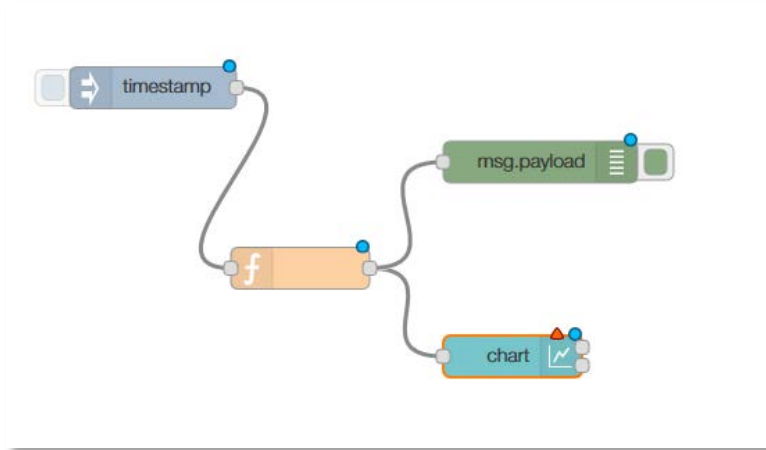
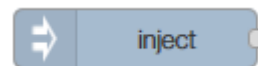


Fig. 1 - Our first example

Drag and drop the following nodes from the left column in the main area:

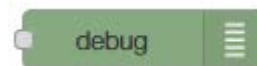
- **Inject** node (from the *input* nodes list)



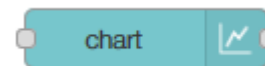
- **Function** node (from the *function* nodes list)



- **Debug** node (from the *output* nodes list)



- **Chart** node (from the *dashboard* nodes list)



The “inject” node will change the name in “timestamp” and the “debug” node will change the name in “msg.payload”. Connect the nodes by clicking the small rounded square of each node to have the same result as in the Figure 1 above.

Before we look a how the chart node works, let's configure the inject node to send a timestamp every 5 seconds by setting the *payload* to timestamp and the *repeat* field to an interval of 5 seconds. Double click the timestamp node and set the parameters as in the image below.

