RAK7289V2/RAK7289CV2 Supported LoRa Network Servers

AWS IoT Core for LoRaWAN

Execute the following steps to set up your AWS account and permissions:

Set up Roles and Policies in IAM Add an IAM Role for CUPS Server

Adding an IAM role will allow the Configuration and Update Server (CUPS) to handle the wireless gateway credentials.

This procedure needs to be done only once, but must be performed before a LoRaWAN gateway tries to connect with AWS IoT Core for LoRaWAN.

- 1. Go to the IAM Roles 🖄 page on the IAM console.
- 2. Choose Create role.
- 3. On the Create Role page, choose Another AWS account.
- 4. Enter your Account ID, then select Next.
- 5. In the search box next to the Filter Policies, type AWSIoTWirelessGatewayCertManager.
 - If the search results show the policy named AWSIoTWirelessGatewayCertManager, select it by clicking the checkbox.
 - If the policy does not exist, create one.
 - Choose Create Policy, then select the JSON tab to open the policy editor.
 - Replace the existing template with a trust policy document.

ł	json
"Version": "2012-10-17",	
"Statement": [
{	
"Sid": "IoTWirelessGatewayCertManager",	
"Effect": "Allow",	
"Action": [
"iot:CreateKeysAndCertificate",	
"iot:DescribeCertificate",	
"iot:ListCertificates",	
"iot:RegisterCertificate"	
],	
"Resource": "*"	
}	
]	
}	

- Select Next Tags and click Next: Review to open the Review Page.
- For the Name, type AWSIoTWirelessGatewayCertManager.

NOTE:

You must enter the name as *AWSIoTWirelessGatewayCertManager* and must not use a different name. This is for consistency with future releases.

- For the **Description**, enter a description of your choice.
- Then choose **Create policy**. You will see a confirmation message showing the policy has been created.
- 6. Click Next.

7. In Role name, enter IoTWirelessGatewayCertManagerRole, and then choose to Create role.

📝 NOTE:

You must not use a different name. This is for consistency with future releases.

- 8. In the confirmation message, choose *IoTWirelessGatewayCertManagerRole* to edit the new role.
- 9. In the Summary, choose the Trust relationships tab, and then choose Edit trust relationship.
- 10. In the Policy Document, change the Principal property to represent the IoT Wireless service:



• After changing the Principal property, the complete policy document should look like the following:



11. Choose **Update Trust Policy** to save your changes and exit. At this point, you have created the *IoTWirelessGatewayCertManagerRole* and you won't need to do this again.

NOTE:

The examples in this document are intended only for dev environments. All devices in your fleet must have credentials with privileges that authorize only intended actions on specific resources. The specific permission policies can vary for your use case. Identify the permission policies that best meet your business and security requirements. For more information, refer to **Example Policies** and **Security Best Practices**

Add IAM Role for Destination to AWS IoT Core for LoRaWAN

Creating a Policy

Creating a policy gives the role permissions to describe the IoT endpoint and publish messages to AWS IoT.

- 1. Go to the IAM console 🖄 .
- 2. Choose **Policies** from the navigation pane.
- 3. Choose Create Policy, then choose the JSON tab to open the policy editor. Replace the existing template with this trust policy document:



- 4. Click Next: Tags then Next: Review to open the Review page.
- 5. For Name, enter a name of your choice.
- 6. For **Description**, enter a description of your choice.
- 7. Choose Create policy. You will see a confirmation message indicating that the policy has been created.

Creating the Role

- 1. In the IAM console, choose Roles from the navigation pane to open the Roles page.
- 2. Choose Create Role.
- 3. In Select type of trusted entity, select AWS account and click Next.
- 4. In Account ID, enter your AWS account ID, and then choose Next: Permissions.
- 5. Search for the IAM policy you just created by entering the policy name in the search bar.
- 6. In the search results, select the checkbox corresponding to the policy.
- 7. Click Next.
- 8. For **Role name**, enter an appropriate name of your choice.
- 9. For **Description**, enter a description of your choice.
- 10. Choose Create role. You will see a confirmation message indicating that your role has been created.

Updating your Trust Policy

Update your role's trust relationship to grant AWS IoT Core for LoRaWAN permission to assume this IAM role when delivering messages from devices to your account.

- 1. In the IAM console, choose **Roles** from the navigation pane to open the Roles page.
- 2. Enter the name of the role you created earlier in the search window, and click on the role name in the search results. This opens up the **Summary** page.
- 3. Choose the Trust relationships table to navigate to the Trust relationships page.

 Click Edit trust relationship. The principal AWS role in your trust policy document defaults to root and must be changed. Replace the existing policy with this:



5. Choose Update Trust Policy.

Add the Gateway to AWS IoT

Requirements

To complete setting up your gateway, you need the following:

- LoRaWAN region. For example, if the gateway is deployed in a US region, the gateway must support LoRaWAN region US915.
- Gateway LNS-protocols. Currently, the LoRa Basics Station protocol is supported.
- Gateway ID (GatewayEUI) or serial number. This is used to establish the connection between the LNS and the gateway. Consult the documentation for your gateway to locate this value.
- Add minimum software versions required, including Basics Station 2.0.5.

