RUCKUS AND KONTAKT.IO

DEPLOYMENT GUIDE

APRIL 2019

TABLE OF CONTENTS

Contents

Intended Audience

This document outlines the steps in the deployment and configuration of the Kontakt.io connected BLE tags using the Ruckus Networks wireless infrastructure. The document has been written for use by systems engineers.

This document is written for and intended for use by technical engineers with some background in Wi-Fi design and 802.11/wireless engineering principles. A background in the Ruckus wireless infrastructure as well as the connected Kontakt.io tags and bracelets is recommended.

For more information on how to configure Ruckus products, please refer to the appropriate Ruckus user guide available on the Ruckus support site, http://support.ruckuswireless.com.

Introduction

Internet of Things (IoT) deployments are often complex and involve products and services at various layers: devices/endpoints, network infrastructure, middleware and platform/services. Market complexity at each layer makes deployments risky and requires extensive integration services. Network silos at each layer necessitate the duplication of equipment and cabling, thereby making deployments expensive. Furthermore, such patched together solutions give rise to security vulnerabilities.

Ruckus + Kontakt.io offer an integrated, unified solution for connected, asset, patient, and employee tracking the solution leverages the Ruckus Wireless Wi-Fi infrastructure and their leading positions in the IoT marketplace. This integrated approach simplifies device/endpoint onboarding, establishes uniform security protocols and unifies device/endpoint management and policy setting.

This document provides a step-by-step guide to setting up and configuring a connected lock solution using locks provided by Kontakt.io and wireless infrastructure provided by Ruckus.

OVERVIEW

MAJOR SUBSYSTEMS

The figure below depicts the major system components. Their functionality is summarized in the following table.

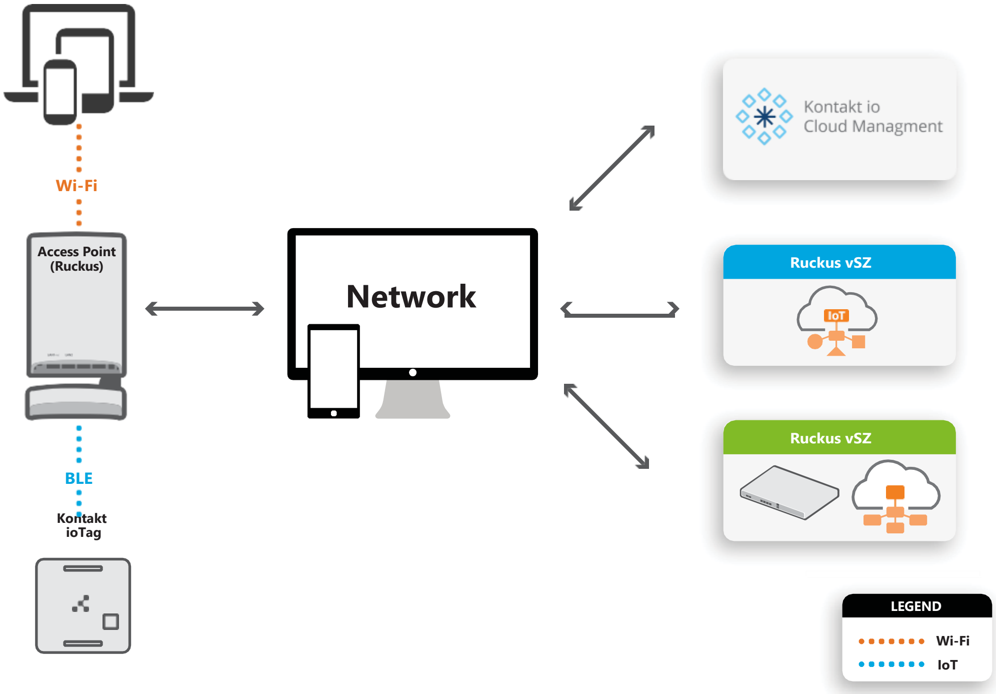


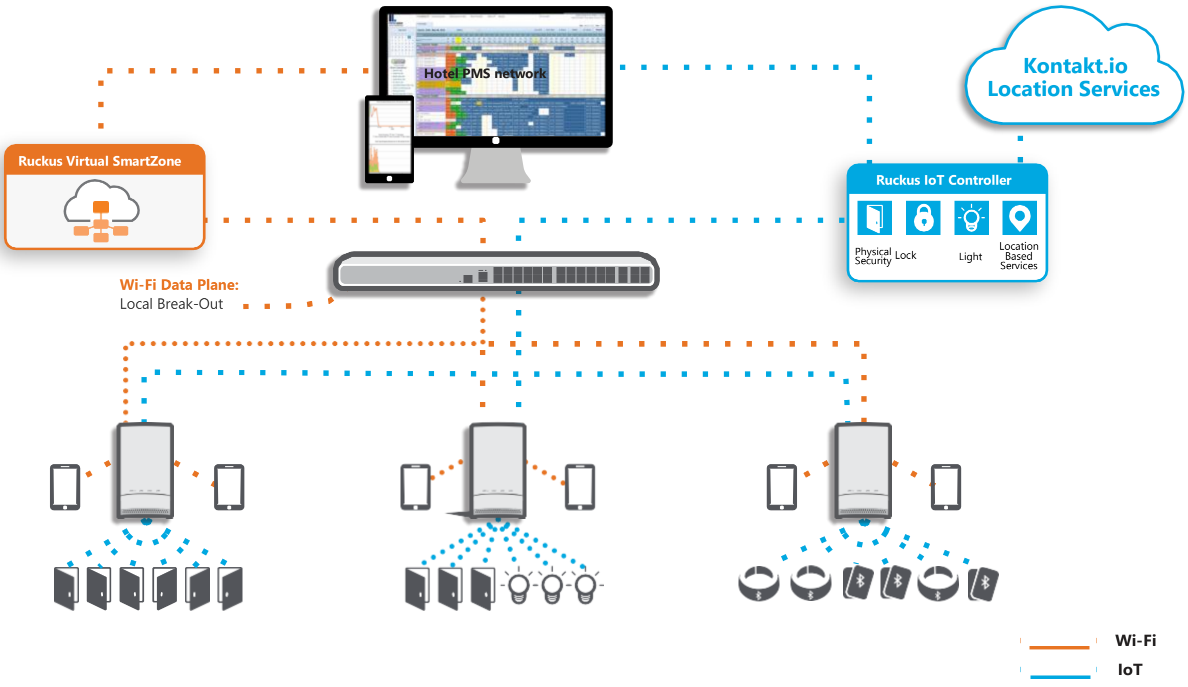
Figure 1. Ruckus and Kontakt.io System Components

Subsystem / Component	Description
SmartZone Controller	The Ruckus SmartZone is a WLAN controller and is available either as an appliance (SZ-100) or a virtual Controller (vSZ). It onboards and manages the Ruckus Access Points including downloading the appropriate firmware to the Ruckus APs.
Access Points	The Ruckus family of Wi-Fi Access Points includes a full suite of APs available in a variety of form factors, antenna options, RF streams, price points, etc. Some models can be equipped with the I100 IoT module with ZigBee capabilities, making them both Wi-Fi and ZigBee capable. These include: H510, R510, T310, E510, R610, T610, R710 and R720.
IoT Controller	The Ruckus IoT Controller is a management platform for provisioning and onboarding Ruckus IoT APs, Smart Door locks while also providing connectivity to smart lock management platforms such as the Assa Abloy Vision- line platform. The Ruckus IoT controller is available as a virtual machine.
Card Tag CT18-3	The Kontakt.io card tags are BLE capable and communicate with the Kontakt.io cloud management system.
Kontakt IO Location Server	The Kontakt.io location server is a web-based services and location management platform for controlling and setting policy for the Kontakt.io devices.
Wi-Fi devices	These are other Wi-Fi devices such as smartphones, tablets, etc. that utilize the Ruckus Wireless infrastructure.
Network	The wired network that provides connectivity between the various components above.

DATA PATHS

The end-to-end data path is between the Kontak.io tag and the Kontakt.io Cloud Dashboard. A smart door lock communicates with the Ruckus AP over BLE. The Ruckus AP then forwards the data to the Ruckus IoT Controller via the wired network infrastructure. The Ruckus IoT Controller then forwards this to the Vision Line server, thus completing the data path.

There exists a management data path between the Ruckus SmartZone Controller and the Ruckus Access Point. However, this is used for onboarding and management of the APs and is not a part of the end-to-end data path.



VERSIONS

The following versions/options are required for the integrated setup.

Component	Version
Ruckus SmartZone	3.6.1.2.10051 or higher. Note that the version must be IoT capable. Higher versions (such as 5.0) may not be IoT capable. 5.1.1 is IoT compatible.
Ruckus IoT Controller	1.0 or higher
	2.0
Kontakt.io Cloud Manager	Online Access to Portal.
Kontakt.io BLE cards	Card Tag • CT18-3
	Bracelet Tag • BT18-3

DESIGN & INSTALLATION

This section addresses the design and installation considerations pertaining to the various system components.

WI-FI INFRASTRUCTURE, COVERAGE & AP PLACEMENT

The design and installation of the Wi-Fi infrastructure (number and placement of APs, switches, power, etc.) is beyond the scope of this document. There are several Design and Best Practice Guides available from Ruckus on how to deploy Wi-Fi in a CCRC environment.

BLE COVERAGE

The ZigBee coverage requirement is -65 dBm. In most cases, not all APs need to be equipped with the I100 module to provide enough BLE coverage to the all the door locks in a hospitality environment. In a hotel, APs are deployed to provide in-room as well as in-corridor coverage. This typically results in placing APs in the corridors as well as in the room. Equipping the corridor APs with I100 should provide enough ZigBee coverage in most cases.

All hospitality sites are different from a RF Coverage perspective and it is recommended that a site survey be performed with an RF sniffer to ensure that enough BLE coverage as available at the door locks.

RUCKUS IOT READY INFRASTRUCTURE

As described in the Ruckus IoT Suite Getting Started Guide[1], this consists of installing the following:

1. Ruckus SmartZone Controller
2. Ruckus Access Points
3. Ruckus I100 IoT Module on the Ruckus AP
4. Ruckus IoT Controller

RUCKUS SMARTZONE CONTROLLER

The Ruckus SmartZone controller is available as either an appliance (SZ-100) or a virtual controller. The virtual controller is available as an *.ova file and can be installed on the VMWare/ESXi hypervisor. Specifications regarding the minimum requirements for CPU, Memory, Disk, etc. are provided in the Getting Started Guide[2]. This guide also contains detailed installation instructions for various virtualization platforms.

Given below is a brief summary of the relevant steps for setting up the vSZ.

1. Download the relevant image (such as *.ova file) and upload to the Hypervisor
2. Configure the CPU, RAM, Disk, etc. for the image. This is determined by the number of APs, clients, etc. to be supported and is specified in the Guide mentioned above. Also, delete Network Adapter 2 and 3.
3. Launch the image and wait for it to power up on the hypervisor console
4. Perform the basic configuration:
    a. Login in with ‘admin’ for both username and password
    b. Enter ‘setup’ command and press enter
    c. For example, enter ‘1’ for Essentials and press enter
    d. Enter ‘Y’ and press enter

[1] Ruckus IOT Suite Getting Started Guide

[2] Ruckus SmartZone Getting Started Guide

e. Enter ‘1’ for IPv4 and press enter

f. Enter ‘2’ for DHCP and press enter

g. Note down the assigned IP address to use later for accessing vSZ web UI and enter ‘y’ for yes to and press enter

h. Enter the primary DNS that the IoT Controller uses (refer back to ‘Get Network Info’ command) and press enter

i. Enter secondary DNS if applicable, otherwise leave blank and press enter

j. Enter Control NAT IP if applicable, otherwise leave blank and press enter

k. Enter ‘restart network’ to restart changes made and press enter

l. Type ‘setup’ and press enter

m. Type ‘NO’ when asked setup network and press enter

n. Type ‘c” to create new cluster and press enter

o. Enter a cluster name (ie: Vingcard) and press enter

p. Enter a controller description (ie: vrIoT) and press enter

q. Type ‘y’ to confirm settings are correct and press enter

r. Enter a controller blade name and press enter

s. Press enter @ system UTC

t. Press enter @ NTP server

u. Type ‘N’ when asked to convert ZoneDirector APs and press enter

v. Enter a new admin password and press enter, then enter it again and press enter

w. Enter a CLI enable command password and press enter, then enter it again and press enter

x. Wait for setup to complete

5. Access the instance via https using its IP address and port 8443. The username is admin and the password are the one set above.

RUCKUS ACCESS POINTS (APS)

Deployment of Ruckus APs is beyond the scope of this document. Depending on the AP model number, detailed installation instructions are available from Ruckus.

After the APs have been installed, insure that they have been “discovered” by the Ruckus SmartZone controller. Once this discovery has taken place, the controller will automatically upgrade the AP firmware.

RUCKUS I100 IOT MODULE

Detailed instructions for installing the I100 IoT module on the AP are available in the I100 Setup Guide.[3]

RUCKUS IOT CONTROLLER

The Ruckus IoT Controller is a virtual controller that is available as a *.ova file that can be installed on the VMware ESXi hypervisor. Detailed installation instructions including the CPU, RAM, Disk, etc. requirements are contained in the IoT Controller Installation Guide.[4]

[3] Ruckus I100 IoT Module Quick Setup Guide

[4] Ruckus IOT Controller, Software Installation Guide

Given below is a summary of the relevant steps for installing this on the VMWare ESXi platform.

1. Download the *.ova image and upload to the ESXi server
2. Verify that the instance has 2 vCPUs, 2 GB RAM and 8GB disk
3. Power up the instance and access it via the console. Credentials are admin/admin
4. Enter 1 to get the IP address of the virtual machine
5. Access the instance via a Web browser. Both http and https are supported
6. In the initialization page, select all the services and specify the FQDN for this instance
7. Confirm the configuration information and click Start
8. The IoT Controller page is now displayed. Credentials are admin/admin

KONTAKT.IO DASHBOARD

This is a cloud-based server, after configuration, the user name and password should be noted.

KONTAKT BEACON CARDS

1. You will need 2 x CT18-3 Card Tags, 2 x BT318-3 Bracelets
2. Kontakt.io operator with access to add tags to the Kontakt.io cloud manager.
3. Navigate to Beacons on the Cloud Manager.
4. Enter the Order ID corresponding with the card ID printed on the back of the card.
5. Repeat for each Kontakt device.