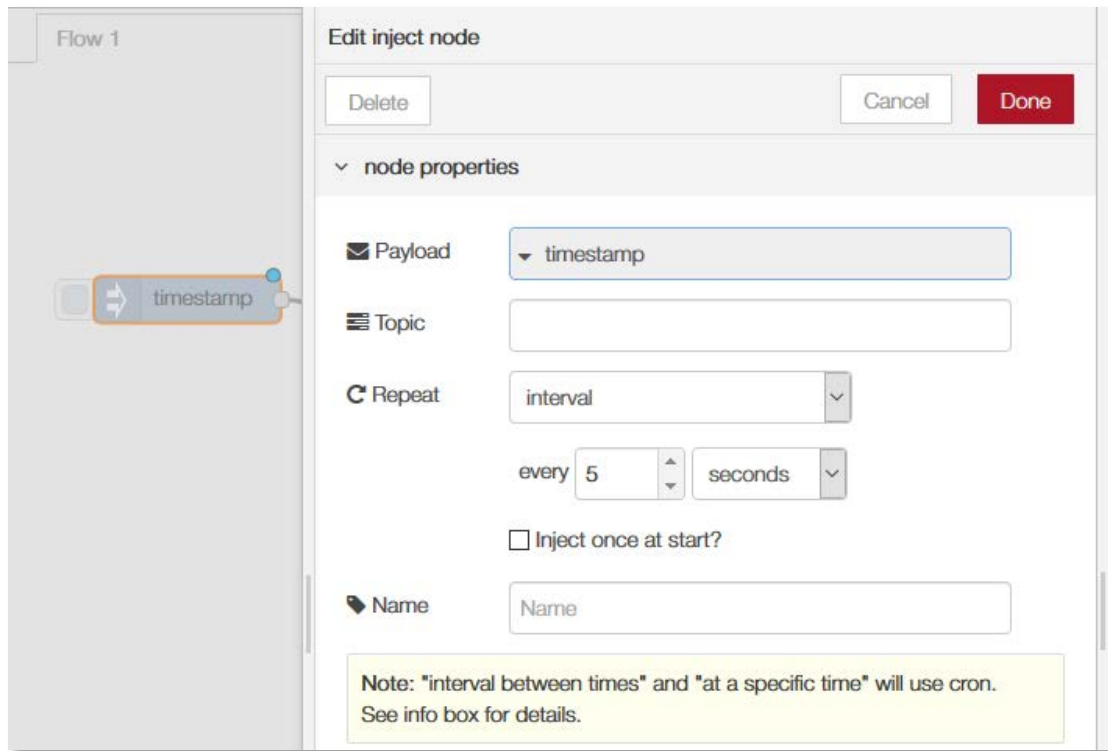


Fig. 2 – Edit *inject* node

Click the **Done** button to confirm. A small circle arrow will appear after the *timestamp* name.



This will act as a repeating trigger. Now we need to set up the function node to generate a random number – we'll use a simple JS math function to do this:

```
msg.payload = Math.round(Math.random() * 100);  
return msg;
```

Double click on the **function** node and:

- Insert the name “Random number”
- Copy the JS math function in the “**Function**” empty field
- Click **Done** to confirm

The final result is shown in Fig. 2. This will generate a random number between 0 ~ 99 which is passed to the chart node.

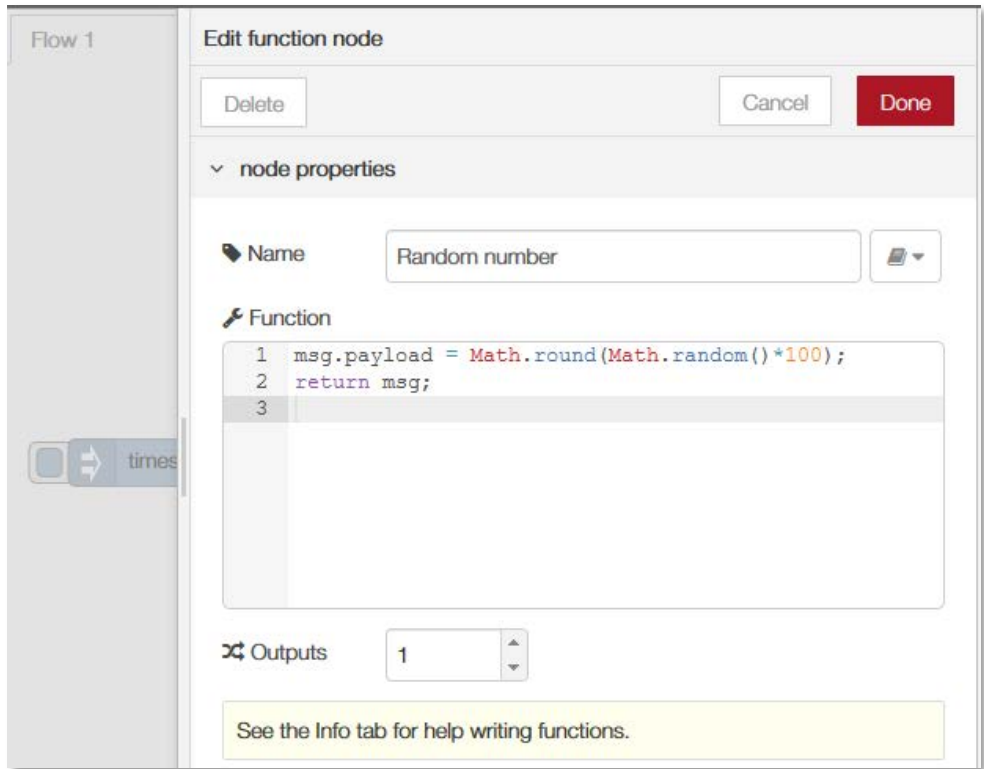


Fig. 3 - Edit function node

Double click the **chart** node to enter in the configuration options. Set the options as in the Fig. 4.

Click the **Group** field button to add and configure a dashboard group (fig. 5).

The **Tab** option allows you to specify which tab of the UI page you will see the UI element on – in this case our chart. The default tab is Home – which we are using here. If you select the edit button to the right of the Tab field you can create a new tab and then select that. However, we'll use the default home for now.

The **Name** field is the standard Node-RED node name – by default this is chart but you can set it to anything you like.