Add the LoRaWAN Gateway

To register the gateway with AWS IoT Core for LoRaWAN, execute these steps:

- 1. Go to the AWS IoT console \square .
- 2. Select Wireless connectivity in the navigation panel on the left.
- 3. Choose Gateways, and then click Add Gateway.
- 4. In the Add gateway section, fill in the GatewayEUI and Frequency band (RF Region) fields.
- Enter a descriptive name in the Name optional field. It is recommended that you use the GatewayEUI as the name.
- 6. Choose Add gateway.
- 7. On the **Configure your Gateway** page, find the section titled **Gateway certificate**.
- 8. Select Create certificate.
- Once the Certificate created and associated with your gateway message is shown, select Download certificates to download the certificate (*xxxxx.cert.pem*) and private key (*xxxxx.private.key*).
- 10. In the section **Provisioning credentials**, choose **Download server trust certificates** to download the **CUPS** (cups.trust) and LNS (Ins.trust) server trust certificates.
- 11. Copy the CUPS and LNS endpoints and save them in a .txt file for use while configuring the gateway.
- 12. Choose **Submit** to add the gateway.

Add a LoRaWAN Device to AWS IoT

Requirements:

- Locate and note the following specifications about your endpoint device.
 - LoRaWAN Region This must match the gateway LoRaWAN region. The following Frequency bands (RF regions) are supported: o EU868 o US915 o EU433
 - MAC Version This must be one of the following: o V1.0.2 o v1.0.3 o v1.1
 - OTAA v1.0x and OTAA v1.1 are supported.
 - $\circ~$ ABP v1.0x and ABP v1.1 are supported.
- · Locate and note the following information from your device manufacturer:
 - For OTAA v1.0x devices: DevEUI, AppKey, AppEUI
 - For OTAA v1.1 devices: DevEUI, AppKey, NwkKey, JoinEUI
 - For ABP v1.0x devices: DevEUI, DevAddr, NwkSkey, AppSkey
 - For ABP v1.1 devices: DevEUI, DevAddr, NwkSEnckey, FNwkSIntKey, SNwkSIntKey, AppSKey

Verify Profiles

AWS IOT Core for LoRaWAN supports device profiles and service profiles. Device profiles contain the communication and protocol parameter values the device needs to communicate with the network server. Service profiles describe the communication parameters the device needs to communicate with the application server.

Some pre-defined profiles are available for device and service profiles. Before proceeding, verify that these profile settings match the devices you will be setting up to work with AWS IoT Core for LoRaWAN.

- 1. Navigate to the AWS IoT console 🖸 . In the navigation pane, choose Wireless connectivity then click Profiles.
- 2. In the **Device Profiles** section, there are some pre-defined profiles listed.
- 3. Check each of the profiles to determine if one of them will work for you. If not, select **Add device profile** and set up the parameters as needed. For US 915 as an example, the values are:
 - MacVersion 1.0.3
 - RegParamsRevision RP002-1.0.1
 - MaxEirp 10
 - MaxDutyCycle 10
 - RfRegion US915
 - SupportsJoin true
- 4. Click Add device profile once you have set a device profile that will work for you.
- 5. In the Service Profiles section, click Add service profile and set up the parameters as needed. As an example, the default service profile parameters are shown below. However, only the AddGwMetadata setting can be changed at this time.
 - UIRate 60
 - UlBucketSize 4096
 - DIRate 60
 - DIBucketSize 4096
 - AddGwMetadata true
 - DevStatusReqFreq 24
 - DrMax 15
 - TargetPer 5

- MinGwDiversity 1
- 6. Proceed only if you have a device and service profile that will work for you.

Set up a Destination for Device Traffic

Because most LoRaWAN devices don't send data to AWS IoT Core for LoRaWAN in a format that can be consumed by AWS services, traffic must first be sent to a Destination. A Destination represents the AWS IoT rule that processes a device's data for use by AWS services. This AWS IoT rule contains the SQL statement that selects the device's data and the topic rule actions that send the result of the SQL statement to the services that will use it.

For more information on Destinations, refer to the AWS LoRaWAN Developer Guide 🗹 .

A destination consists of a Rule and a Role. To set up the destination, execute the following steps:

- 1. Navigate to the AWS IoT console 2 . In the navigation pane, choose Wireless connectivity, and then Destinations.
- 2. Choose Add Destination.
- 3. For the **Destination name**, enter *ProcessLoRa*, and then add an appropriate description under **Destination** description optional.
- 4. For **Rule name**, enter *LoRaWANRouting*. Ignore the section **Rules configuration Optional** for now. The Rule will be set up later in the "Hello World" sample application. See Create the IoT Rule for the destination.
- 5. In the **Permissions** section, choose **Select an existing service role** and select the IAM role you had created earlier, from the drop-down.

NOTE:

The Destination name can be anything. For getting started and consistency, choose **ProcessLoRa** for the first integration with AWS IoT Core for LoRaWAN.

5. Choose **Add Destination**. You will see a message "*Destination added*", indicating the destination has been successfully added.

Register the Device

Now, register an endpoint device with AWS IoT Core for LoRaWAN as follows:

- 1. Go to the AWS IoT console 🗹 .
- 2. Select Wireless connectivity in the navigation panel on the left.
- 3. Select **Devices**, then choose **Add wireless device**.
- On the Add device page, select the LoRaWAN specification version in the drop-down under Wireless device specification.
- Under LoRaWAN specification and wireless device configuration, enter the DevEUI and confirm it in the Confirm DevEUI field.
- 6. Enter the remaining fields as per the OTAA/ABP choice you made above.
- 7. Enter a name for your device in the Wireless device name optional field.
- 8. In the **Profiles** section, under **Wireless device profile**, from the drop-down option find the device profile you have created or the one that corresponds to your device and region.