CONFIGURATION

KONTAKT.IO DEVICE MANAGEMENT DASHBOARD

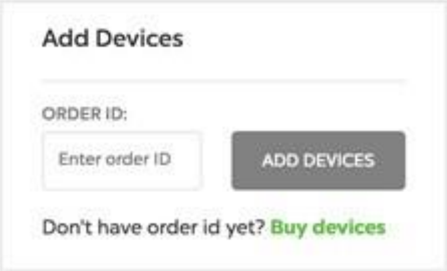
Kontakt.io Device Management Dashboard is available as cloud-based service: https://panel.kontakt.io/app/dashboard After

creating login credentials, the user name and password should be noted.

PROVISIONING KONTAKT.IO BEACONS AND TAGS

Purchased Kontakt.io beacons, bracelets and tags, are being shipped to end-user together with Sales Order ID printed on the little green form. The user must add newly received devices to his/her Kontakt.io account, by following few simple steps:

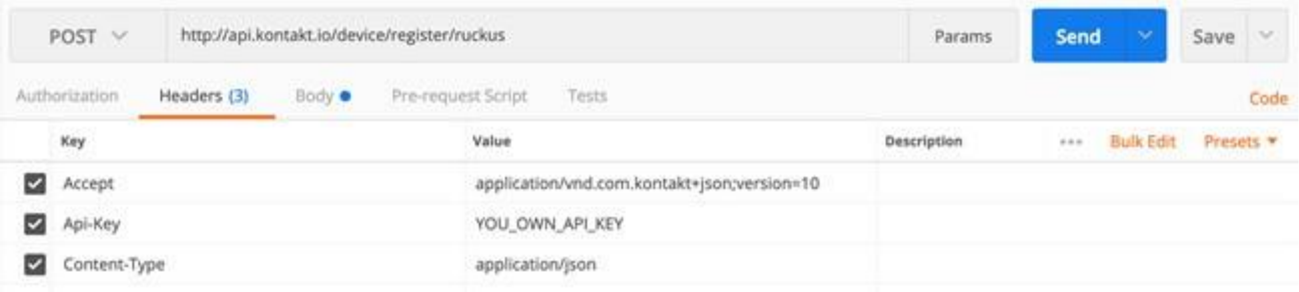
1. Login to Kontakt.IO Device Management Dashboard
2. On main page enter the Order ID received with the device in the “Add Devices” section.
3. Repeat for each separate order.



PROVISIONING KONTAKT.IO SOFT-GW-ID

Ruckus Access Points, while working as BLE GW for Kontakt.IO devices, send data streams to Kontakt.io cloud server. This is done on a third party non-Ruckus or Kontakt.io API posting software. In order to uniquely identify the Access Point as origin of data steam Kontakt.IO introduce a concept of Soft-GW ID. These Soft-GW IDs must be provisioned using Kontakt.IO API for every Access Point. Example of Postman API call to URL: http://api.kontakt.io/device/register/ ruckus as below:

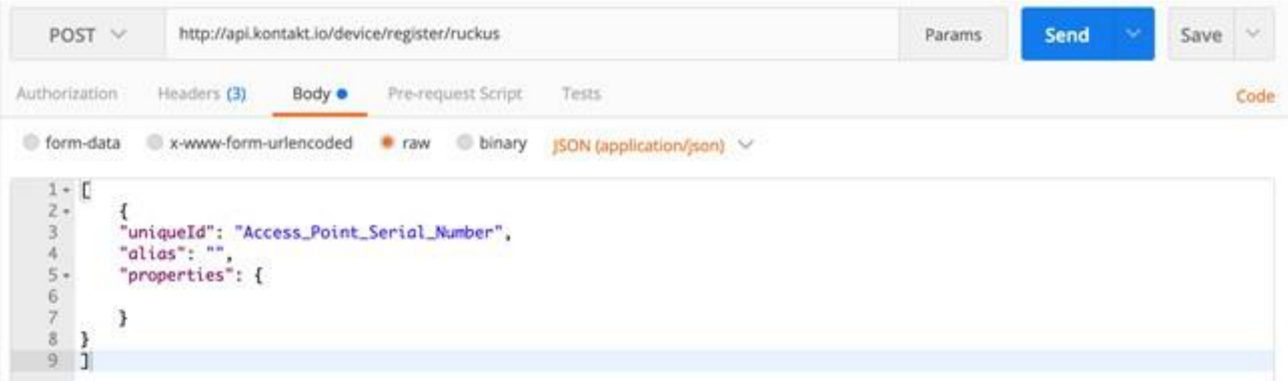
1. Header example. Please note, that API key must be copied from Kontakt.IO Device Management Dashboard



POST http://api.kontakt.io/device/register/ruckus

	Key	Value	Description
☑	Accept	application/vnd.com.kontakt+json;version=10	
☑	Api-Key	YOU_OWN_API_KEY	
☑	Content-Type	application/json	

2. Body Example. Please note, that AP name must be unique, hence the recommendation to use AP Serial Number.



```
[
    {
    "uniqueId": "Access_Point_Serial_Number",
    "alias": "",
    "properties": {

    }
}
]
```

RUCKUS ACCESS POINTS

Access Point needs to be provisioned with the IoT controller IP address. This can be done either using DHCP Option 43 or using the AP CLI.

AP Controller provisioning using DHCP

DHCP Option 43 is used for automatic IoT Controller IP provisioning. Option 43 sub-codes usage:

- Option 43, subcode 6 used for vSZ IP
- Option 43, subcode 21 used for IoT Controller IP (corresponding with iotg-mqtt-brokerip AP command)
- Option 43, subcode 22 used for specifying VLAN ID for IoT traffic (corresponding with iotg-ip-vlan AP command)

The DHCP server could be run on ICX switch or any DHCP server. Example DHCP server configuration for ICX switch is depicted below. Please note the use of option 43, sub-codes 6 for vSZ controller IP and 21 (hex 15) for the IoT controller IP.

```
ip dhcp-server pool group1
  excluded-address 172.16.101.254
  lease 1 0 0
  network 172.16.101.0 255.255.255.0
  option  3 ip 172.16.101.254
  option  6 ip 8.8.8.8
  option  43 hex 060e3137322e31362e3230302e323030150e3137322e31362e3230302e313030
  deploy
```

Other useful references are:

- Option 43 generator: https://shimi.net/services/opt43/
- ICX DHCP options vid: https://www.youtube.com/watch?v=OCE0eEQqQAY

Access Point Controller Provisioning using CLI

Follow the steps to provision IoT Controller IP using AP CLI:

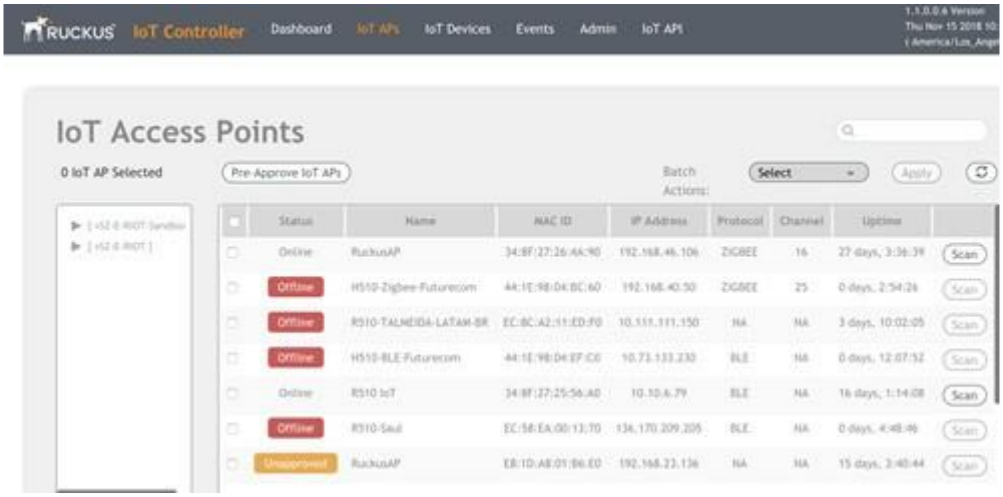
1. Log into the AP via ssh (Terminal on MAC or Putty on a PC)

2. Credentials are those listed in the SmartZone Controller (Access Points->AP->Configure->AP Configuration->AP Admin Logon)

3. Enter the following commands

```
set iotg-mqtt-brokerip <IoT Controller IP>
set iotg-mqtt-ssl 1
set iotg-mqtt-port 8883
set iotg-enable 1
```

As shown below, the AP will now appear in the IoT Controller "IOT APs" screen. Repeat the above for all Aps or if every AP in a zone will be an IoT AP an AP CLI can be created and ran on that zone in the SmartZone controller.



IoT Access Points

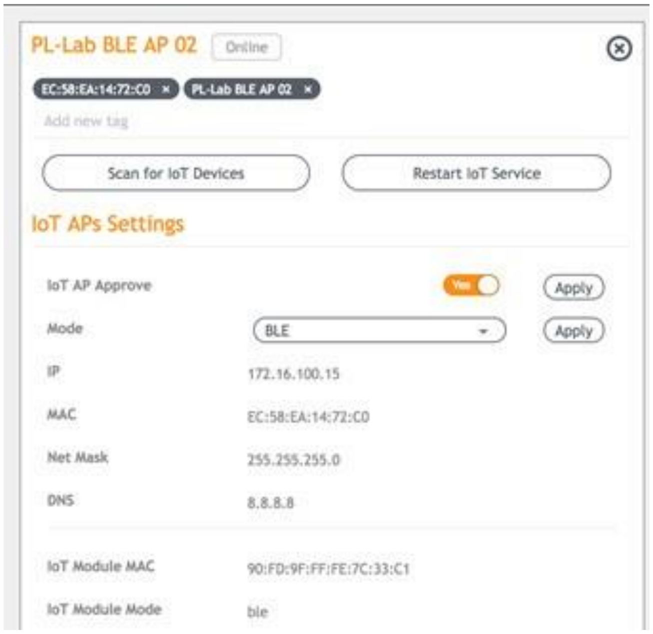
0 IoT AP Selected

	Status	Name	MAC ID	IP Address	Protocol	Channel	Uptime	
□	Online	RuckusAP	34:8F:27:26:AA:90	192.168.46.106	ZIGBEE	16	27 days, 3:36:39	Scan
□	Offline	H510-Zigbee-Futurecom	44:1E:98:D4:BC:60	192.168.40.50	ZIGBEE	25	0 days, 2:54:26	Scan
□	Offline	R510-TALMEIDA-LATAM-BR	EC:8C:A2:11:ED:F0	10.111.111.150	NA	NA	3 days, 10:02:05	Scan
□	Offline	H510-BLE-Futurecom	44:1E:98:D4:EF:C0	10.73.133.230	BLE	NA	0 days, 12:07:52	Scan
□	Online	R510 IoT	34:8F:27:25:56:A0	10.10.6.79	BLE	NA	16 days, 1:14:08	Scan
□	Offline	R510-Saul	EC:58:EA:00:13:70	136.170.209.205	BLE	NA	0 days, 4:48:46	Scan
□	Unapproved	RuckusAP	E8:1D:A8:01:B6:E0	192.168.23.136	NA	NA	15 days, 3:40:44	Scan

AP Mode Setting

Access Point must be configured to use BLE mode in order to pass traffic from BLE beacons and tags. Before changing the mode, the AP must be first approved on the controller. Please browse to "IoT Access Points" menu, select the AP, select Approve and confirm.

Consecutively set the I100 module mode of operation to BLE, by browsing to AP settings (double click on AP name), selecting the Mode: **BLE** and confirming the change.



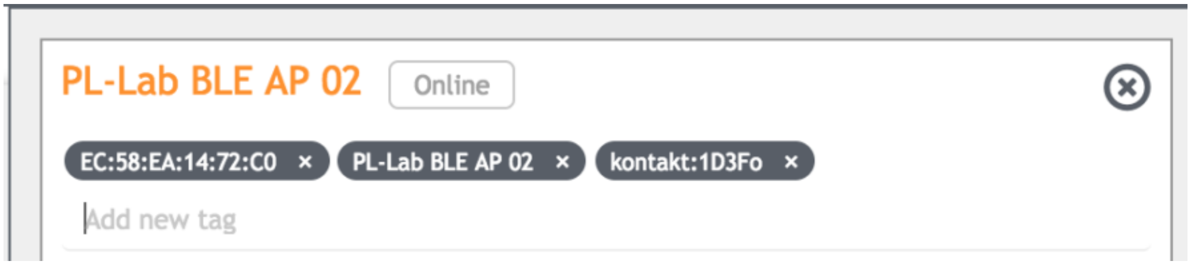
PL-Lab BLE AP 02

IoT APs Settings

IoT AP Approve	Yes
Mode	BLE
IP	172.16.100.15
MAC	EC:58:EA:14:72:C0
Net Mask	255.255.255.0
DNS	8.8.8.8
IoT Module MAC	90:FD:9F:FF:FE:7C:33:C1
IoT Module Mode	ble

AP Tagging with Soft-GW-ID

Access Point must be configured to use pre-provisioned Soft-GW-ID. Please browse to "IoT Access Points" menu, select the AP, add the TAG using given format: "**kontakt**:***your-gw-id***"



PL-Lab BLE AP 02 Online

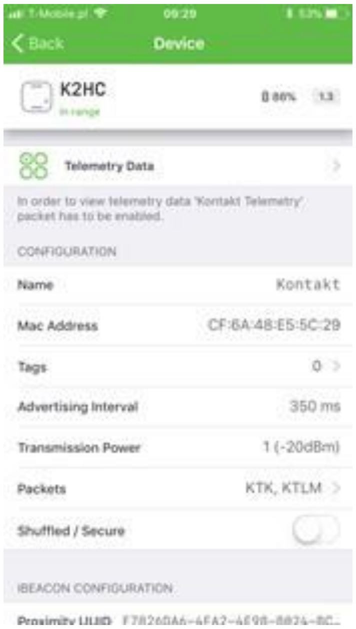
EC:58:EA:14:72:C0 ×   PL-Lab BLE AP 02 ×   kontakt:1D3Fo ×

Add new tag

SETUP KONTAKT.IO DEVICES TO SEND TELEMETRY DATA

Kontakt.io BLE devices support multiple GAP profiles: iBeacon, Eddystone and finally Kontakt Telemetry. This frame format is configurable by following below steps:

1. Download Kontakt.IO Mobile App
2. Login to the app using your Kontakt.io account credentials
3. Select every beacon, browse to *Packets* Menu and enable *Kontakt Telemetry* frames



< Back   Device

K2HC
In range

Telemetry Data

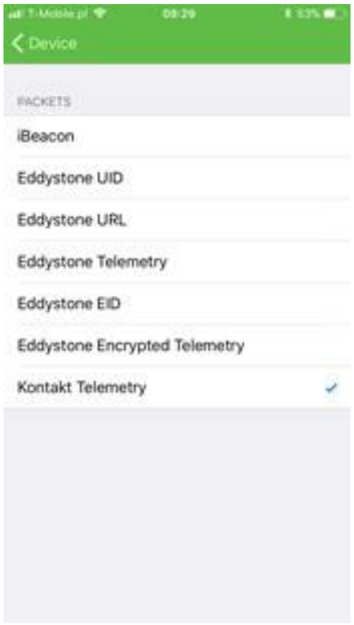
In order to view telemetry data 'Kontakt Telemetry' packet has to be enabled.