The **Group** field allows you to group UI elements – we'll show you how that works when we add another UI element so let's use group "My first dashboard[Home]" for now – of course, you can use any string you like.

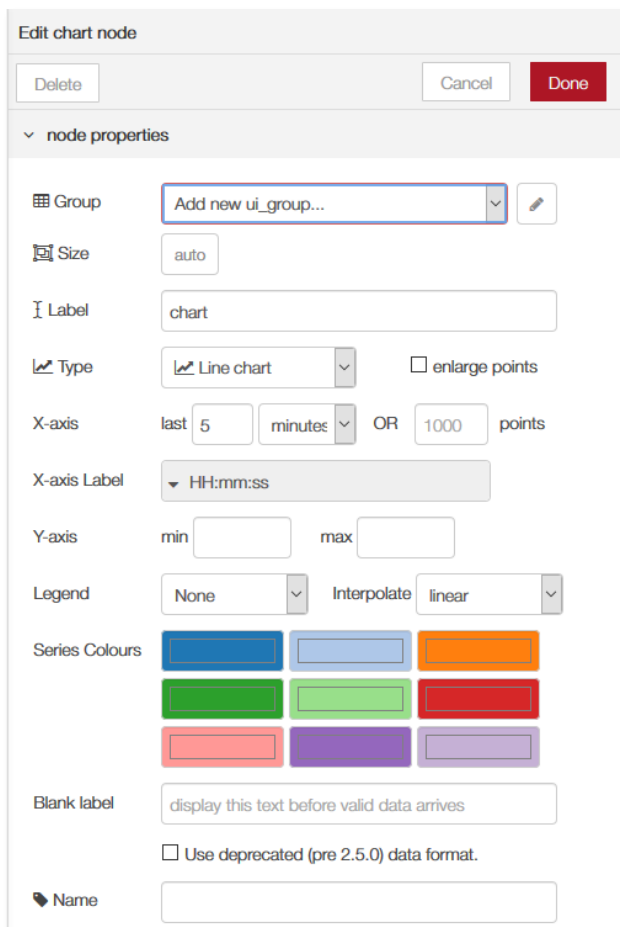


Fig. 4 - chart node properties

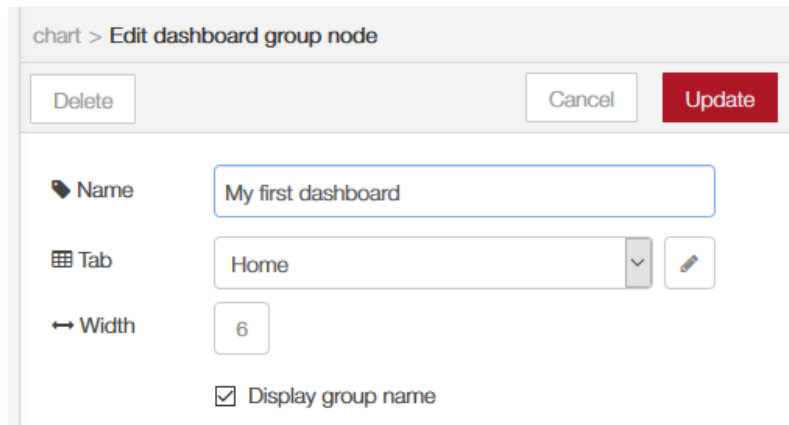


Fig. 5 - Edit dashboard group node

Click the “**Update**” button to confirm the settings and to return to the **Edit chart node** properties (Fig. 5).

The *X-axis* field allows you to tell the chart how much data it should store and display – the longer the ‘*last*’ field is set to, the more data is stored and displayed by the chart. Let’s use a short 5 mins which will start to throw away the data that is 5 minutes old.

Lastly the *Interpolate* field defines how the chart will interpolate values in between actual data values it receives, you can select between linear, step, b-spline and cardinal – which are standard [interpolation algorithms](#). We’ll use the default linear.