VOTE:

Compare your device details to ensure the device profile is correct. If there are no valid default options, you will have to create a new profile. See the Verify Profiles section.

- 9. Click **Next** and then choose the destination you created earlier (*ProcessLoRa*) from the drop-down under **Choose destination**.
- 10. Choose **Add device**. You will see a message saying "*Wireless device added*", indicating that your device has been set up successfully.

Set up the Gateway

- Set up the Gateway Hardware \square
- Set up the Gateway Software $\ensuremath{\mathbb{Z}}$

Configure the Gateway Device

1. Using your preferred Web browser, access the gateway. To access the gateway, see the Quick Start guide.

W/S Gate		1	
Login Login Posward	Ø Login		
		Simple access to your gateways Set up, connect, monitor, analyze and run gateways.	
©2022 RAFinitiaises Te			

Figure 1: Web User Interface Log-in

Come Gate	RAK7268C Overview LoRa Statistics	
.≙ .∿ ⊙	RAK7268C O serial number	Packet capture
	ACIF09FF N/A EUI REPORTED COORDINATES ACIF:09: 868 MHz	Performance
	MAC ADDRESS FREDUENCY BAND 8-CHANNEL 12d 1h 16m 6s NUMBER OF CHANNELS UPTIME Wed Apr 13 15:21:51 2022 LOCAL TIME	сри (т) 25 / 100% ШШШШ (ППП) (ППП) (ППП) (ППП) исмоят (т) 51
	WisCateOS_2.x_master Firmware details	
	WAN Interfaces	
	 Ethernet 192.168.0.151 19.4558555 	

Figure 2: Checking the Firmware Version

- 2. Configure Network Mode to Basics Station.
- 3. Navigate to LoRa. For Work mode, select Basics station and click Configure Basics Station server setup to expand the Basics Station settings.

SI Cate	RAK7268C	
₩ ∴ ~	Work mode	Packet forwarder Basics station Built-in network server
*	Log Level	Log Level V DEBUG V
	Basics station server setup	Configure Basics Station server setup. Basics Station Server Type CUPS-BOOT Server CUPS Server UNS Server UNS Server
		Server URL Server Port Authentication Mode None
		Save changes

Figure 3: Basics Station work mode

4. Select LNS Server from Server, then choose TLS Server and Client Authentication from Authentication Mode.

on server setup		
	Configure Basics Station server setup.	
	Basics Station Server Type	
	CUPS-BOOT Server	
	CUPS Server	
	LNS Server	
	Server URL	Server Port
	Authentication Mode	
	TLS Server & Client Authentication	-
	Trust (CA Certificate)	Can
	Drop your certificate file here or	
	choose file	
	Client certificate	Can
	Drop your certificate file here or choose file	
	Client key	Can
	Drop your certificate file bere or	
	choose file	

Figure 4: Configuring Network Mode to Basics Station

5. Configure URI, Port, and Authentication Mode.

Basics Station Server Type CUPS-BOOT Server CUPS Server CUPS Server CUPS Server Server URL Server URL Server VRL Create certificate files were created. Download them and save them to upload to your gateway. Gateway certificate file Create certif
CUPS-BOOT Server CUPS Server INS Server Server URL Server Port Gateway certificate file Cert.pem
CUPS Server ● LNS Server Server URL Server Port Gateway certificate file Create certificate file
ENS Server Server URL Server Port Gateway certificate file Cert.pem
Server URL Server Port Gateway certificate file .cert.pem
Gateway certificate file .cert.pem
Authentication Mode .private.key
TLS Server & Client Authentication Download certificate files
Trust (CA Certificate)
Provisioning credentials Info Choose file Choose the endpoint that your gateway supports. Then, copy the endpoint and download the server trust certificate so that you can them to your gateway.
Client certificate
Drop your certificate file here or CUPS (Configuration and Update Server) endpoint
https:// .cups.lorawan.us-east-1.amazonaws.com:443
Chent key LNS (LoRaWAN Network Server) endpoint
Drop your certificate file here or wss:// .lns.lorawan.us-east-1.amazonaws.com:443
Server trust certificates
Download your server trust certificate so you can upload the certificate for the endpoint your gateway supports.
Download server trust certificates

Figure 5: Configuring URI, Port, and Authentication Mode

6. Click **Save**. Check if the gateway is online in AWS IoT console.

LoRaWAN specific detai	ls					
GatewayEUI	RFRegion	LastUplinkReceivedAt	Connection status	JoinEuiFilters	NetIdFilters	SubBands
ac1f09ff	EU868	June 07, 2022, 10:26:33 (UTC+0300)	Connected	-	-	-

Figure 6: Verifying Operation

Add End Devices

This section shows an example of how to join the AWS IoT LoRaWAN server.