CONFIGURATION

Name: Kontakt
Mac Address: CF:6A:48:E5:5C:29
Tags: 0
Advertising Interval: 350 ms
Transmission Power: 1 (-20dBm)
Packets: KTK, KTLM
Shuffled / Secure

IBEACON CONFIGURATION

Proximity UUID: F7826DA6-4FA2-4E98-8024-BC…



< Device

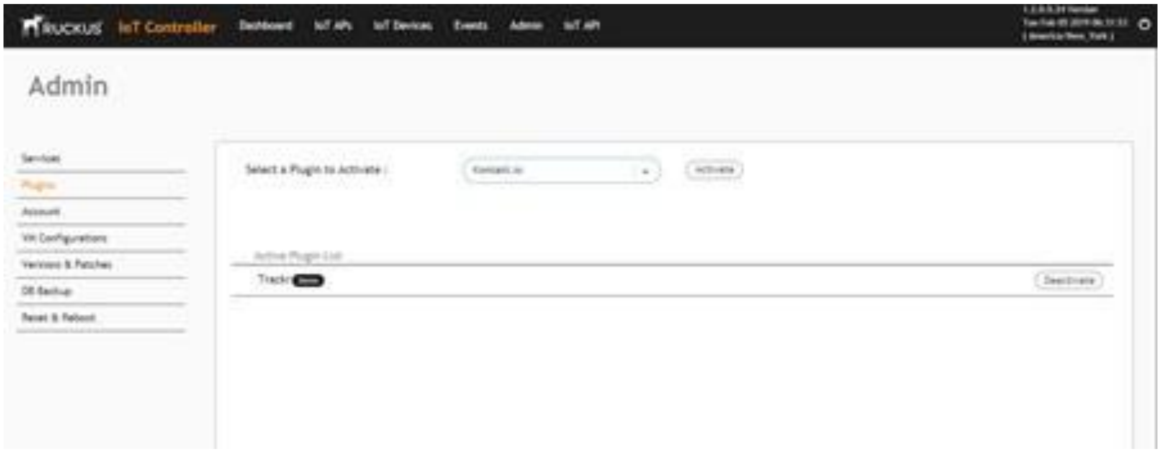
PACKETS

iBeacon
Eddystone UID
Eddystone URL
Eddystone Telemetry
Eddystone EID
Eddystone Encrypted Telemetry
Kontakt Telemetry ✓

RUCKUS IOT CONTROLLER

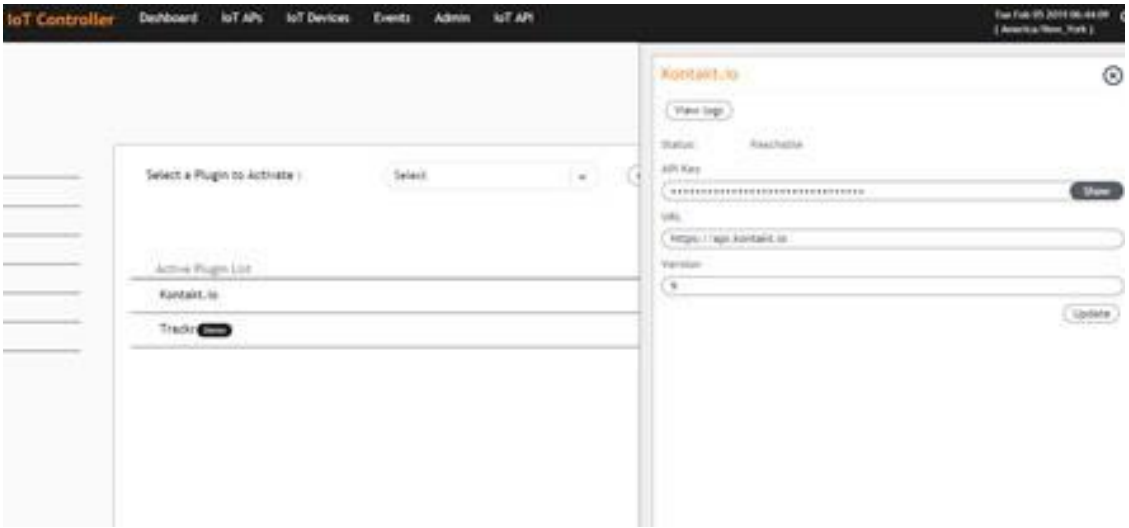
Kontakt.io Plugin

As shown below, the Kontakt.io plugin should be activated in the IoT Controller. Plugin configuration is available from Admin / Plugins menu of the IoT controller WebUI.



Select a Plugin to Activate : Kontakt.io

In this step, select **Kontakt.IO** and press activate. You will be presented with new pop-up window. Enter below settings:



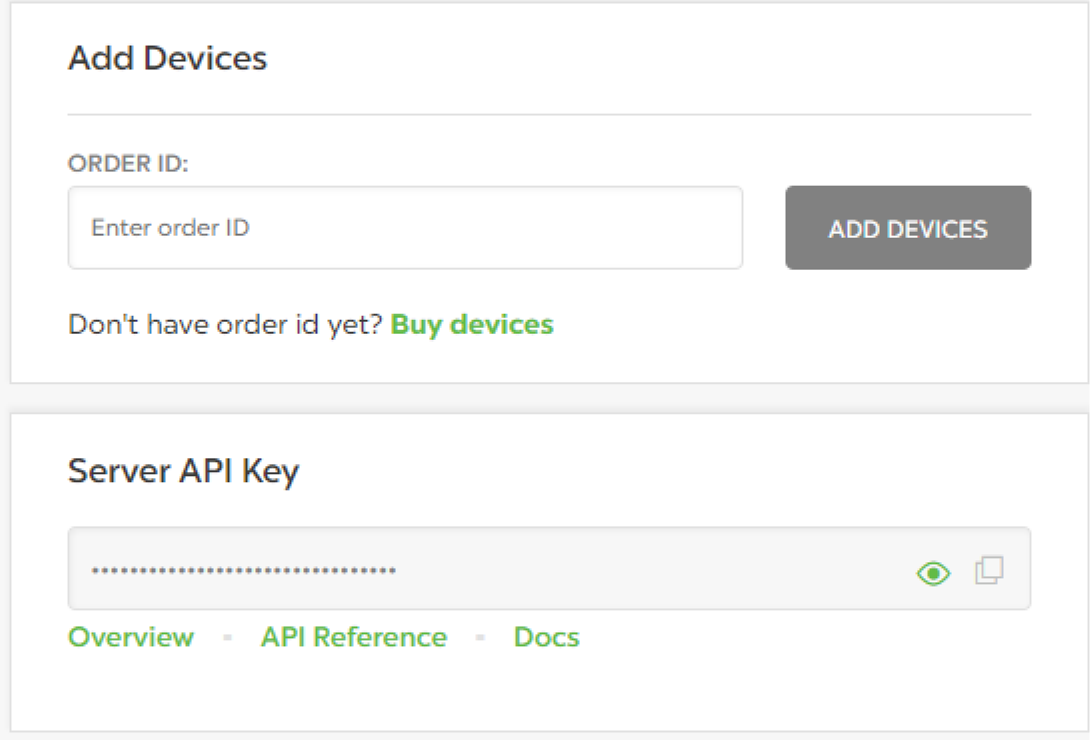
Status: Reachable

API Key: ********************************

URL: https://api.kontakt.io

Version: 9

The API key must be copied from Kontakt.IO Device Management Dashboard.



Add Devices

ORDER ID: Enter order ID

Don't have order id yet? Buy devices

Server API Key

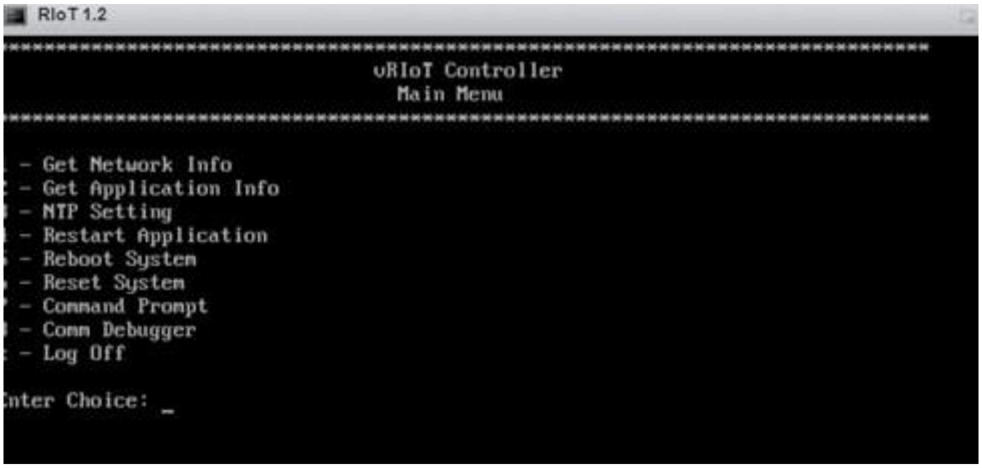
****

VERIFICATION & TROUBLESHOOTING

KONTAKT.IO BLE TRAFFIC DEBUG

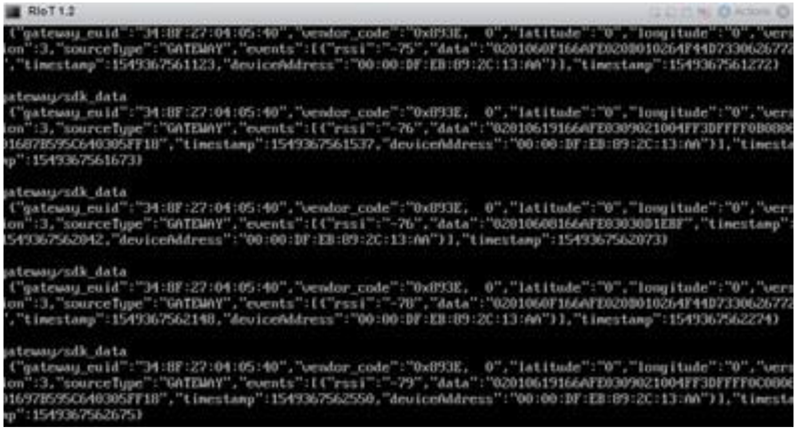
Verify BLE traffic using the following steps.

1. Log into the Ruckus IOT server via CLI and choose option 8 or Comm Debugger.



```
RIoT 1.2
********************************************************************************
                           vRIoT Controller
                              Main Menu
********************************************************************************

 - Get Network Info
 - Get Application Info
 - NTP Setting
 - Restart Application
 - Reboot System
 - Reset System
 - Command Prompt
 - Comm Debugger
 - Log Off

nter Choice: _
```

2. Enter through the next 2 prompts until you see data logs passing.



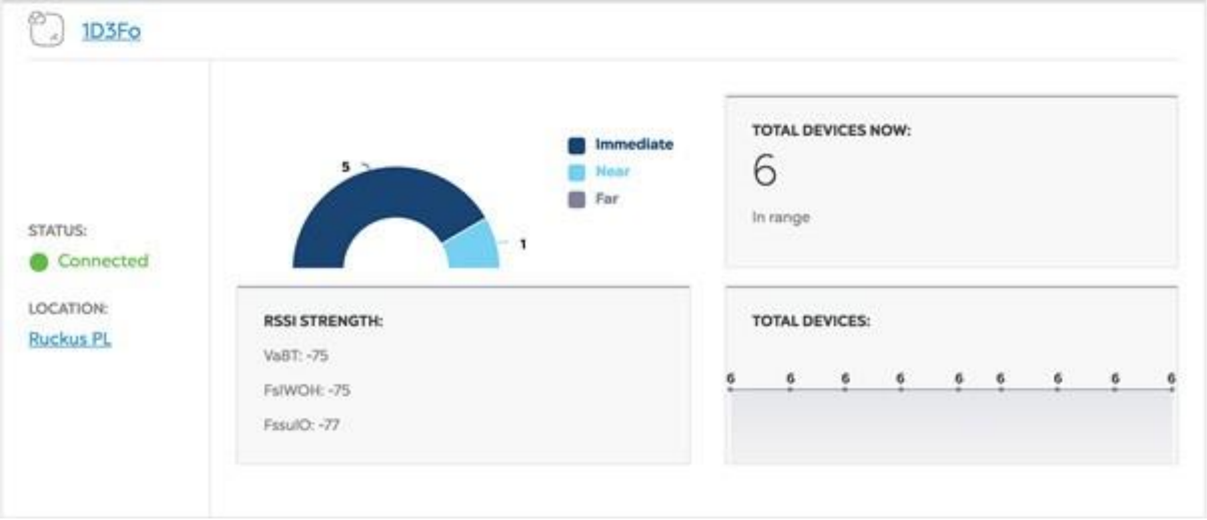
```
RIoT 1.2
{"gateway_euid":"34:8F:27:04:05:40","vendor_code":"0x893E,  0","latitude":"0","longitude":"0","vers
ion":3,"sourceType":"GATEWAY","events":[{"rssi":"-75","data":"0201060F166AFE0208010264F44D7330626772
","timestamp":1549367561123,"deviceAddress":"00:00:DF:EB:89:2C:13:AA"}],"timestamp":1549367561272}

ateway/sdk_data
{"gateway_euid":"34:8F:27:04:05:40","vendor_code":"0x893E,  0","latitude":"0","longitude":"0","vers
ion":3,"sourceType":"GATEWAY","events":[{"rssi":"-76","data":"02010619166AFE0309021004FF3DFFFF0B000
01687B595C640305FF18","timestamp":1549367561537,"deviceAddress":"00:00:DF:EB:89:2C:13:AA"}],"timesta
mp":1549367561673}

ateway/sdk_data
{"gateway_euid":"34:8F:27:04:05:40","vendor_code":"0x893E,  0","latitude":"0","longitude":"0","vers
ion":3,"sourceType":"GATEWAY","events":[{"rssi":"-76","data":"0201060816 6AFE0303001EBF","timestamp":
1549367562042,"deviceAddress":"00:00:DF:EB:89:2C:13:AA"}],"timestamp":1549367562073}

ateway/sdk_data
{"gateway_euid":"34:8F:27:04:05:40","vendor_code":"0x893E,  0","latitude":"0","longitude":"0","vers
ion":3,"sourceType":"GATEWAY","events":[{"rssi":"-78","data":"0201060F166AFE020801026 4F44D7330626772
","timestamp":1549367562148,"deviceAddress":"00:00:DF:EB:89:2C:13:AA"}],"timestamp":1549367562274}

ateway/sdk_data
{"gateway_euid":"34:8F:27:04:05:40","vendor_code":"0x893E,  0","latitude":"0","longitude":"0","vers
ion":3,"sourceType":"GATEWAY","events":[{"rssi":"-79","data":"02010619166AFE0309021004FF3DFFFF0C000
01697B595C640305FF18","timestamp":1549367562550,"deviceAddress":"00:00:DF:EB:89:2C:13:AA"}],"timesta
mp":1549367562675}
```

Kontakt.io **vendor code** value is 0x893E, hence seeing those frames proves the data passing.

