# Explore Physical Web with Kitra 520

Google has an interesting initiative called the Physical Web. It is an approach to quickly enable interaction with physical objects. For example, a vending machine may broadcast a URL that redirects you to a payment system. A burger shop can send out a link to customers in line for online ordering.

The Physical Web lets a Bluetooth low energy devices broadcast beacons, typically a short web address, so that people in the radio range can pick up the beacons with their phones. BLE beacons broadcast over Eddystone protocol, specifically Eddystone-URL for URL broadcasting.

In our tutorial, we will use Kitra520 to set up a physical web beacon. You can follow their Getting Started Guide to set up Kitra board.

## Broadcast a simple Physical Web beacon

### Broadcast  a URL from Kitra 520

We will start from a very basic example, that uses Kitra 520 to broadcast a product URL. This is so simple that you can do it by running the 3 commands below. Thanks to this blog, that you can easily turn a URL into a BLE beacon format.

```
[root@localhost ~]# hciconfig hci0 up
[root@localhost ~]# hciconfig hci0 leadv 3
[root@localhost ~]# hcitool -i hci0 cmd 0x08 0x0008 16 02 01 06 03 03 aa fe 0e 16 aa
fe 10 00 00 61 72 74 69 6b 2e 69 6f 00 00 00 00 00 00 00 00
< HCI Command: ogf 0x08, ocf 0x0008, plen 32
  16 02 01 06 03 03 AA FE 0E 16 AA FE 10 00 00 61 72 74 69 6B
  2E 69 6F 00 00 00 00 00 00 00 00
> HCI Event: 0x0e plen 4
  01 08 20 00
```

- The first command enables Kitra520 as a Bluetooth device.
- The second command activates advertising on the device, at the same time, it doesn't allow connections to the advertised service.
- The third command configures the advertising data to be sent. In our case, it is a URL: www.artik.io

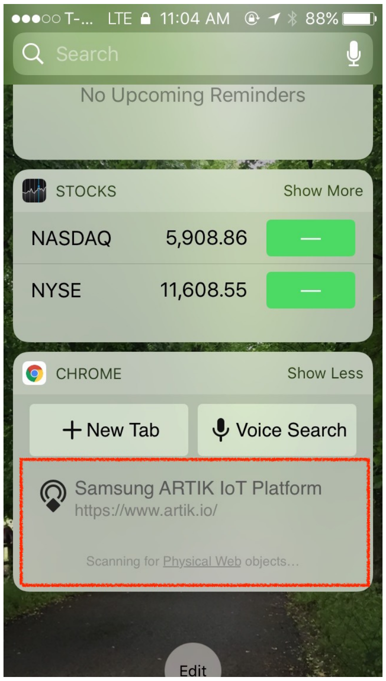
### Detect beacons with your phones

Android/iOS phone users with latest firmware images can install Chrome, which automatically posts a notification whenever a Physical Web beacon is in range. Please follow the instructions here to configure your Chrome.

For users with older phone firmware, you can install Physical Web Application on your Android and iOS.

**Note**: Chrome's Physical Web implementation only supports HTTPS addresses. HTTP URLs will not be displayed.

Here is the notification I received on my phone. Click on it will launch ARTIK website.



●●●○○ T-... LTE 11:04 AM 88%

No Upcoming Reminders

STOCKS

| | |
|---|---|
| NASDAQ | 5,908.86 |
| NYSE | 11,608.55 |

CHROME

Samsung ARTIK IoT Platform
https://www.artik.io/

Scanning for Physical Web objects...