1. Add Device Profile.

Default profiles are based on your selected LoRaWAN OTAA device sustomized your profile per your device vendor specifications.	class and your LoRaWAN radio frequency band. You may need to
EU868 - A	•
Device profile name Type a descriptive name for this device profile.	Frequency band (RFRegion) Choose the LoRa supported frequency band for this profile.
rak4200	EU868
MAC version The MACVersion of the LoRaWAN devices that use this profile.	Regional parameters version
	Select the region parameters version identifier for this profile.
1.0.3 MaxEIRP inter the MaxEIRP	RP002-1.0.1 (recommended)
1.0.3 MaxEIRP Enter the MaxEIRP value for this device profile. 5 Supports Class B Choose to enter the values for Class B support.	RP002-1.0.1 (recommended)
1.0.3 MaxEIRP Enter the MaxEIRP value for this device profile. 5 Supports Class B Choose to enter the values for Class B support. Supports Class C Choose to enter the values for Class C support.	RP002-1.0.1 (recommended)

Figure 7: Adding the Device Profile

2. Add Service Profile.

Service profile Info A service profile describes the features that are enabled for the user(s), and the rate of messages the	nat can be sent over the network.
Service profile name - optional Enter a descriptive profile name.	
rak4200	
AddGWMetaData Add additional gateway metadata (RSSI, SNR, GW geoloc., etc.) to the packets sent by device	s.
Tags - <i>optional</i>	
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value your resources or track your AWS costs.	e. You can use tags to search and filter
You don't have any tags attached to this resource.	
Add new tag	
You can add up to 50 tags.	

Figure 8: Adding the Service Profile

3. Add Destination.

Before adding the destination, follow the Add IAM role for Destination to AWS IoT Core for LoRaWAN section to configure the IAM policy and role.

e destination name appears in the device and gateway destinati	ion selection lists.
ProcessLoRa	
Pestination description - optional rovide a helpful description of your destination.	
Destination description.	
• Enter a rule name	Publish to AWS IoT Core message broker
• Enter a rule name Enter the name of the rule or a rule/topic that will process the messages sent to this destination.	Publish to AWS IoT Core message broker If you need a publish/subscribe broker to distribute messages to multiple subscribers

Figure 9: Adding Destination

4. Add Device.

Before adding a device to AWS IoT, retrieve the **DevEui**, **AppEui**, and **AppKey** from the end Device's console. You can use AT command at+get_config=lora:status to obtain the information.

For more AT commands, refer to the RAK4200 AT Command Manual.



Wireless device specification Your device specifications consist of the LoRaWAN version (1.1 or 1.0 Authentication By Personalization). Once selected, your data is encry	0.x) and your authentication process (Over The Air Authentication or pted with a key that AWS owns and manages for you.
OTAA v1.0.x	
DevEUI	Confirm DevEUI
70B3	70B3
The 16-digit hexadecimal DevEUI value found on your wireless device.	Reenter the DevEUI.
АррКеу	Confirm AppKey
F573D2	F573D2
The 32-digit hexadecimal AppKey value that your wireless device vendor provided.	Reenter the AppKey.
AppEUI	Confirm AppEUI
70B3D	70B3D
The 16-digit hexadecimal AppEUI that your wireless device vendor provided. Wireless device name - optional	Reenter the AppEUI.
Device name	
A descriptive name to make the wireless device easier to locate.	
Wireless device description - optional	

Figure 10: LoRaWAN specifications and wireless device configuration

Profiles	
Nireless device profile	
noose a wireless device profile so your device can pass the correct messages to your gateway.	
Service profile	
Choose a service profile.	
rak4200	•

Figure 11: Choosing a Wireless Device Profile

Choose destination	
Destination name Destinations route LoRaWAN messages from your wireless device to other AWS services.	
ProcessLoRa	▼

Figure 12: Choosing a Destination

5. Restart the end Device, and it should join the AWS IoT LoRaWAN server.

EVENT:0:STARTUP		
SYSLOG:4:0TAA Join Request		
SYSLOG:4:0TAA Join Success		
EVENT:1:JOIN_NETWORK		
SYSLOG:4:LoRa Tx :		

Device ID Associated thing name	Name rak4200 Description -			Destination ProcessLoRa Last uplink received at June 07, 2022, 10:13:23 (UTC+0300)			
Device traffic Clicking refreshing button will incur costs C							
Last connected gateway	DevEUI	RSSI (dBm)	SNR (dB)	Frequency	Data rate		
ac1f09ff <mark>1</mark>	70b3d	-57	13.75	867100000	5		



Verifying Operation

Once setup is completed, provisioned OTAA devices can join the network and start to send messages. Messages from devices can then be received by AWS IoT Core for LoRaWAN and forwarded to the IoT Rules Engine.

Instructions for a sample Hello World application are given below, assuming that the device has joined and is capable of sending uplink traffic.





Create a Lambda Function for Destination Rule

Create the lambda function to process device messages processed by the destination rule.