KONTAKT.IO BLE TRAFFIC ON CLOUD SERVER

Kontakt.IO Device Management Dashboard presents for each Soft-GW ID assigned to specific BLE devices frames are being passed to the cloud server. Browse to *Gateways / Overview* Menu to observe data flow.

Note: the below diagram will show no data until an activated K.IO beacon is within range of the AP



1D3Fo

STATUS:
● Connected

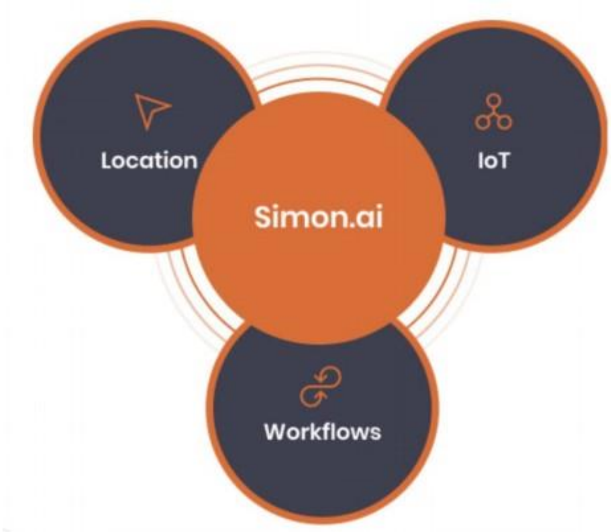
LOCATION:
Ruckus PL

TOTAL DEVICES NOW:
6
In range

RSSI STRENGTH:
VaBT: -75
FsIWOH: -75
FssuIO: -77

TOTAL DEVICES:

KONTAKT IO SIMON AI GUIDE



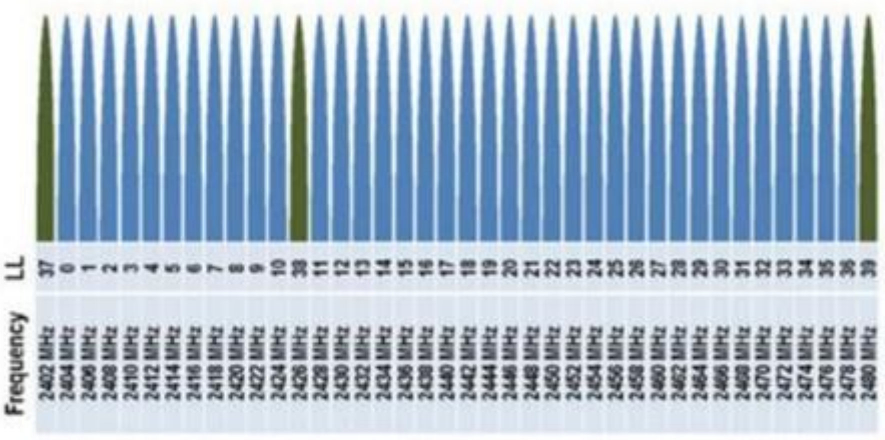
Simon is the next generation IoT platform driving the consumerization of enterprise IoT within the industrial world, healthcare and public spaces. The platform helps businesses understand, digitize and optimize physical workflows by making them transparent, predictive, actionable and data-driven.

Powered by Bluetooth Low Energy (BLE) technology, Simon fuses RTLS, IoT, and Workflow Management Systems to deliver scalable, end-to-end solutions using accurate location positioning and sensor data.

This Deployment will explain how get started with Simon.ai deployment and what to consider when planning the physical deployment for the specific use case. Basics of BLE Beacons and Tags are small Bluetooth radio transmitters that repeatedly transmits BLE signals that other devices can see.

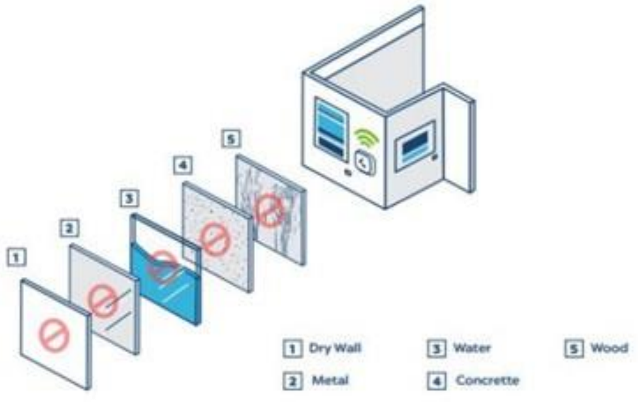
BLE stands for Bluetooth Low Energy; it’s a power-efficient version of Bluetooth originally introduced in 2010. BLE is designed to provide considerably lower power consumption compared to ‘classic’ Bluetooth, which makes it extremely useful for a wide range of new applications such as Beacons and other sensor driven implementations. BLE Tags are designed so that they don’t require persistent connection with receiver to deliver a use case, e.g. asset tracking.

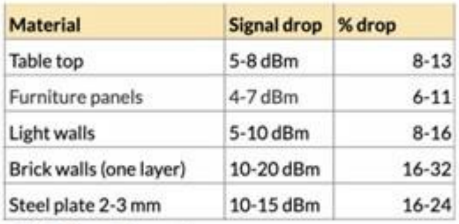
BLE operates in the 2.4GHz ISM band with only 40 channels spaced 2MHz apart. It can transmit at a rate of 1Mbit/s using GFSK modulation. Like Classic Bluetooth, it uses frequency hopping, but it uses adaptive frequency hopping and at a slower rate. It uses 3 of the 40 channels to advertise which allow for device discovery. After a device is discovered and connected the remaining 37 channels are used to transmit data.



Kontakt.io Tags kontakt.io BLE Tags are based on NRF52 chipset. They come with different shapes, sizes, battery capacity, and onboard sensors. kontakt.io has programmed tags with a functionality that allows configuring tag’s settings such as transmission power. To design a use case transmission power is relevant element in controlling how far signal of individual tag should reach. With the highest tx-power (7/7, 4 dBm) kontakt.io Tags can reach up to 70 meters. At this point it’s good to understand that this is the maximum distance signal can travel within the optimal conditions (open air, line of sight, no interference). To make it more concrete BLE tag has a small antenna and its battery-powered, which makes its physical material (e.g. walls) penetration abilities limited. BLE-signal is susceptible for multiple different kind of interferences i.e. reflections, absorption, diffraction. Therefore, the maximum signal should be treated as indicative as life life condition – such as building structure – greatly affect to real signal range. In the same fashion it’s worth reminding that human both, which is composed of 80% of water, will also block the signal and should be accounted for when designing the use case. Finally, it is good to remember that the longer distance signal travels before receiver detects it, more interference there will be on the way. To put it other way, the further the receiver detects the signal the less accurate its estimated distance from the Tag is. Therefore, while setting Tags to broadcast with the full power in the loose density receiver network may result in-optimal results.

Signal Penetration BLE signal can penetrate different materials, however signal will always be attenuated to some extent. Attenuation always depends on the material type and thickness. Finally, whether signal will be received and recognized as a BLE signal depends the quality (sensitivity) of Receiver's BLE antennae.





Material	Signal drop	% drop
Table top	5-8 dBm	8-13
Furniture panels	4-7 dBm	6-11
Light walls	5-10 dBm	8-16
Brick walls (one layer)	10-20 dBm	16-32
Steel plate 2-3 mm	10-15 dBm	16-24

Table 1. Signal attenuation

How to design proper density of Receivers

Unlike in Wi-Fi-design, you are not just optimizing the deployment density in relation to building structures but rather from the use case point of view. While the existing receiver density (e.g. IoT-enabled APs) may be enough for the perfect Wi-Fi-coverage, they may provide required density for use in question. To get to the bottom of the challenge, let's think about the following example.

Use case: asset (wheel chair) tracking in hospital Business rationale

Finding available wheelchairs takes too much time, and actual number of wheelchairs in venue is not reliable is not known, which leads to overstocking of wheelchairs and poor usage utilization of existing assets. Currently hospital staff has no visibility where wheelchairs are, and they no other way than looking for them.

Improvement to current situation:

1. Ability to know every day how many wheelchairs there are in total and per department.
2. Ability to find out where is the closest wheelchair. Floor level and wing level is enough.

Minimum requirement to achieve the improvement:

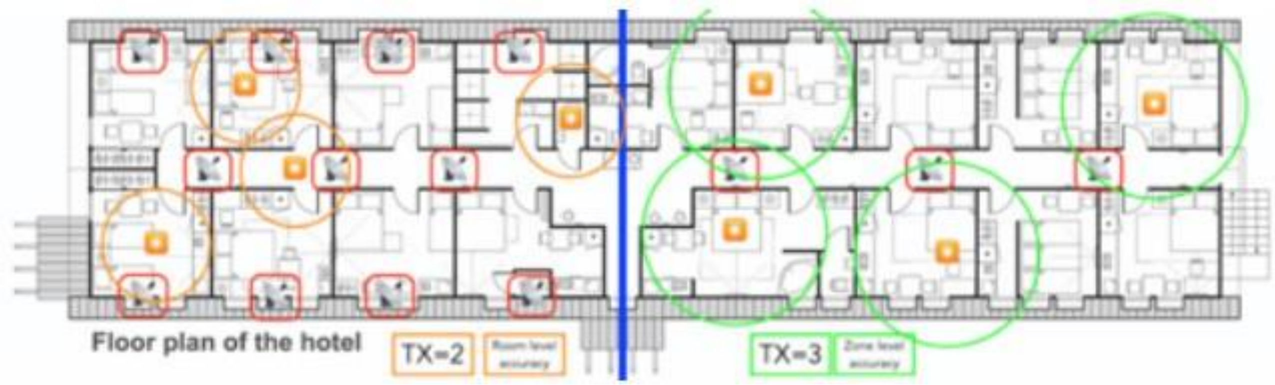
1. Receivers covering all areas where wheelchairs can move.
2. Receivers density at least covering all areas of the department. No room accuracy required.

Deployment logic:

1. Understand that each receiver will forward all any signal it receives from Tags.
2. Understand that Tags are broadcasting their chosen transmission power, that can may or may not be blocked of walls and other physical structures of the building.
3. Deploy each receiver so that considering the transmission power of Tags as well as the building structures signal of Tags will be received by the receivers.
4. Increase the density of receivers to cover areas more granular manner if desired accuracy increases from department to room level and deploy one receiver per room.

Physical deployment instructions:

1. Rule of thumb: always aim to line of sight. Visualize how Tags will move around in the venue and place receivers so that their probability of receiving the signal of the said Tags is maximized.
2. Place Receivers at least 2.5 meters high (out of hands reach) with even interval.
3. Avoid placing receiver too high in the ceiling as the estimated distance between Tag and Receiver is measured based on RSSI (received signal strength indication) and therefore higher in the ceiling can make estimate more inaccurate.



Tag's TX-power vs. use-case in the imaginary picture below describes the same use case (people tracking) with two different accuracy requirements. Respectively both wings of the pictures have different density of receiver as well tag have two different signal strengths. Respectively in both assets can be detected but with different accuracies. On the left-hand side Receivers are place in each room as well as to the corridor making receivers placed roughly every 10 meters from each other. With this density
TX-power of Tags is tuned to TX-2 thus avoiding unnecessary signal overlapping with multiple receiver. Meanwhile on the right-hand side receivers are placed only to the corridor, which requires Tags to have slightly higher transmission power TX-3 to be detected by the Receivers. Both scenarios differ also from the use case point of view. On the left-hand side, the higher density (receiver/room) of Receivers enable system to identify assets' location on the room level accuracy. Whilst of the right-hand side having receivers only in

corridor can only tell that assets are in wing but not exactly in which room. Also having Receivers placed only corridor requires Tags to transmit their signals with higher power to penetrate the walls and to be detected by the Receivers.

Example 1—asset tracking as explained above, this use case usually requires the highest accuracy and thus highest density of Receivers. The more assets there are in the same area the more accuracy is typically required to correctly pinpoint asset's location. Designing an infrastructure for asset tracking should always be evaluated from use case requirements' point of view. Below simplified example:

- One Receiver per each zone which is relevant for that asset.
- Tag's TX-power tuned to the level that enables it to be seen all relevant tracking areas assuming the Receiver infrastructure.

Example 2—duress alert Duress alerts are usually time and location sensitive. Therefore, infrastructure requirements for may sometimes be higher than those in asset tracking. For example, creating an infrastructure for a lone worker alert system for hotel maids it is often required that each room where maid is working is equipped with the Receiver. In this use case accuracy may not be the key driver, but rather a fact that when Panic Button of a Tag is being pressed it must be being picked up by Receiver. The closer the Receiver, the more likely the alert is being detected by the Receiver.

Example 3—condition monitoring in condition monitoring use cases density requirements are the most laxed. Tags are constantly transmitting (multiple times per second) their sensor readings and for most of the use cases it's enough if these readings are detected every couple of minutes or even hours. Also, often Asset's which telemetry is being monitored are in fixed locations, and therefore high Receiver density is not required for accuracy purposes. When designing a deployment for condition monitoring use cases it's the most relevant to receive the signal from the Tag within time frame required by the use case.