Wire these nodes up, hit the **Deploy** button – check that your debug node is showing that random values are showing. Then head over to your default dashboard page to see the results. By default, you’ll see your UI at:

<https://{your nodered website }/ui/>

When you visit that page you’ll see your initial chart as shown below:

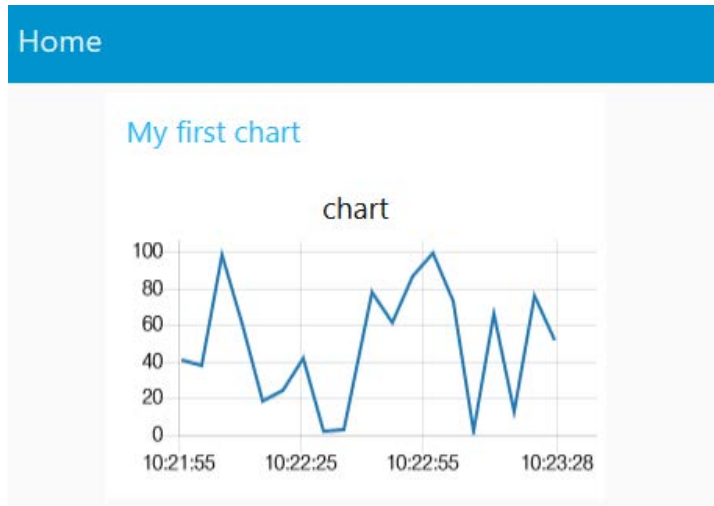


Fig. 6 - The final result

2. Add other elements in the UI

Now we will create a gauge to show the last data value sent. Drag a **Gauge** node available in the **dashboard** list of nodes from the UI palette and wire it to the **Random Number function** node.

You will have a result similar to Fig. 7

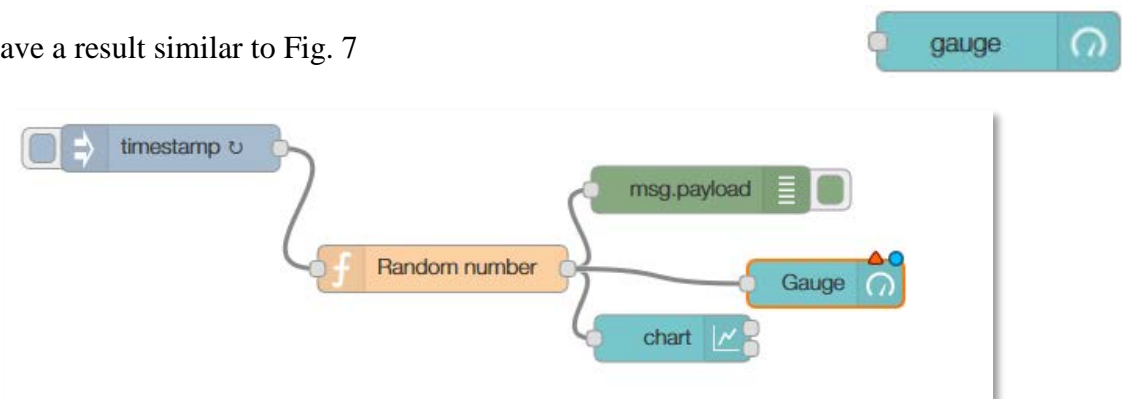


Fig. 7 - Add a new gauge node

Then double click the **Gauge** node to open up and let's configure it as in Fig. 8 and click the **Done** button to confirm.