- 1. Go to the AWS Lambda console $\ensuremath{\square}$.
- 2. Click **Functions** in the navigation pane.
- 3. Click Create function.
- 4. Select **Author** from scratch.
- 5. Under **Basic Information**, enter the function name and choose *Runtime Python 3.8*. from the drop-down under **Runtime**.
- 6. Click Create function.
- 7. Under Function code, paste the copied code into the editor under the *lambda_function.py* tab.

```
import base64
import json
import logging
import ctypes
import boto3
# define function name
FUNCTION_NAME = 'RAK-HelloWorld'
# Second Byte in Payload represents Data Types
# Low Power Payload Reference: https://developers.mydevices.com/cayenne/docs/lora/
DATA_TYPES = 1
TYPE_TEMP = 0 \times 67
client = boto3.client('iot-data')
logger = logging.getLogger(FUNCTION_NAME)
logger.setLevel(logging.INFO)
def decode(event):
    data_base64 = event.get('PayloadData')
    data_decoded = base64.b64decode(data_base64)
    result = {
        'devEui': event.get('WirelessMetadata').get('LoRaWAN'
                ).get('DevEui'),
        'fPort': event.get('WirelessMetadata').get('LoRaWAN'
                ).get('FPort'),
        'freq': event.get('WirelessMetadata').get('LoRaWAN'
                ).get('Frequency'),
        'timestamp': event.get('WirelessMetadata').get('LoRaWAN'
                ).get('Timestamp'),
    if data_decoded[DATA_TYPES] == TYPE_TEMP:
        temp = data_decoded[DATA_TYPES + 1] << 8 \land
            | data_decoded[DATA_TYPES + 2]
        temp = ctypes.c_int16(temp).value
        result['temperature'] = temp / 10
    return result
def lambda_handler(event, context):
    data = decode(event)
    logger.info('Data: %s' % json.dumps(data))
    response = client.publish(topic=event.get('WirelessMetadata'
                              ).get('LoRaWAN').get('DevEui')
                               + '/project/sensor/decoded', qos=0,
                              payload=json.dumps(data))
    return response
```

ру

- 8. Once the code has been pasted, choose **Deploy** to deploy the lambda code.
- 9. Click the **Configuration** tab of the lambda function and click **Permissions**.
- 10. Change the Lambda Role Policy permission.
 - $\circ~$ Under $\mbox{Execution role},$ click the hyperlink under $\mbox{Role name}.$
 - $\circ~$ On the $\ensuremath{\text{Permissions tab}},$ find the policy name and select it.
 - Choose Edit policy, and choose the JSON tab.
 - · Append the following to the Statement section of the policy to allow publishing to AWS IoT.



· After the change the code should look like this:



- Choose Review Policy, then Save changes.
- 11. Go back to the Lambda function **Code source** and create a test event that will allow you to test the functionality of the lambda function.
- Click **Test** next to **Deploy**.
- In the Configure test event, enter a name for the test event in the Event name field.
- Paste the following sample payload in the area under Event JSON field:

```
jon
{
"WirelessDeviceId": "65d128ab-90dd-4668-9556-fe47c589610b",
"PayloadData": "AWf/1w==",
"WirelessMetadata": {
    "LoRawAN": {
    "DotRawAN": {
    "DotVEUI": "00000000000088",
    "FPort": 1,
    "Frequency": "868100000",
    "GatewayS": [
        {
        "GatewayEuI": "80029cffXXXXXXX",
    "Rssi": -109,
    "Snr": 5
        }
     ],
    "Timestamp": "2021-02-08T04:00:402"
     }
}
```

- 11. Choose **Save** to save the event.
- 12. In a new window, navigate to the AWS IoT console, choose **Test** on the navigation pane, and select MQTT test client.
- 13. In Subscription topic field type "#" (all topics) and click Subscribe to topic. The MQTT will subscribe to all topics.
- 14. Click on Test in the Lambda function page to generate the test event you just created.
- 15. Verify the published data in the AWS IoT Core MQTT Test client. The output should look similar to this:



Create the Destination Rule

In this section, create the IoT rule that forwards the device payload to your application. This rule is associated with the destination created earlier in Set up a Destination for Device Traffic section.

- 1. Navigate to the AWS IoT console 2 .
- 2. In the navigation pane, choose Act, then select Rules.
- 3. On the Rules page, choose Create.
- 4. On the Create a rule page, enter as follows:
 - Name: LoRaWANRouting
 - Description: Any description of your choice.

NOTE:

The **Name of your Rule** is the information needed when you provision devices to run on AWS IoT Core for LoRaWAN.

- 5. Leave the default Rule query statement: 'SELECT * FROM 'iot/topic' unchanged. This query has no effect at this time, as traffic is currently forwarded to the rules engine based on the destination.
- 6. Under Set one or more actions, choose Add action.
- 7. On the Select an action page, choose **Republish a message to an AWS IoT topic**. Scroll down and choose **Configure action**.
- 8. On the Configure action page, for Topic, enter *project/sensor/decoded*. The AWS IoT Rules Engine will forward messages to this topic.
- 9. Under Choose or create a role to grant AWS IoT access to perform this action, select Create Role.
- 10. For Name, enter a name of your choice.
- 11. Choose **Create role** to complete the role creation. You will see a "**Policy Attached**" tag next to the role name, indicating that the Rules Engine has been permitted to execute the action.
- 12. Choose Add action.
- 13. Add one more action to invoke the Lambda function. Under Set one or more actions, choose Add action.
- 14. Choose Send a message to a Lambda function.
- 15. Choose **Configure action**.
- 16. Select the Lambda function created earlier and choose Add action.
- 17. Then choose Create rule.
- 18. A "Success" message will be displayed at the top of the panel, and the destination has a rule bound to it.

You can now check that the decoded data is received and republished by AWS by triggering a condition or event on the device itself.

- Go to the AWS IoT console. In the navigation pane, select Test, and choose MQTT client.
- Subscribe to the wildcard topic '#' to receive messages from all topics.
- Send message from endDevice using AT command: at+send:lora:1:01670110 .
- You should see traffic similar to that shown below.

```
393331375d387505/project/sensor/decoded Februar
{
    "devEui": "393331375d387505",
    "fPort": 1,
    "freq": "867100000",
    "timestamp": "2021-02-09T06:47:20Z",
    "temperature": 27.2
}
```

February 09, 2021, 14:47:21 (UTC+0800)



Configuring Amazon SNS

You will be using the Amazon Simple Notification Service to send text messages (SMS) when certain conditions are met.