XY-positioning Simon positions Assets by default to the closest (the Zone of the) Receiver. While many of the use cases can be achieved with the room level accuracy some of them require higher accuracy. Therefore kontakt.io provides also optional more accurate (XY) positioning. The rule of thumb for doing the XY-positioning is that, wherever assets are wanted to be positioned, they must be seen by at least 3 Receivers. This means that when designing the Receiver density, one must evaluate how well Tags' signals can penetrate the walls and other structures, and place Receivers accordingly. Achieving proper XY-accuracy is a combination of venue characteristics, amount and locations of receiver, Tags' properties, as well as positioning algorithms and related calibrations. While the market is shouting for sub meter accuracy, the reality is that a lot of it is noise. Finally, it is good rule of thumb is that accuracy has decreasing marginal utility, meaning that in most of the use cases benefits of increased accuracy are lower than
costs lost increased accuracy. XY-positioning is always a custom project in which site survey is required, and each venue must be evaluated case by case.

Setting up Simon.ai WebApp

Simon.ai combines IoT, location (RTLS), and workflow management technologies to translate new data into business outcomes. The main user interface is an easy to use WebApp which is configured according to the goals of the use case at hand.

Prerequisites:

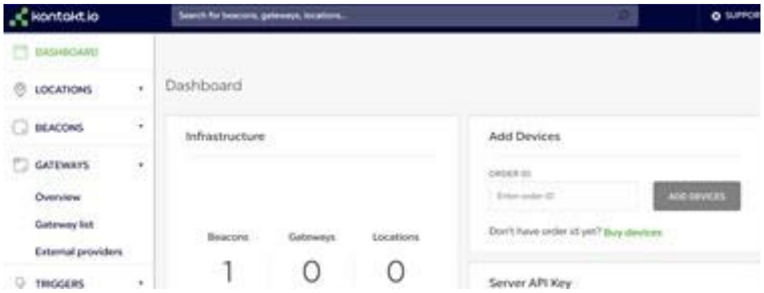
To get started with Simon you will need the following things:

- Kontakt.io Panel account (for infrastructure management)
- Kontakt.io Simon instance (using same login credentials as Panel)
- Kontakt.io Beacons/Tags on your Panel account (with TLM-frame configured and subscription activated
- Ruckus on your Panel account.
- Active Licenses for using the Simon.ai.

Setting up:

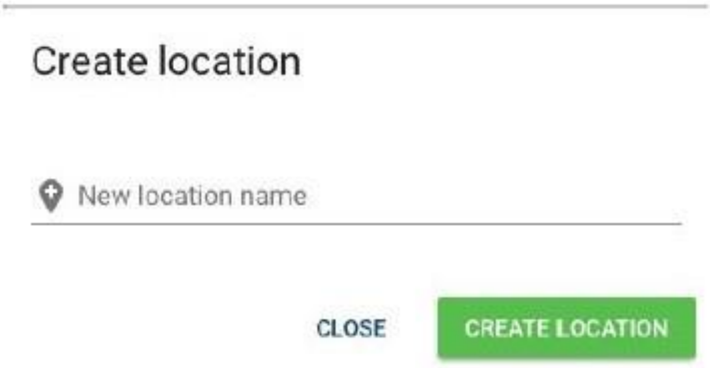
Steps to be taken on Panel WebApp (panel.kontakt.io)

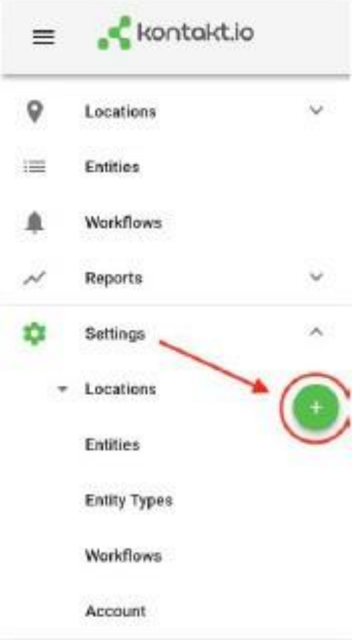
1. Create Panel Account. https://support.kontakt.io/hc/en-gb/articles/204804112-Creating-an-account-in-the-Kontakt-io-Panel
2. Add your Kontakt.io devices to your Panel account by using the Order ID https://support.kontakt.io/hc/en-gb/articles/204804482-Adding-beacons-to-your-account
3. Setup your Ruckus APs to your Panel account. https://support.kontakt.io/hc/en-gb/articles/207305070-How-to-set-up-your-Gateway



Steps to be taken on Simon WebApp (ba.kontakt.io)

Before setting up things on Simon side, all the above steps must be completed. Only once Receivers and Tags have been assigned, and licenses have been activated, we can start configuring Simon for customer. Below instructions will walk you through the process, which must be done in this order.





(1) Create Locations and Zones

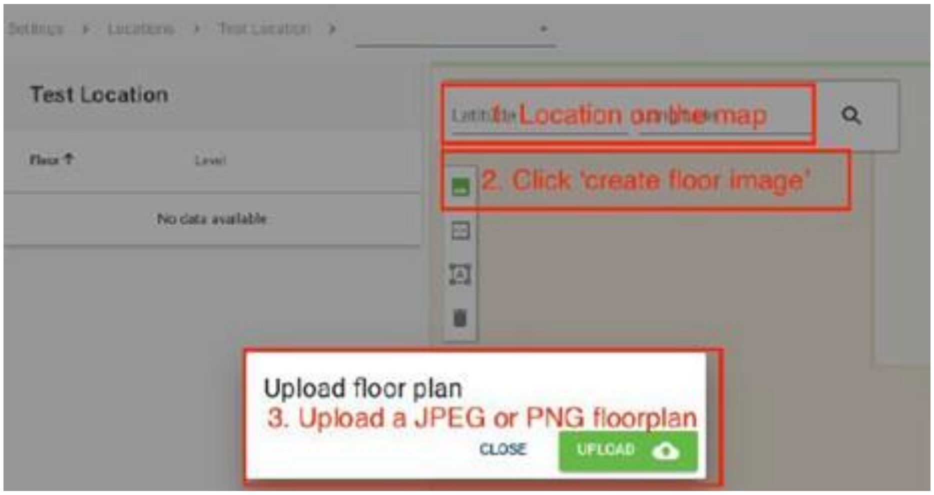
Simon.ai organizes its historical data according to the Location and Zones defined by users. Also, workflows are usually tied around specific zones.

Create a Location

- Start off by pressing the green (+) under the Settings and given Location a name.

Floorplan(s)

- Next find your deployment location from the map by scrolling or by adding Latitude and Longitude into the respective fields.
- Click small picture icon in the top left corner, this pops up a download button.
- Upload a floorplan in JPEG or PNG file format.



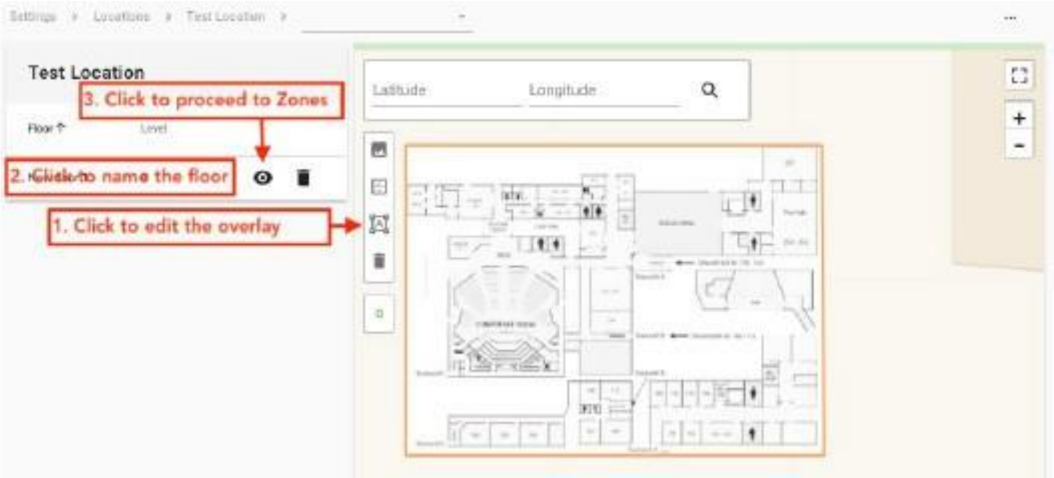
1. Location on the map

2. Click 'create floor image'

3. Upload a JPEG or PNG floorplan

Adjust Floorplan(S) and name Floor(s)

- Once floorplan is uploaded, you may edit it by clicking [A] -icon (3rd from the top). Editing happens dragging map from its corners. Once ready, stop editing by clicking [A] -icon again.
- Next you may also give to a name for the floor, by clicking the “New floor 1” text.
- If you have a multifloor-building, you can repeat steps 3-6 as many times as necessary.
- Once floorplans are uploaded, click a little “eye” -icon next to the floor name. This takes you to the respective floor for building the Zones and adding the Gateways



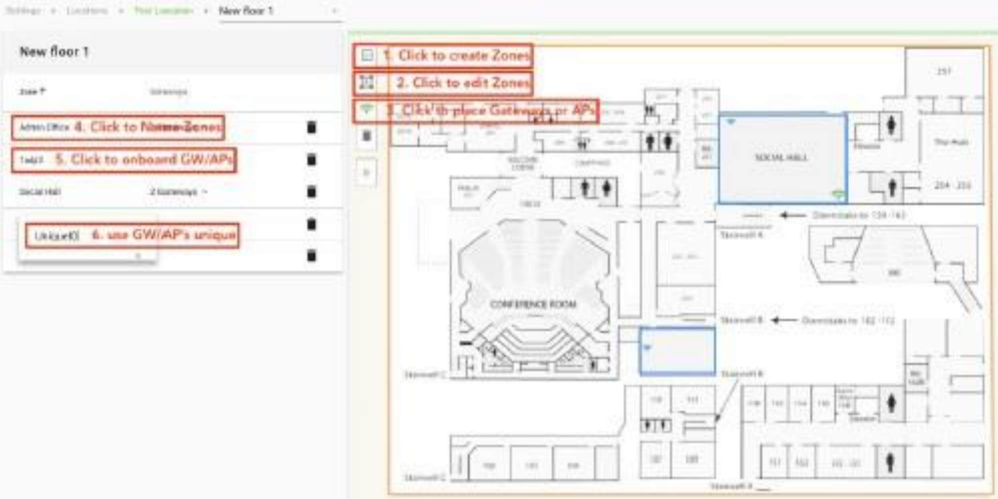
3. Click to proceed to Zones

2. Click to name the floor

1. Click to edit the overlay

Create and Name Zones

- Once the correct floor has opened, you may start to create Zones by clicking rectangular icon on the top left corner. Once ready, stop creating by clicking the same icon again.
- To edit Zones, click [A] -icon (2nd from the top). Editing happens dragging Zones from its corners. Once ready, stop creating by clicking the same icon again.



1. Click to create Zones

2. Click to edit Zones

3. Click to place Gateways or APs

4. Click to Name Zones

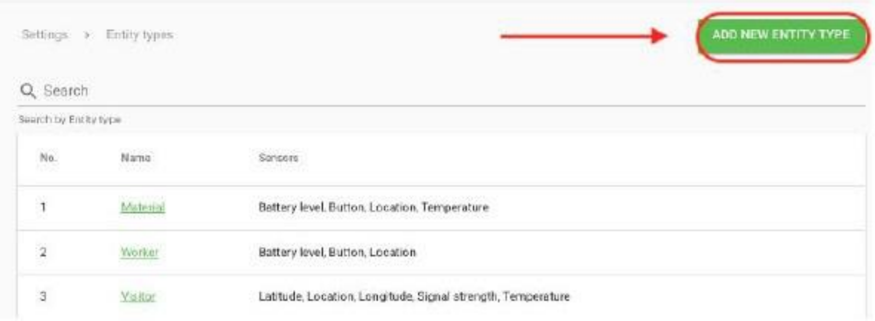
5. Click to onboard GW/APs

6. use GW/AP's unique

Pinpoint & Onboard Gateways and IoT-capable Access Points

- Next may place the Gateways & Access Points. To do this, click the “wifi” -icon (3rd from the top). Placing happens by dropping items into respective locations. Once ready, stop creating by clicking the same icon again.
- Next you may also give to a name for the Zones, by clicking the “New Zone 1” text.
- Once done its time to onboard Gateways & APs. To edit the name, you will need to click the placeholder name (number sequence) of a given Gateway. Once active you will need to us Gateway’s (or Access Points) UniqueID of that individual Gateways installed location. After renaming press Enter to save.
- Should you have multifloor building repeat steps 8-13 as many times as necessary.

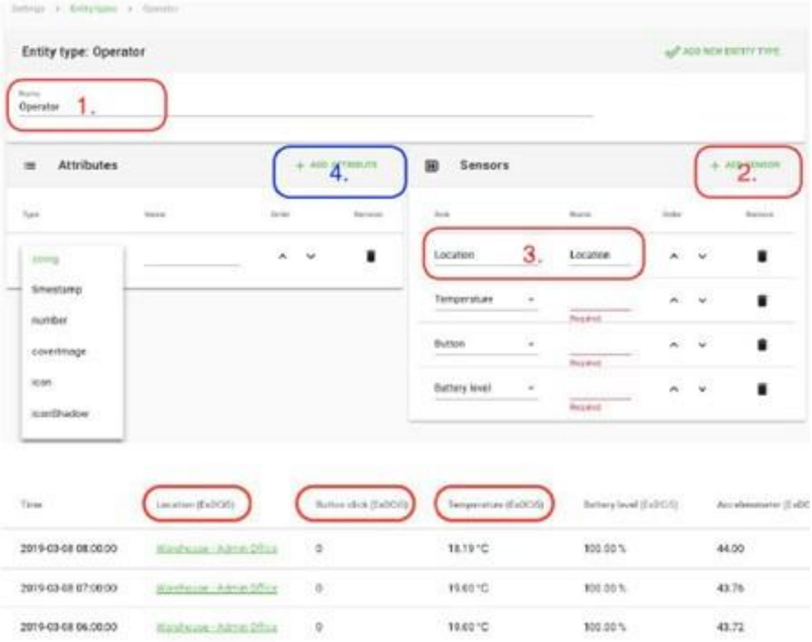
(2) Create Entity Types
All people, assets, vehicle that are being monitored in Simon are called Entities. To structure data, business rules, and workflows efficiently Entities are broken down into different groups according to the granularity needs of a given business case. These groups are called Entity Types.



No.	Name	Sensors
1	Material	Battery level, Button, Location, Temperature
2	Worker	Battery level, Button, Location
3	Visitor	Latitude, Location, Longitude, Signal strength, Temperature

Add New Entity Type

- To start creating click Entity Type, under Settings.
- Click ADD NEW ENTITY TYPE.