## Real-time sensor monitoring

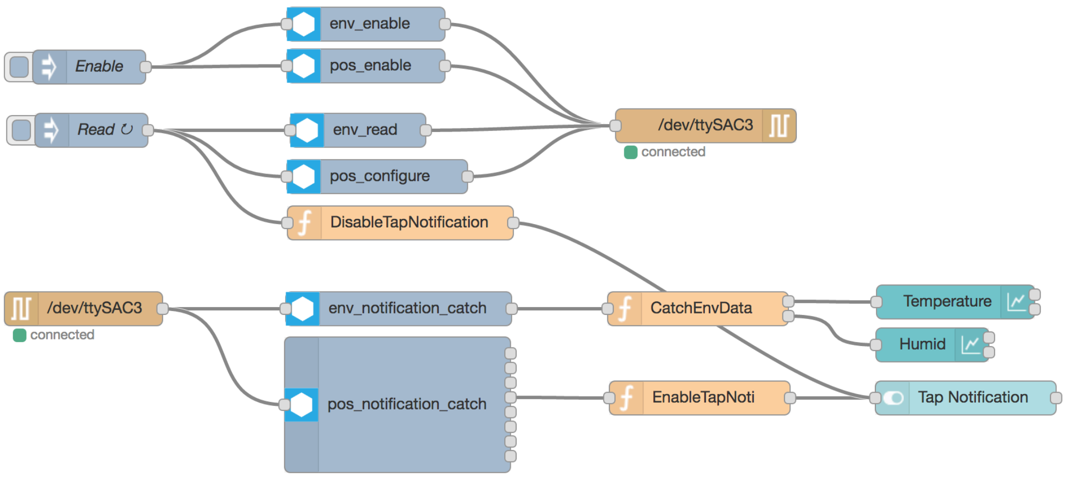
To make our Physical Web more meaningful, we are going to collect the sensor data from a Kitra520 and advertise a web page with the aggregate information. In a real-life scenario, you can mount your Kitra to a machine, an appliance or any equipment for proximity-based real-time sensor monitoring.

### Create sensor dashboard

Kitra's Getting Started guide explains in detail how to use Node-RED to capture built-in sensor data. I am extending it by displaying the sensor data onto a dashboard.  Install node-red-dashboard package in order to build a dashboard.

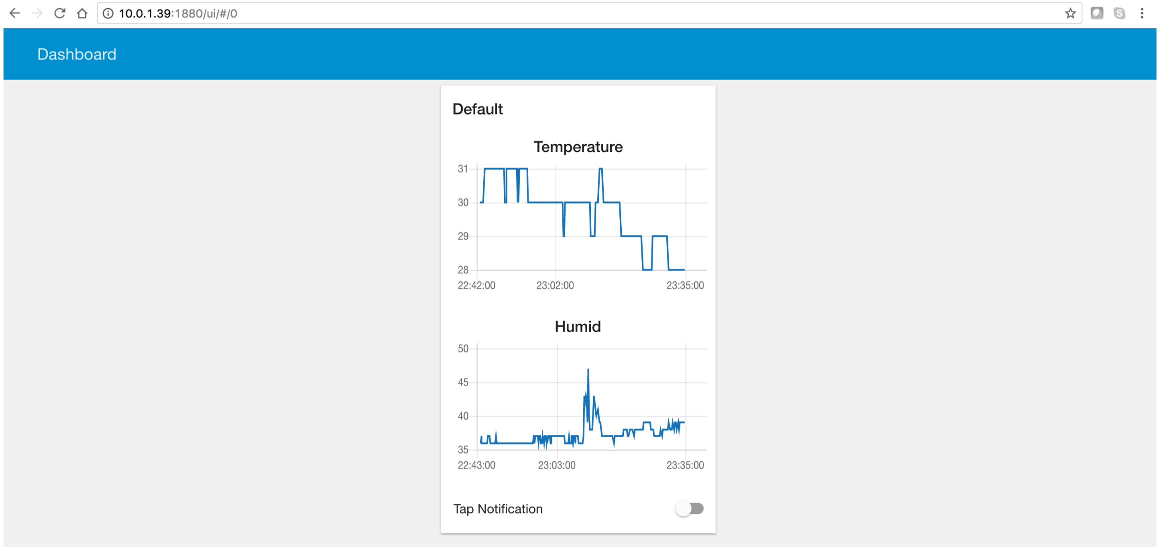
```
[root@localhost ~]# npm install –g node-red-dashboard
```

And here is my Node-RED flow to read temperature, humidity sensor data every 15 seconds, and the flow captures tap notifications at the same time.