- 1. Go to the Amazon SNS console \square .
- 2. Select Text Messaging (SMS) and choose Publish text message.
- 3. Under Message type, select Promotional.
- 4. Enter your phone number (phone number that will receive text alerts).
- 5. Enter "Test message" for the Message and choose Publish message.
- 6. If the phone number you entered is valid, you will receive a text message and your phone number will be confirmed.
- 7. Create an Amazon SNS Topic as follows:
 - In the navigation pane, choose **Topics**.
 - Select Create topic.
 - Under Type, select Standard.
 - Enter a name of your choice. Here, you will use "text_topic".
 - Select Create topic.
- 8. Create a subscription for this topic:
 - On the newly created *text_topic* page, choose the **Subscriptions** tab.
 - Choose Create subscription.
 - In **Topic ARN**, choose the topic you have created earlier.
 - Select **Protocol** as **SMS** from the drop-down.
 - Under Endpoint, enter the previously validated phone number to receive the SMS alerts.
 - Choose Create subscription. You should see a "_Subscription to text_topic created successfully_" message.

Add a Rule for Amazon SNS Notification

Now, add a new rule to send an Amazon SNS notification when certain conditions are met in a decoded message.

1. Navigate to the AWS IoT console 2 .

- 2. In the navigation pane, choose **Act**. Then, choose **Rules**.
- 3. On the Rules page, choose **Create**.
- 4. Enter the Name as *text_alert* and provide an appropriate Description.
- 5. Under the Rule query statement, enter the following query:

SELECT devEui as device_id, "Temperature exceeded 25" as message, temperature as temp, timest

- 6. Under Set one or more actions, choose Add action.
- 7. Choose Send a message as an SNS push notification.
- 8. Choose Configure action.
- 9. Under SNS target, select *text_topic* from the drop-down.
- 10. Select RAW under Message format.
- 11. Under Choose or create a role to grant AWS IoT access to perform this action, choose Create role.
- 12. Enter a name for the role and choose **Create role**.
- 13. Choose **Create rule**. You should see a "**Success**" message, indicating that the rule has been created.

Test the Rule for Amazon SNS Notification

After adding the rule for Amazon SNS notification, you should receive a text message when hitting the event.

Wait for an uplink from the device. Here is the message from mobile after sending an uplink message.



Send Downlink Payload

This section shows how to send downlink payload from AWS IoT LoRaWAN Server to the end Device.

- 1. Install the AWS SAM CLI
- 2. Deploy SAM template to AWS
- 3. Send Payload to End Device.
 - Go to the AWS IoT console.
 - In the navigation pane, select Test and choose MQTT client.
 - Subscribe to the wildcard topic '#' to receive messages from all topics.
 - Specify the topic to *cmd/downlink/{WirelessDeviceId}* and a base64-encoded message.

Subscriptions	#	Export Clear Pause
Subscribe to a topic Publish to a topic	Publish Specify a topic and a message to publish with a QoS of 0.	
• # *	cmd/downlink/3eff83dd-9159-	Publish to topic

Figure 15: Specifying a topic

4. You should see traffic on AWS similar as shown below:



Figure 16: Traffic on AWS

5. You should see traffic on your console of end device similar as shown below.



IoT Analytics

You will use IoT Analytics to visually display data via graphs if there is a need in the future to do further analysis.

Create an IoT Analytics Rule

Create a Rule First

- 1. Navigate to the AWS IoT console 🖸 .
- 2. In the navigation pane, choose **Act** and then, choose **Rules**.

- 3. On the Rules page, choose Create.
- 4. Enter the Name as Visualize, and provide an appropriate Description.
- 5. Under the Rule query statement, enter the following query:

SELECT * FROM 'project/sensor/decoded'

- 6. Choose Add action.
- 7. Select Send a message to IoT Analytics.
- 8. Choose Configure Action.
- 9. Choose Quick Create IoT Analytics Resources.
- 10. Under Resource Prefix, enter an appropriate prefix for your resources, such as LoRa.
- 11. Choose Quick Create.
- 12. Once the Quick Create Finished message is displayed, choose Add action.
- 13. Choose Create rule. You should see a Success message, indicating that the rule has been created.

Configure AWS IoT Analytics

Set up AWS IoT Analytics

- 1. Go to the AWS IoT Analytics console \square .
- 2. In the navigation panel, choose Datasets.
- 3. Select the data set generated by the Quick Create in Create an IoT Analytics Rule
- 4. In the Details section, click **Edit** in the **SQL query**.
- 5. Replace the query with as follows:

SELECT devEui as device_id, temperature as temp, timestamp as time FROM LoRa_datastore

- 6. Click **Update.
- 7. Navigate to **Schedule**, and click **Edit**.
- 8. Under Frequency, choose Every 1 minute, and then click Update.

Configure Amazon QuickSight

Amazon QuickSight lets you easily create and publish interactive BI dashboards that include Machine Learningpowered insights.

- 1. Go to AWS Management console 🗹 .
- 2. From the management console, enter QuickSight in the "Search for services, features.." search box.
- 3. Click on QuickSight in the search results.
- 4. If you haven't signed up for the service before, go ahead and sign up, as there is a free trial period.
- 5. Select the Standard Edition, and choose Continue.
- 6. Enter a unique name in the field QuickSight account name.
- 7. Fill in the Notification email address.
- 8. Review the other checkbox options and change them as necessary. The **AWS IoT Analytics** option must be selected.
- 9. Choose Finish. You will see a confirmation message.
- 10. Choose Go to Amazon QuickSight.
- 11. Select Datasets.
- 12. Select New dataset.
- 13. Select AWS IoT Analytics.