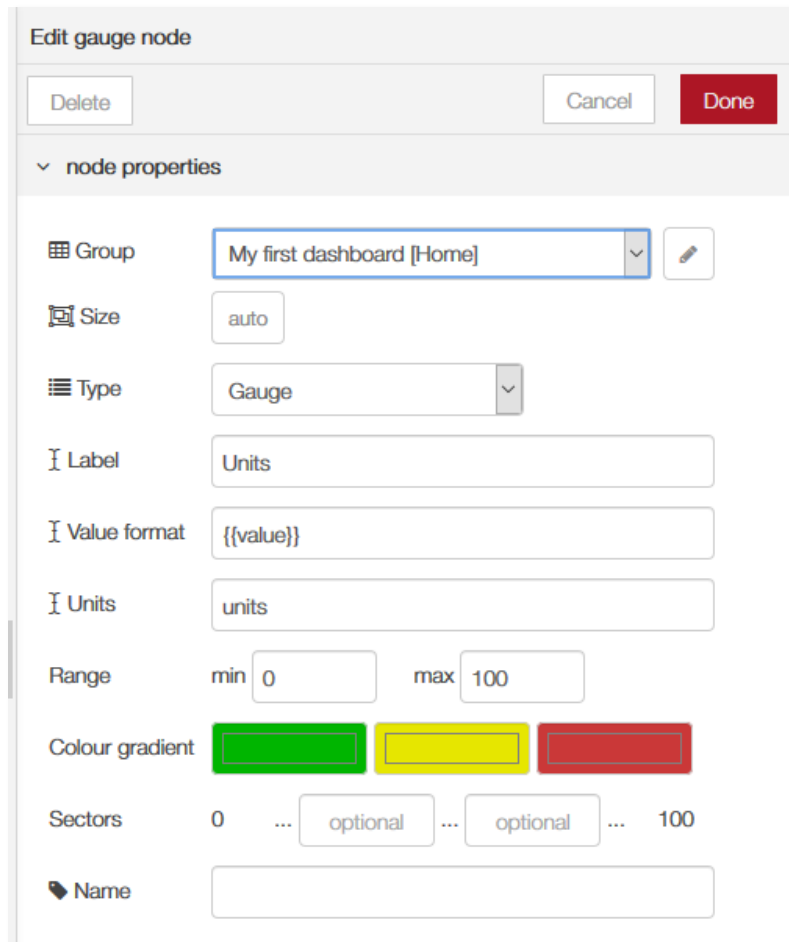


Fig. 8 - Edit Gauge node

We'll use the same Tab, home and we'll also add it to the same group – “My first dashboard[Home]”. The Min and Max fields allow you to set the min and max values the gauge will show. Make sure the max is set to 100 which is the most that the random number function node will generate. You can also change the *Colour gradient* to show different colours on the widget, but we will leave it as default for now.

Hit **Deploy** and then head over to your dashboard and you'll see that the chart and gauge are displayed in a group with the chart now showing the last 5 minutes of data and the gauge the latest value.

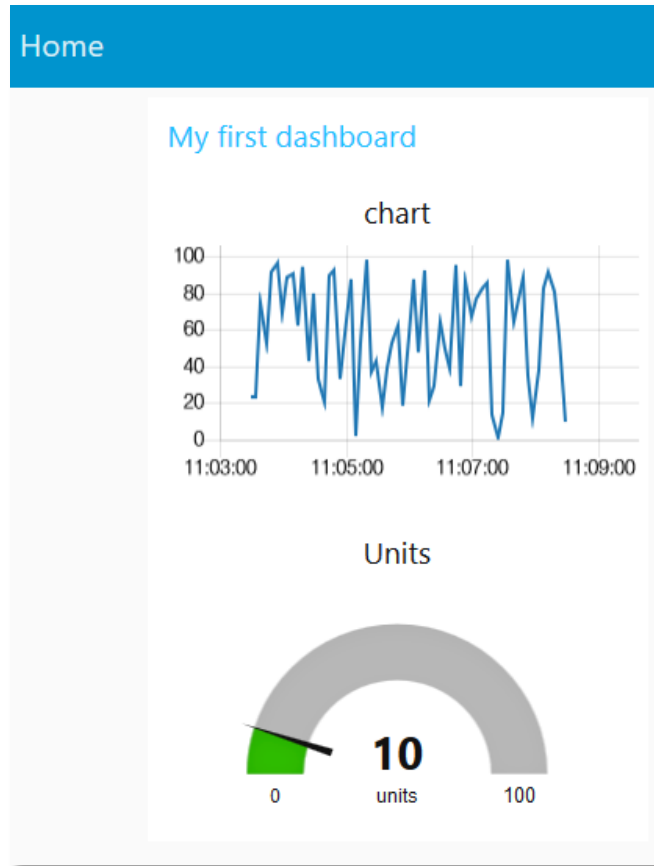


Fig. 9 - New gauge added

Now we will add a couple of other UI nodes, a **slider** node and a **text** node to show the same data on a slider and as a text string.