You can find the exported Node-RED flow here.

Launch your browser, and go to http:<your_ARTIK_target_ip_address>:1880/ui, you can see the dashboard.



## Create HTTP tunnel

Node-RED is a local instance running on my Kitra520. To make its URL available externally, we will use *ngrok* to create an http tunnel for our local Node-RED instance.  Ngrok exposes your localhost to the web.

```
[root@localhost ~]# dnf install npm bluez-lib-devel
[root@localhost ~]# npm install uri-beacon ngrok
```

Create phyweb.js as below. In this script, we use ngrok to expose http://localhost:1880/ui.

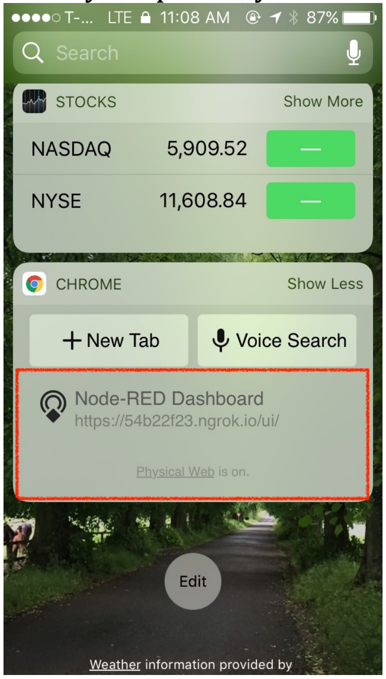
```
var url = require('url');
var beacon = require('uri-beacon');
var ngrok = require('ngrok');

ngrok.connect(1880, function(err, url) {
  beacon.advertise(url+'/ui');
});
```

Launch *phyweb.js* from Kitra command line.

```
[root@localhost ~]# node phyweb.js
```

From your phone, you should receive a notification like below:



●●●●○ T-... LTE 11:08 AM 87%

STOCKS Show More

NASDAQ 5,909.52

NYSE 11,608.84

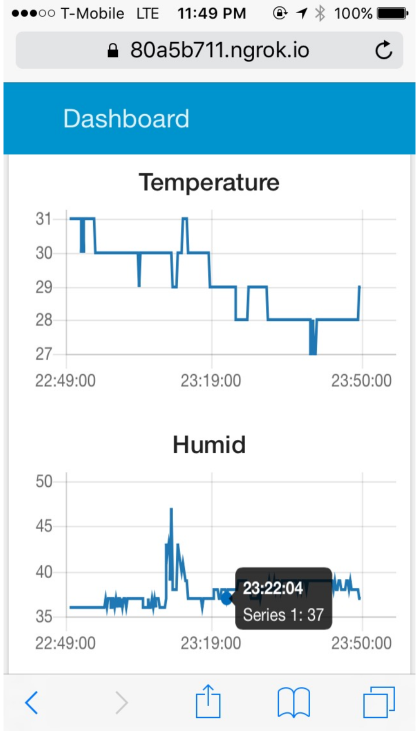
CHROME Show Less

Node-RED Dashboard
https://54b22f23.ngrok.io/ui/

Physical Web is on.

Weather information provided by

Clicking on the link from PhyWeb app, it will launch the sensor Dashboard page.


●●●○○ T-Mobile LTE 11:49 PM 100%
80a5b711.ngrok.io
Dashboard
Temperature
31
30
29
28
27
22:49:00
23:19:00
23:50:00
Humid
50
45
40
35
23:22:04
Series 1: 37
22:49:00
23:19:00
23:50:00