- 14. Under Select an AWS IoT Analytics data set to import, choose the data set created in **Create an IoT Analytics Rule**.
- 15. Choose Create data source, and then choose Visualize.
- 16. Select the dataset created, then select **Refresh** or **Schedule Refresh** for a periodic refresh of the dataset.

Testing Your "Hello Word" Application

Using your device, create a condition to generate an event such as a high-temperature condition. If the temperature is above the configured threshold then you will receive a text alert on your phone. This alert will include key parameters about the alert.

You can also visualize the data set as follows:

- 1. Go to the AWS IoT Analytics console \square .
- 2. Choose Data sets.
- 3. Select the dataset created earlier.
- 4. Select Content and ensure there are at least few uplink entries available in the data set.
- 5. Go to the QuickSight console \Box .
- 6. Choose New analysis.
- 7. Choose the dataset created in Create an IoT Analytics Rule.
- 8. Select time on the X-axis, Value as temp (Average), and Color as device_id to see a chart of your dataset.

Debugging

After login to the device using the web browser, the system log can be viewed from **Diagnostics** > **System Log**.

Diag	nostics	
Syste	n log Network utilities	Auto refre
wea	pr is 14:02:19 2022 user.notice lora_pkt_two[∠S920]: # ix errors: 0	
Wed	pr 13 14:02:19 2022 user.notice lora_pkt_fwd[23920]: ### SX1302 Status ###	
Wed	pr 13 14:02:19 2022 user.notice lora_pkt_fwd[23920]: # SX130X counter (INST): 90774116	
Wed	pr 13 14:02:19 2022 user.notice lora_pkt_twd[23920]: # SX130X time (PPS): 0, offset us 0	
wed	pr 13 14:02:19 2022 user.notice lora_pkt_twd[23920]: # BEACON queued: 0	
wed	pr 13 14:02:19 2022 User.notice lora_pkt_two[23920]: # BEALON sent so tar: 0	
Wed	pr 13 14:02:19 2022 User.notice lora_pkt_two[23920]: # BEALUN rejected: 0	
Wed	pr 13 14:02:19 2022 user.notice lora_pkt_twd[23920]: ### [Jii] ###	the sector
wed	pr 13 14:02:19 2022 User.into lora_pkt_two[23920]: src/jitqueue.c:441:jit_print_queue(): INFU: [jit] o	queue is empty
Wed	pr 13 14:02:19 2022 User.Holice tola_DKt_HW0[23920]: #	
Wed	pr 13 14:02:10 2022 User.Hotice tora_pkt_Hut(23020): #### [Ur3] ###	
Wed	pr 13 14:02:19 2022 User.notice lora_pkt_nwd[23920]: # Gr5 sync is disabled	
Wed	pr 13 14:02:19 2022 User notice tora_pkt_rwo[23920]: #### Concentrator[0] temperatore: 30 C ###	
Wed	pr 12 14.02.10 2022 User House Cola_pkt_Hud(22020). wwwwere but warman	
Wed	pr 13 14:02:19 2022 user.debug tora_pkt_wu[23920]: watchdog watt result 0 nr 13 14:02:20 2022 user info lora nkt fud(23920): JCCN un. /"etat"./"time"."2022_04_13 14:02:10 UTC"	" "rynh", a "rynk", a "ryfu", a "arkr", 100 a "dunh"
Wed	nr 12 14.02.20 2022 User info rostifu: 127 0.0 1 [12/Apr/2022 14.02.20] #5[27#6ET /dashbaard/lara	/etatietice WTTP/1 00100mH 200 -
Wed	nr 12 14.02.20 2022 user info loss akt fud(22020). TNED, (down) RULL ACK received in 74 me	Varactactes WHYPI.0010W 200 -
Wed	nr 13 14/02/23 2022 User info rostify: 127 0 0 1 [13/Anr/2022 14/02/23] "DI37mGET /eve/cottinne HT	TTP/1 80(8m" 288 -
Wed	nr 13 14:02:26 2022 user info lors nkt fud(230201; TNEO; (down) RULL ACK received in 74 me	1177.00[01 200 -
Wed	nr 13 14-02-20 2022 deel into toto projectimo (23020). Into: [down] toto_net received in 74 ms	
Wed	nr 13 14:02:29 2022 user info lora okt fwd[23920]: INED: host/radio[0] time offset=(1649858449 s:20600	87 µs) - drift≡-241 µs
Wed	nr 13 14:02:30 2022 user info restify: 127.0.0.1 = = [13/4or/2022 14:02:30] "D[37mGET /dian/syslon HT]	TP/1.0010m" 200 -
Wed	or 13 14:02:31 2022 user info lora okt fwd[239201: INFO: [down] PULL ACK received in 73 ms	
Wed	pr 13 14:02:34 2022 user.info restify: 127.0.0.1 – _ [13/Apr/2022 14:02:34] "∏[37mGET /diag/syslog HT]	TP/1.0⊓[0m" 200 -
Wed	pr 13 14:02:36 2022 user.info lora_pkt_fwd[23920]: INFO: [down] PULL ACK received in 74 ms	
Wed	pr 13 14:02:37 2022 user.info restify: 127.0.0.1 – – [13/Apr/2022 14:02:37] "∏[37mGET /diag/syslog HT]	TP/1.0⊓[0m" 200 -
Wed	pr 13 14:02:39 2022 user.debug lora_pkt_fwd[23920]: DEBUG: timersync wait result 145	
Wed	pr 13 14:02:39 2022 user.info lora_pkt_fwd[23920]: INFO: host/radio[0] time offset=(1649858449 s:20621	14 us) - drift=207 us