- Drag the slider node and the text node from the dashboard list of nodes (Fig. 10)

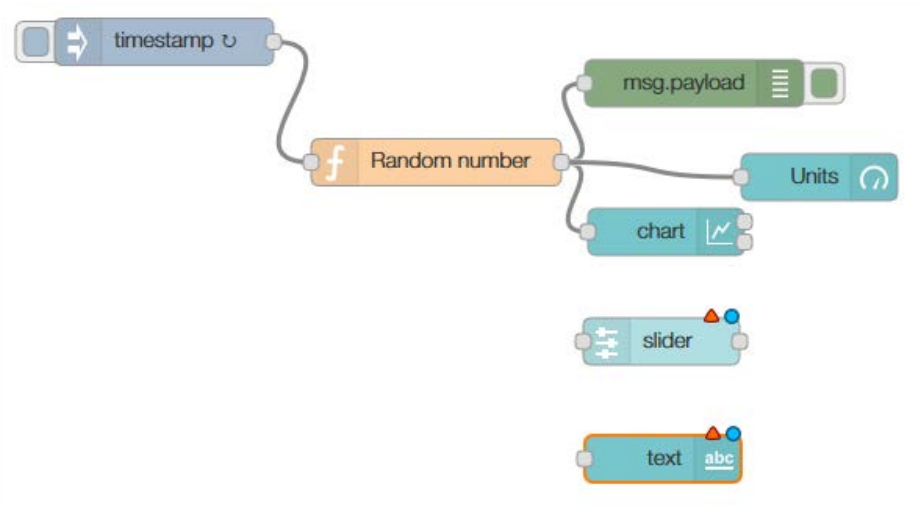


Fig. 10 - 2 new nodes added (slider and text)

- Connect the Random number node to the two new nodes (Fig. 11).

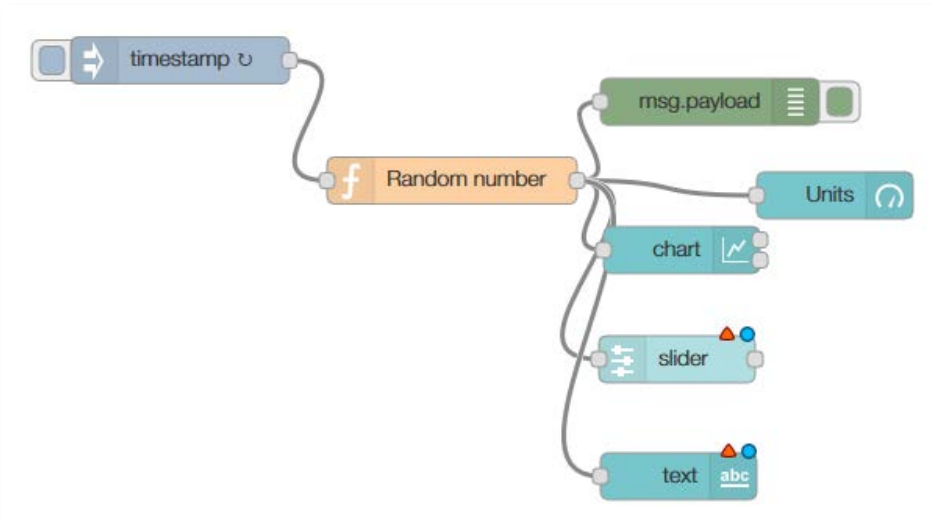


Fig. 11 - the two new nodes (slider and text) are now connected to the Random number node.

- Double click the **slider** node and configure a new group name. You need to click “Add new UI_group” icon of the **Group** field, and then click the edit button (Fig. 12).