Time	Location ([illegible])	Button click ([illegible])	Temperature ([illegible])	Battery level ([illegible])	Accelerometer ([illegible]
2019-03-08 08:00:00	Warehouse - Admin Office	0	18.19 °C	100.00 %	44.00
2019-03-08 07:00:00	Warehouse - Admin Office	0	19.60 °C	100.00 %	43.76
2019-03-08 06:00:00	Warehouse - Admin Office	0	19.60 °C	100.00 %	43.72

Name Entity Type

- Once new window opens, you may now create a default ‘configuration’ for all this type of Entities.
- First give name to this Entity Type. E.g. Operator

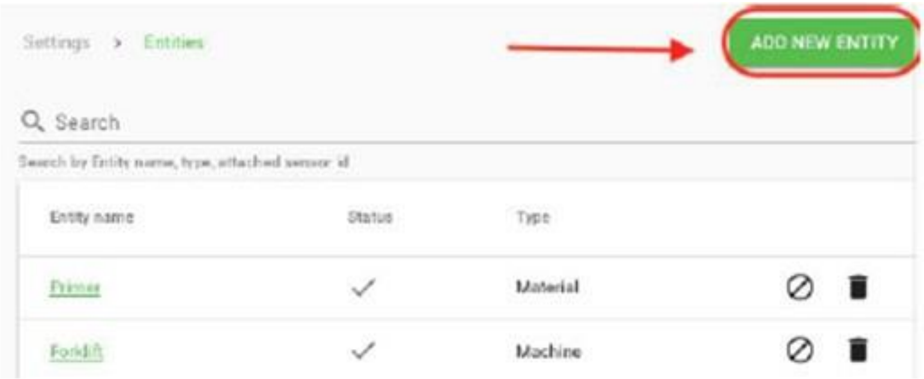
Add Default Sensors

- Click + ADD SENSOR
- Choose from drop-down list the sensor you would want to monitor.
- Add name to this sensor. This creates also the name for the column, when individual Entity’s historical data is viewed in tabular form.

Add Default Attributes

- Once you have added and named all relevant sensors, you may not click +ADD ATTRIBUTE, which brings up an additional drop-down list.
- Attributes are additional (non-mandatory) meta-data fields that are relevant for all
- Entities of this Type.
- Attribute could be e.g. product number, purchase date, custodian and other data that will be added to each Entity.
- Attribute can also be a picture of the Entity or e.g. fixed location (lat, lng) where this Entity resides.
- Attributes can also be added later for individual Entities but adding them already to Entity is as if you would create a template for it.

Once all the steps are done, you finish Entity Type creation by clicking ✓ SAVE ALL



Search by Entity name, type, attached sensor id

Entity name	Status	Type
Primer	✓	Material
Forklift	✓	Machine

(3) Create Entities

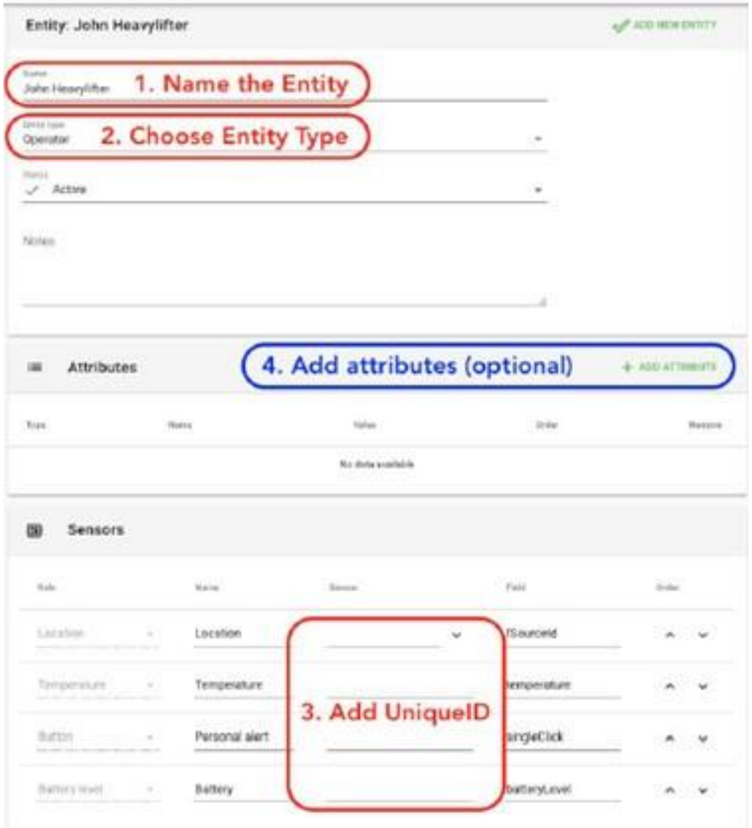
After creating all necessary Entity Types, its time create individual Entities. To start creating click Entities under the Settings. Add a

new Entity and choosing its Entity Type

- Click ADD NEW ENTITY, which opens a new window to create an Entity.
- First give Entity a name.
- Choose a correct Entity Type from a drop-down menu, which automatically brings Entity Type specific Sensor and Attribute fields visible below. Attach the UniqueID (of a Tag) to the Entity
- IMPORTANT: Add opened Sensor-field the UniqueID of that Tag/Beacon(s) you are going to physically attached to this Entity or give to this Person.
- Please, be aware that in order to have Simon to display the sensor data, the chosen Beacon/Tag must have had all relevant sensors on-board and have TLM- frame configured to be broadcasted.

Add optional Attributes for the Entity

- After you have added Unique IDs to the Sensor fields you may proceed to fill in Attributes (if applicable for this Entity).
- Once everything is ready, you need to press SAVE on upper right corner and you are done.



Entity: John Heavylifter

Name: John Heavylifter

Entity type: Operator

Status: ✓ Active

Notes

Attributes

Type	Name	Value	Order	Remove
No data available				

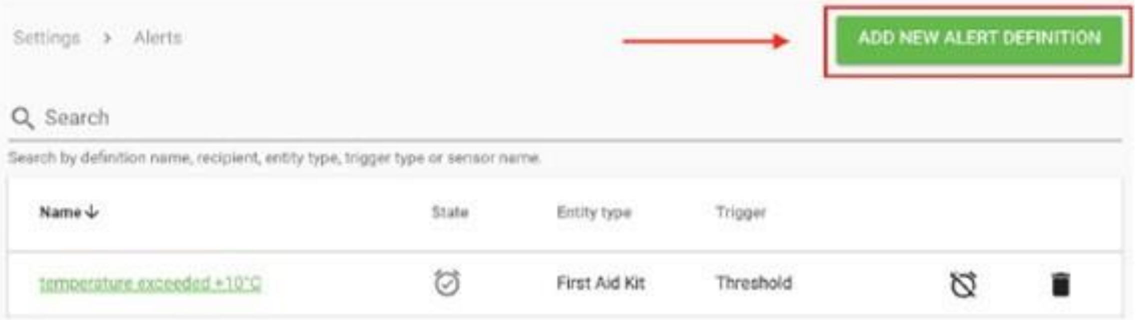
Sensors

Role	Name	Sensor	Field	Order
Location	Location		Sourceld	⌃ ⌄
Temperature	Temperature		temperature	⌃ ⌄
Button	Personal alert		singleClick	⌃ ⌄
Battery level	Battery		batteryLevel	⌃ ⌄

(4) Create Workflows

Workflows are predefined policies, that combines location or sensor data with a rule which include conditions for resulting event and action. The simplest example is monitoring fridge temperature with Tag, and setting threshold to send an alert & email, if this threshold is breached.

Workflows may use any sensor Tags and Beacons have, and they can trigger an alert in the Simon itself, as an SMS (paid extra), email, or some other predefined actions. After creating all necessary Entities, its time create individual Workflows.



Settings > Alerts

ADD NEW ALERT DEFINITION

Search

Search by definition name, recipient, entity type, trigger type or sensor name.

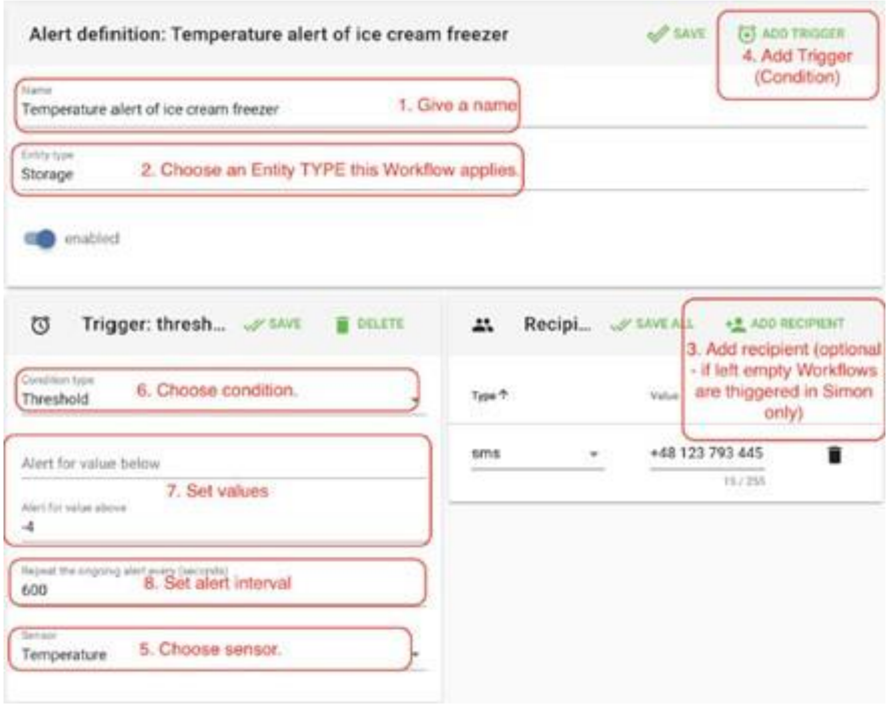
Name ↓	State	Entity type	Trigger
temperature exceeded +10°C	⏰	First Aid Kit	Threshold

Add and Name a new Workflow definition

- To start, click Workflows, under the Settings.
- Then, click "ADD NEW ALERT DEFINITION"

Name the Workflow and Choose which Entity Type it applies to

- First give a name for a Workflow. Keep in mind that this name will be included into alert when triggered.
- Choose the Entity Type this Workflow applies to.
- Add Recipient and Resulting Event
- Add a Recipient for the Workflow (optional). If not chosen Workflow will be triggered in the Simon Webapp only.
- Add conditions triggering the Workflow
- Next press "ADD TRIGGER" which unveils settings for configuring the Workflow.
    1. Choose Sensor, to which will be based on.
    2. Choose a Condition
    3. Set Values for triggering the Condition.
    4. Set the Interval to how often Workflow will be re-triggered when conditions are still met.



Alert definition: Temperature alert of ice cream freezer

SAVE

ADD TRIGGER

4. Add Trigger (Condition)

Name: Temperature alert of ice cream freezer

1. Give a name

Entity type: Storage

2. Choose an Entity TYPE this Workflow applies.

enabled

Trigger: thresh... SAVE DELETE

Condition type: Threshold

6. Choose condition.

Alert for value below

7. Set values

Alert for value above: -4

Repeat the ongoing alert every (seconds): 600

8. Set alert interval

Sensor: Temperature

5. Choose sensor.

Recipi... SAVE ALL ADD RECIPIENT

3. Add recipient (optional - if left empty Workflows are thiggered in Simon only)

Type ↑	Value
sms	+48 123 793 445

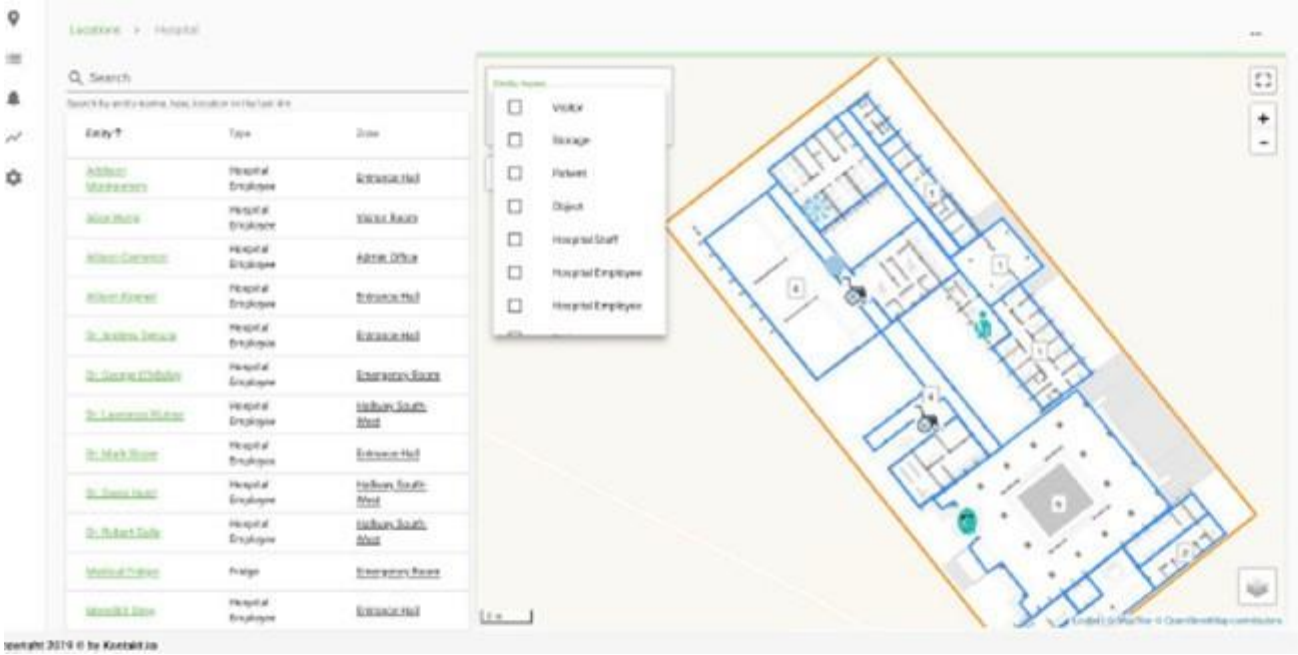
15 / 255

How to use Simon?

After setting everything up, you are good to go from Simon's point of view. At this point, you should have all the Gateways and Access Points installed inside the building and onboarded and pinpointed in their respective location on Simon. Also, all your Tags and Beacons should now be Tagged to assets or people of the interest. If everything is ok, you should already see data being received and visualised in Simon. Platform has four main modules with one or more submodules, which all help users solve different problems from slightly different angles.