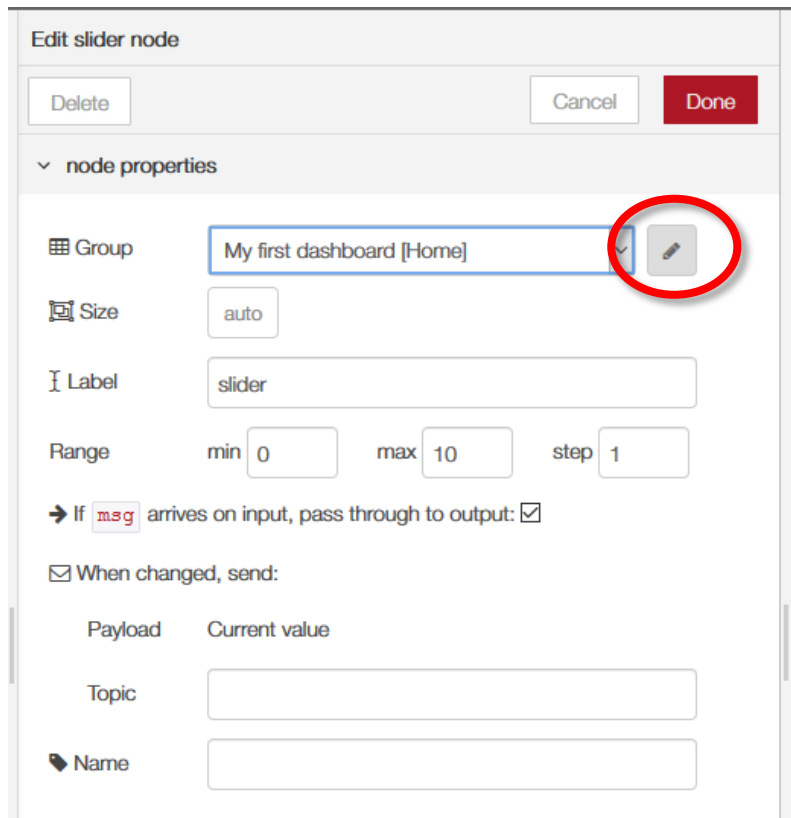


Fig. 12 - Add new ui_group

Write a new group name, for example *new_group* and confirm. Set the *new_group* for the new nodes **slider** (Fig. 13) and **text** (Fig. 14) as Group. Click **Done** button to confirm.

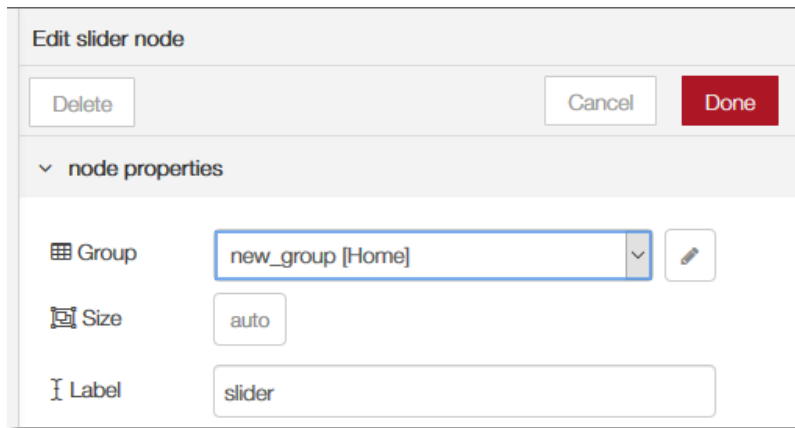


Fig. 13 - new group association (slider node)

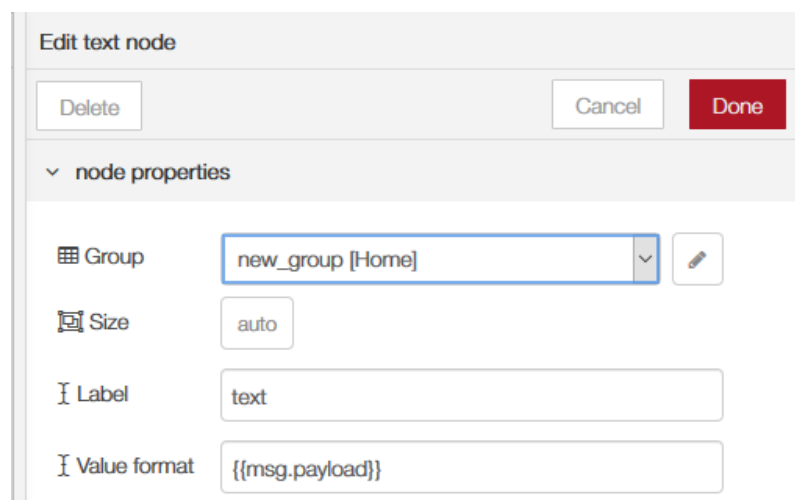


Fig. 14 - new group association (text node)

You will also need to change the **max** value to 100 for the **slider** node to show the correct position of the slider.

Deploy the new dashboard. The final result should be as in Fig. 15.