Main view: Locations | Sub-view: Floorplans

Locations is essentially a map view. In this view user may search, filter, and monitor her assets' locations in real time in different locations, building, floors, and zones. This view helps user for example quickly find required tool by simple text-based search or by using filtering showing all same type of tools in the chosen location. List on the left-hand side of map shows all the Entities corresponding filtering or the search query. Furthermore, items in the list enables user quickly to jump into detailed historical location and sensor data of that entity.



Locations > Hospital

Search

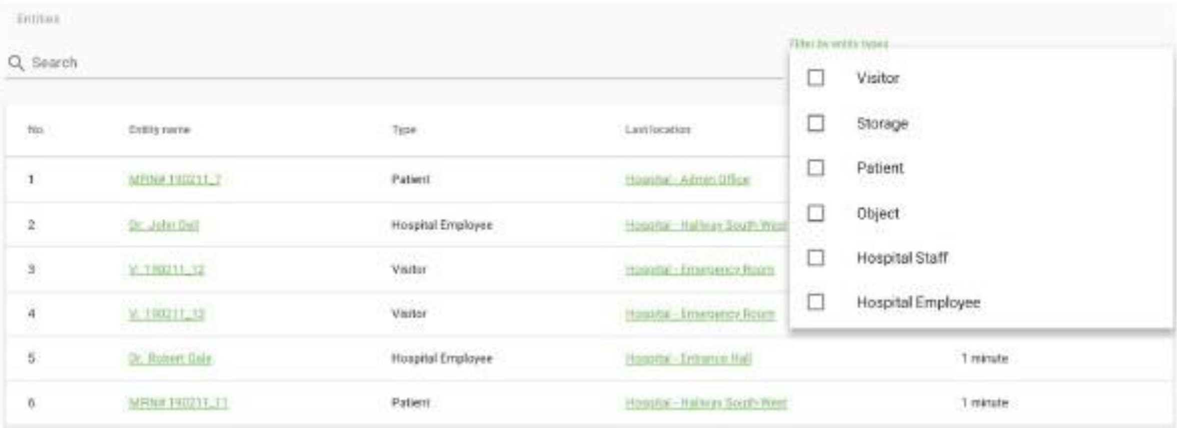
Entity ↑	Type	Zone
Addison Montgomery	Hospital Employee	Entrance Hall
[illegible]	Hospital Employee	Visitor Room
[illegible]	Hospital Employee	Admin Office
[illegible]	Hospital Employee	Entrance Hall
[illegible]	Hospital Employee	Entrance Hall
Dr. George O'Malley	Hospital Employee	Emergency Room
[illegible]	Hospital Employee	Hallway South West
Dr. Mark Sloan	Hospital Employee	Entrance Hall
[illegible]	Hospital Employee	Hallway South West
[illegible]	Hospital Employee	Hallway South West
Medical Fridge	Fridge	Entrance Hall
Meredith Grey	Hospital Employee	Entrance Hall

□ Visitor
□ Storage
□ Patient
□ Object
□ Hospital Staff
□ Hospital Employee
□ Hospital Employee

Copyright 2019 © by Kontakt.io

Main view: Entities | Sub-view: Individual Entity

In the Entity view user may view latest locations of her Entities. In this view user can also search Entities by name or location, and filter Entities by their types. From this user may also jump to back to the specific Zone in the Location view or into to the detailed view of individual Entity.



Entities

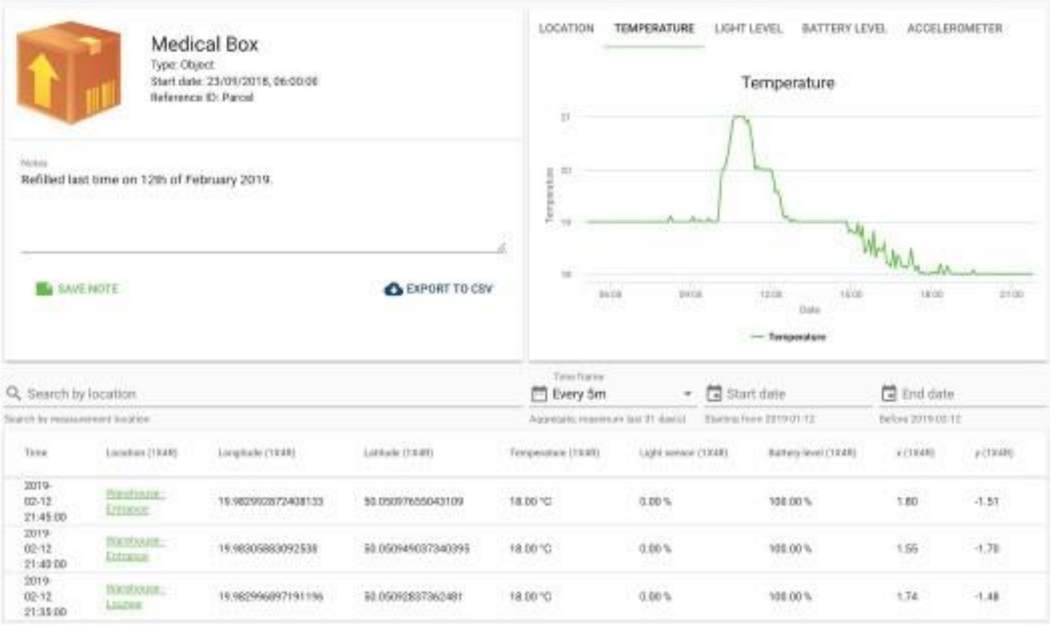
Search

Filter by entity types

□ Visitor
□ Storage
□ Patient
□ Object
□ Hospital Staff
□ Hospital Employee

No.	Entity name	Type	Last location	
1	MRN# 190211_7	Patient	Hospital - Admin Office	
2	Dr. John Doe	Hospital Employee	Hospital - Hallway South West	
3	V_190211_12	Visitor	Hospital - Emergency Room	
4	V_190211_13	Visitor	Hospital - Emergency Room	
5	Dr. Robert Dale	Hospital Employee	Hospital - Entrance Hall	1 minute
6	MRN# 190211_11	Patient	Hospital - Hallway South West	1 minute

In the Individual Entity view user can search and browse historical location and sensor data of the individual Entity. This view enables user to deep dive into granular data trail from the timeframe of interest, and finally export this data as an CSV-file for further analysis. Each column in the tabular view can be defined by the user, as well as the Entity Attributes visible in the widget on the left- hand corner.



Medical Box
Type: Object
Start date: 23/09/2018, 06:00:00
Reference ID: Parcel

Notes
Refilled last time on 12th of February 2019.

Search by location
Search by measurement location

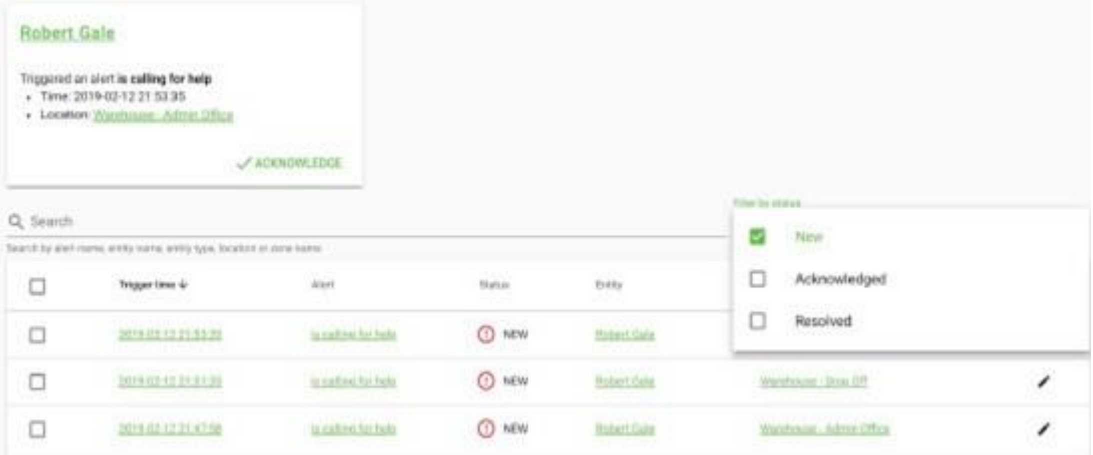
Time frame: Every 5m
Aggregate maximum last 31 day(s)

Start date
Starting from 2019-01-12

End date
Before 2019-02-12

Time	Location (1X48)	Longitude (1X48)	Latitude (1X48)	Temperature (1X48)	Light sensor (1X48)	Battery level (1X48)	x (1X48)	y (1X48)
2019-02-12 21:45:00	Warehouse - Entrance	19.982992872408133	50.050976550431 09	18.00 °C	0.00 %	100.00 %	1.80	-1.51
2019-02-12 21:40:00	Warehouse - Entrance	19.98305883092538	50.050949037340395	18.00 °C	0.00 %	100.00 %	1.56	-1.70
2019-02-12 21:35:00	Warehouse - Lounge	19.982996897191196	50.05092837362481	18.00 °C	0.00 %	100.00 %	1.74	-1.48

Main view: Workflows | Sub-view: Workflow Handling
Workflows are the combinations defined rules and resulting actions. Just like Entities Workflow section has two views; first one showing all triggered Workflows, and the second one detailed view of the individual workflow. In the main view User views, filter and search workflows by Entity and Type, handling status, and location alert workflow. On this view User may also handle the status of multiple alerts in bulk.



Robert Gale

Triggered an alert **is calling for help**

- Time: 2019-02-12 21:53:35
- Location: Warehouse - Admin Office

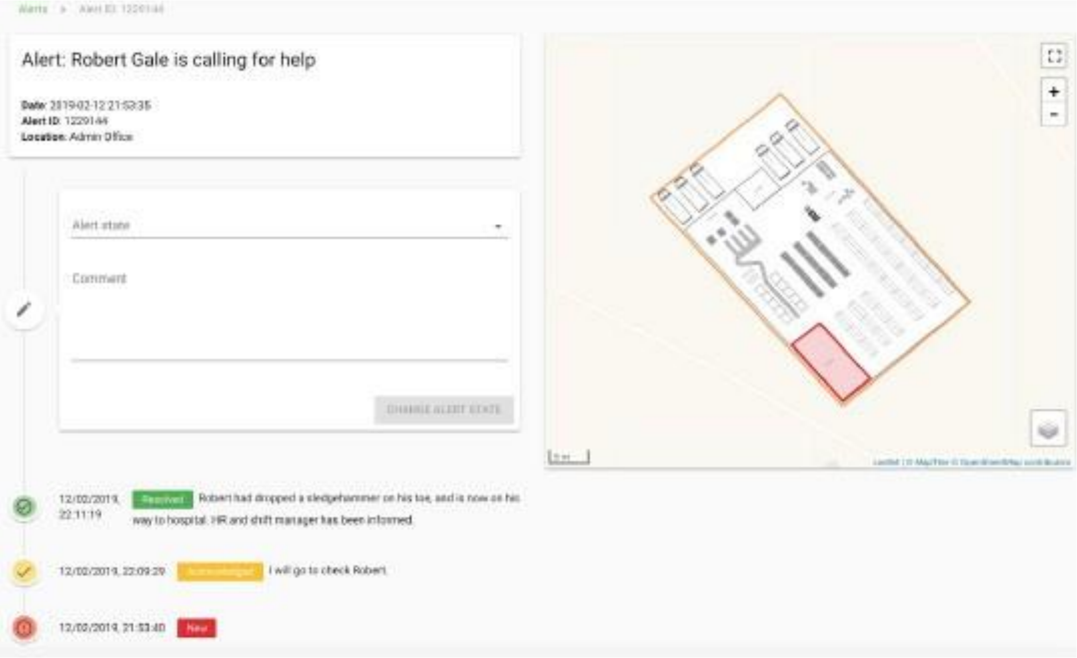
✓ ACKNOWLEDGE

Search
Search by alert name, entity name, entity type, location or zone name

Filter by status
☑ New
□ Acknowledged
□ Resolved

□	Trigger time ↓	Alert	Status	Entity		
□	2019-02-12 21:53:35	is calling for help	NEW	Robert Gale		
□	2019-02-12 21:51:35	is calling for help	NEW	Robert Gale	Warehouse - Drop Off	✎
□	2019-02-12 21:47:58	is calling for help	NEW	Robert Gale	Warehouse - Admin Office	✎

Individual Workflow view provides User with the relevant info about workflow, what triggered it, where and when it was triggered and visualizes its location on the miniature map. In addition to detailed workflow information, this view enables User to handle workflow status and to leave comments e.g. what was actions the triggered workflow has caused. This turns workflows into trackable audit trail whenever compliance is required.