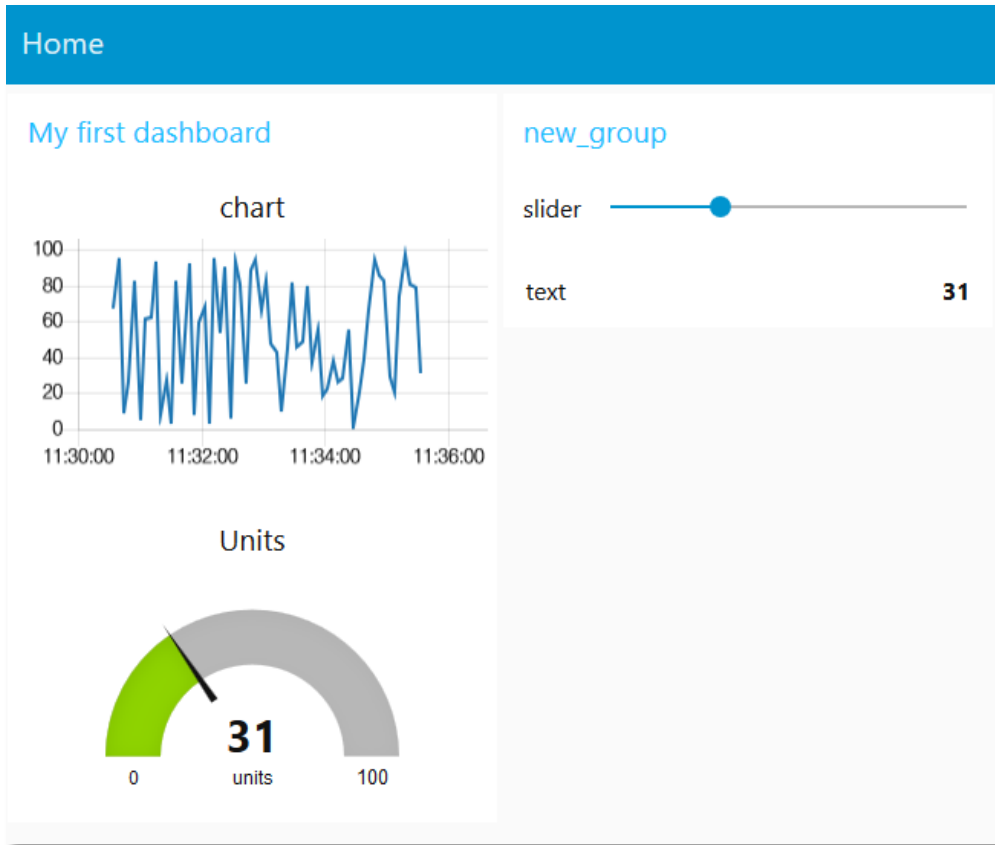


Fig. 15 - dashboard with two new nodes (slider and text) in a new column (new_group)

In the **dashboard** tab beside your **debug** tab, you can also set the theme and order of the elements. If you don't see the dashboard tab, click the menu button at top right corner, then select "View" -> "Dashboard". You can see all the widgets and tabs showing in a tree structure, and you can easily drag the elements to change the orders that they are presented in the dashboard (Fig. 16).

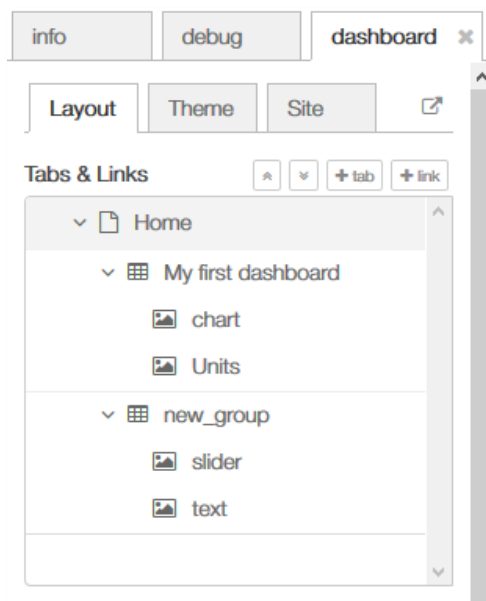


Fig. 16 - Tab and Links pane