Alerts > Alert ID: 1229144

Alert: Robert Gale is calling for help

Date: 2019-02-12 21:53:35
Alert ID: 1229144
Location: Admin Office

Alert state
Comment

CHANGE ALERT STATE

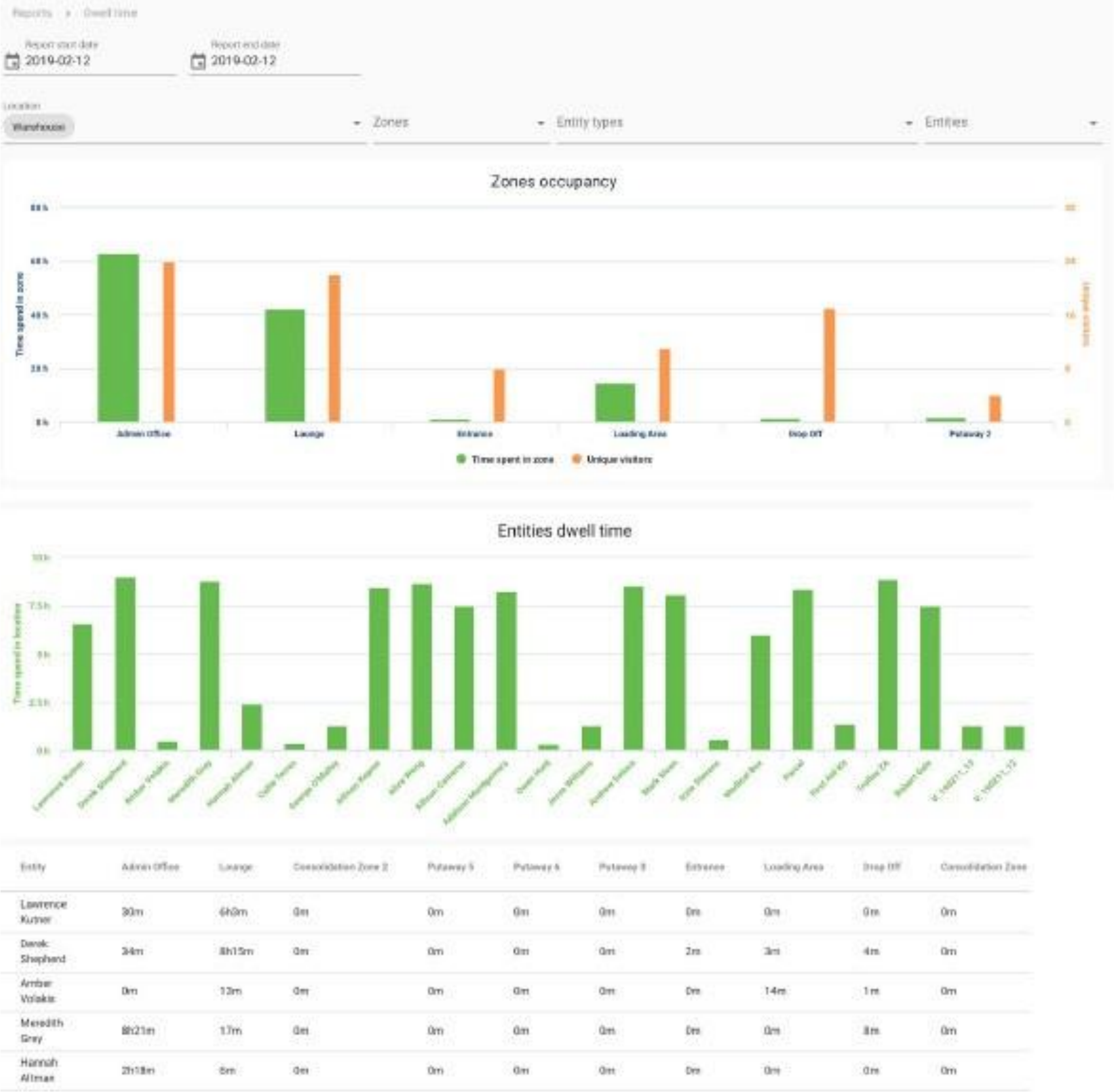
12/02/2019, 22:11:19 Resolved Robert had dropped a sledgehammer on his toe, and is now on his way to hospital. HR and shift manager has been informed.

12/02/2019, 22:09:29 Acknowledged I will go to check Robert.

12/02/2019, 21:53:40 New

Main view: Reports | Sub-view: Dwell time Report

Report view includes customer includes some general reports as well as works as a placeholder for some customer and use-case driven reports. One of common reports is called Dwell Time that essentially breaks down Entities' presence into zonal dwell times per chosen time. Dwell time report enables user to choose the timeframe, Entity Types, Entities, and Locations as per her interests.



Reports › Dwell time

Report start date: 2019-02-12

Report end date: 2019-02-12

Location: Warehouse

Entity	Admin Office	Lounge	Consolidation Zone 2	Putaway 5	Putaway 6	Putaway 8	Entrance	Loading Area	Drop Off	Consolidation Zone
Lawrence Kutner	30m	6h0m	0m	0m	0m	0m	0m	0m	0m	0m
Derek Shephard	34m	8h15m	0m	0m	0m	0m	2m	3m	4m	0m
Amber Volakis	0m	13m	0m	0m	0m	0m	0m	14m	1m	0m
Meredith Grey	8h21m	17m	0m	0m	0m	0m	0m	0m	8m	0m
Hannah Altman	2h18m	6m	0m	0m	0m	0m	0m	0m	0m	0m

VERIFICATION & TROUBLESHOOTING

The overall system operation can be verified by making sure that any action on the lock (such as open/close, use of various test cards, etc.) is reflected in the Vision Line Server. Conversely, any control action (such as door open/close) initiated in the Vision Line Server should be reflected in a corresponding action in the chosen lock.

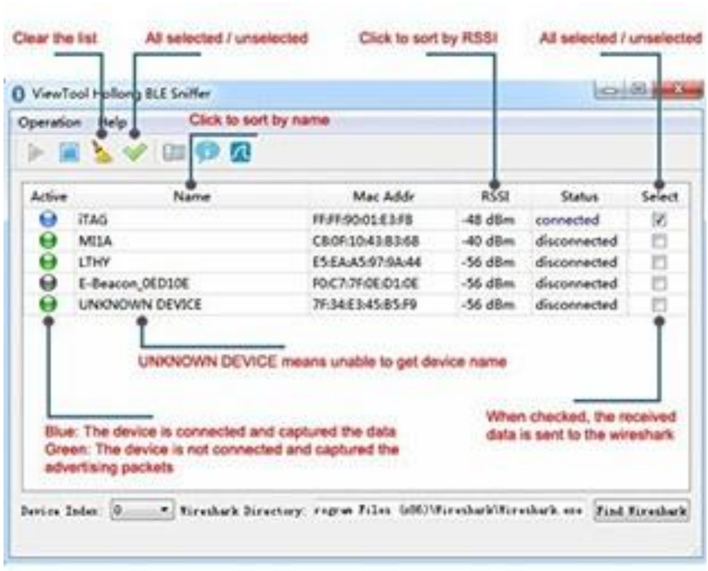
The following troubleshooting tools can be used to verify communications on the various segments of the Kontakt.io Tags.

BLE SNIFFER

Hollong Full Channel Professional Bluetooth 4.0/4.1/4.2 BLE Sniffer Protocol

https://www.amazon.com/gp/product/B075K38YT2/ref=ppx_yo_dt_b_asin_title_o02_o00_s00?ie=UTF8&psc=1

https://www.youtube.com/watch?v=KNjQCVndUvM


Clear the list
All selected / unselected
Click to sort by RSSI
All selected / unselected
ViewTool Hollong BLE Sniffer
Operation
Help
Click to sort by name

Active	Name	Mac Addr	RSSI	Status	Select
●	iTAG	FF:FF:90:01:E3:F8	-48 dBm	connected	☑
●	MI1A	C8:0F:10:43:83:68	-40 dBm	disconnected	□
●	LTHY	E5:EA:A5:97:9A:44	-56 dBm	disconnected	□
●	E-Beacon_0ED10E	F0:C7:7F:0E:D1:0E	-56 dBm	disconnected	□
●	UNKNOWN DEVICE	7F:34:E3:45:B5:F9	-56 dBm	disconnected	□

UNKNOWN DEVICE means unable to get device name

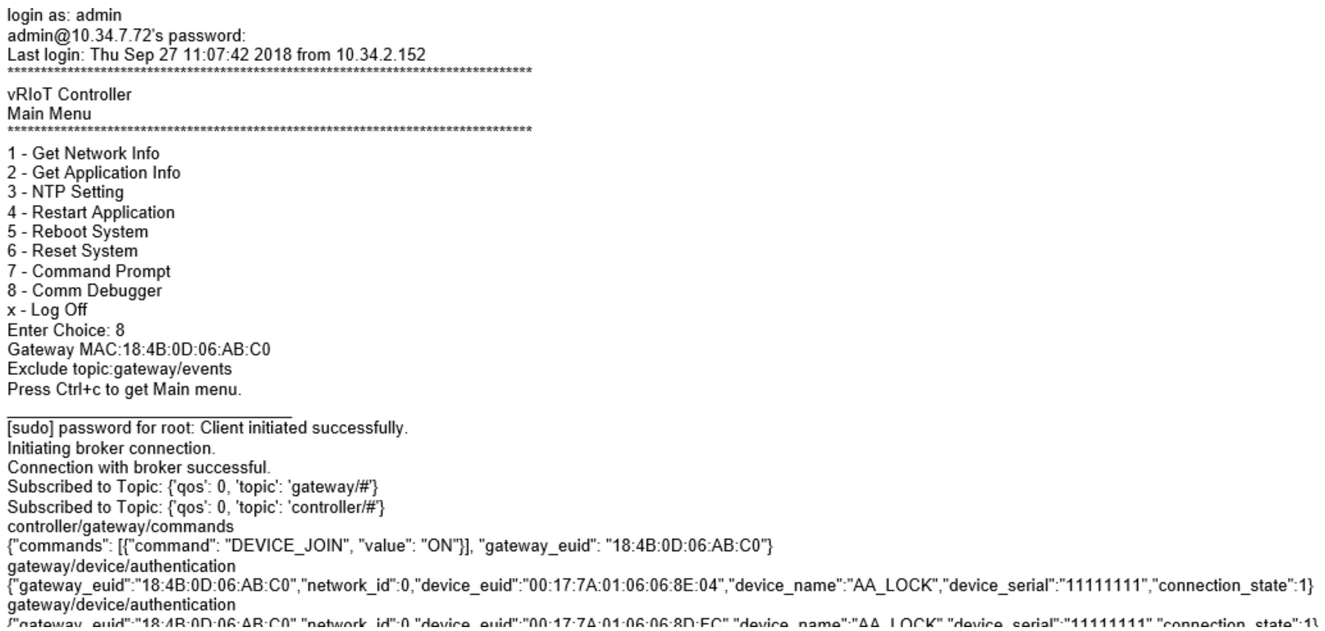
Blue: The device is connected and captured the data
Green: The device is not connected and captured the advertising packets

When checked, the received data is sent to the wireshark

Device Index: 0 Wireshark Directory: rogram Files (x86)\Wireshark\Wireshark.exe Find Wireshark

IOT CONTROLLER COMMUNICATIONS DEBUGGER

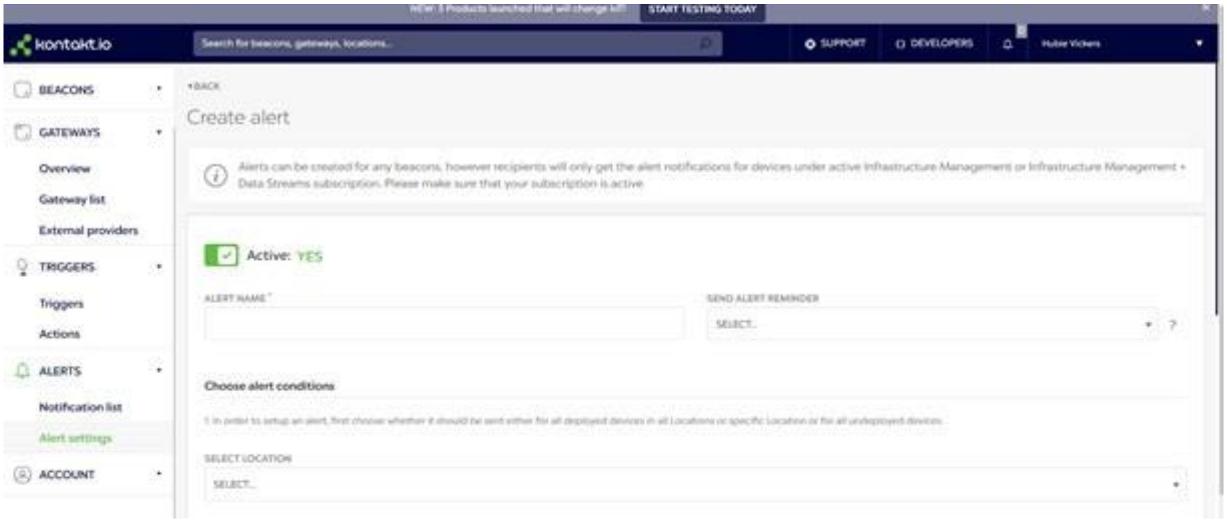
Option 8 on the IoT controller invokes the Comm Debugger, which can be used to examine AP to IoT Controller communications.



```
login as: admin
admin@10.34.7.72's password:
Last login: Thu Sep 27 11:07:42 2018 from 10.34.2.152
*******************************************************************************
vRIoT Controller
Main Menu
*******************************************************************************
1 - Get Network Info
2 - Get Application Info
3 - NTP Setting
4 - Restart Application
5 - Reboot System
6 - Reset System
7 - Command Prompt
8 - Comm Debugger
x - Log Off
Enter Choice: 8
Gateway MAC:18:4B:0D:06:AB:C0
Exclude topic:gateway/events
Press Ctrl+c to get Main menu.
______________________________
[sudo] password for root: Client initiated successfully.
Initiating broker connection.
Connection with broker successful.
Subscribed to Topic: {'qos': 0, 'topic': 'gateway/#'}
Subscribed to Topic: {'qos': 0, 'topic': 'controller/#'}
controller/gateway/commands
{"commands": [{"command": "DEVICE_JOIN", "value": "ON"}], "gateway_euid": "18:4B:0D:06:AB:C0"}
gateway/device/authentication
{"gateway_euid":"18:4B:0D:06:AB:C0","network_id":0,"device_euid":"00:17:7A:01:06:06:8E:04","device_name":"AA_LOCK","device_serial":"11111111","connection_state":1}
gateway/device/authentication
{"gateway_euid":"18:4B:0D:06:AB:C0","network_id":0,"device_euid":"00:17:7A:01:06:06:8D:EC","device_name":"AA_LOCK","device_serial":"11111111","connection_state":1}
```

KONTAKT.IO EVENTS MONITOR

This can be customized to present alarms and events per site.



Create alert

Alerts can be created for any beacons, however recipients will only get the alert notifications for devices under active Infrastructure Management or Infrastructure Management + Data Streams subscription. Please make sure that your subscription is active.

Active: YES

ALERT NAME *

SEND ALERT REMINDER: SELECT...

Choose alert conditions

1. In order to setup an alert, first choose whether it should be sent either for all deployed devices in all Locations or specific Location or for all undeployed devices.

SELECT LOCATION: SELECT...

About Ruckus Networks

Ruckus Networks enables organizations of all sizes to deliver great connectivity experiences. Ruckus delivers secure access networks to delight users while easing the IT burden, affordably. Organizations turn to Ruckus to make their networks simpler to manage and to better meet their users' expectations. For more information, visit www.ruckuswireless.com.

Ruckus Networks | 350 West Java Drive | Sunnyvale, CA 94089 USA | T: (650) 265-4200 | F: (408) 738-2065 ruckuswireless.com

About ARRIS

ARRIS International plc (NASDAQ: ARRS) is powering a smart, connected world. The company's leading hardware, software and services transform the way that people and businesses stay informed, entertained and connected. For more information, visit www.arris.com.

For the latest ARRIS news:

Check out our blog: ARRIS EVERYWHERE
Follow us on Twitter: @ARRIS