Figure 17: System Log

Troubleshooting

- 1. Unable to see the web login:
 - Check that your wifi is connected to RAK7268_XXXX.
 - Try ping **192.168.230.1**.
- 2. Lost password to login to the web login.
 - · Hold the reset button for 10 seconds to factory reset the device

AWS IoT Core Integration

AWS IoT Core Integration is a software service that enables your LoRaWAN gateway to work with AWS IoT Core. The AWS Integration for WisGate Edge V2 I tutorial will show you how to set up a LoRaWAN end-node and view its data on the AWS IoT Console. In addition, it'll show you how to send a message from AWS IoT Console to the end-node as well.

The Things Network (TTN)

WisGateOS 2 Basics Station to TTNv3

This tutorial illustrates how to configure and connect your RAK Edge Gateway V2 with WisGateOS 2 to a LoRaWAN Network Server by using the Basics Station protocol. For this example, it will be shown how to connect the gateway to TTNv3.

NOTE:

LoRa Basics Station is an implementation of a LoRa packet forwarder. This protocol simplifies the management of large-scale LoRaWAN Networks. More information about the Basics Station protocol can be found in the explanatory document is provided by Semtech.

Registering the Gateway

1. Log in first and head on to TTNv3 website 🖸 . If you already have a TTN account, you can use your The Things ID credentials to log in.

(3) THE THINGS ID	
THE THINGS STACK Community Edition	
Sign in with The Things ID Don't have an ID? Sign up for free	
Email Password 🗞	
Login with The Things ID	
Forgot password	

Figure 18: The Things Stack login page

VOTE:

This tutorial is for the EU868 Frequency band.

2. To register a commercial gateway, choose **Register a gateway** (for new users that do not already have a registered gateway) or go to **Gateways** > + **Add gateway** (for users that have registered gateways before).

THE THINGS NETWORX	THE THINOS STACK Community Edition	Overview	Applications	🚽 Gateways	Corganizations		⊕ EU1 Community No support plan ⑦	•
HITWOK					Welcome to the Cons Get started right away by creating an application or r Need help? Have a look at our @ Documentation 2	ole! egistering a gateway. or <u>Get support</u> @.		
				Creat	an application	Register a gateway		

Figure 19: Console Page after a successful login

- 3. You will be redirected to the **Register gateway** page.
- 4. In the Gateway EUI field, type the EUI of the gateway. The gateway's EUI can be found either on the sticker on the casing or by going to the LoRa Network Settings page in the LoRa Gateway menu accessible via the Web UI. Instructions on how to access your gateway via Web UI can be found in the product's Quick Start Guide Z.

THE THINGS NETWORK	THE THINGS STACK Community Edition	Overview	Applications	🝶 Gateways	At Organizations			EU1 Community No support plan ⑦		
			Register ga Register your gatew Learn more in our gr Owner* Gateway EUI Gateway EUI To continue, please	ateway ay to enable data t uide on a adding	affic between nearby end devices and the network.					
© 2022 The Th	ings Stack by The Things Network	and The Things Inc		⊕ en	v3.22.2 (3203f63b2)	Documentation	Status page	③ Get support		

Figure 20: Register gateway

- 5. After typing the EUI, click on **Confirm**. Additional fields will pop up. Fill in the following information:
- Gateway ID This will be the unique ID of your gateway in the Network. An ID based on the EUI is automatically generated. You can change it if you need. Note that the ID must contain only lowercase letters, numbers, and dashes (-).
- Gateway name Optionally, you can type a name for your gateway.
- Frequency plan The frequency plan used by the gateway.

VOTE:

- The other settings are optional and can be changed to satisfy your requirements.
- For this tutorial, we will use Europe 863-870 MHz (SF12 for RX2).

THE THINGS NET WORK	THE THINGS STACK Community Edition	Overview	Applications	🝶 Gateways	A Organizations			EU1 Community No support plan ⑦		
			Register ga Register your gatewa Learn more in our gu	Iteway Iy to enable data tra iide on Adding G	ffic between nearby end devices and the network.					
			Owner*							
			Gateway EUI ⑦ Gateway ID ⑦ *	1.11.11.10	leset					
			eui- Gateway name ②							
			Frequency plan () *		√					
			Require authent Choose this option e	ticated connection g. if your gateway i:	ී powered by <u>LoRa Basic Station</u> යි					
			Share gateway infor Select which informa	mation ation can be seen by hin network ①	other network participants, including <u>Packet Broker</u> 업					
			Register gatewa	y						
@ 2022 The Th	ings Stack by The Things Network	and The Things In	dustries				v2 22 2 (2202f62h2)	Documentation	Statue nano	🔊 Gateunnart



6. To register your gateway, click **Register gateway**.

THE THINGS STACK	E Overview D Applications	🛃 Gateways 🛛 🚢 Organiz	zations		H EU1 Community No SLA applicable ③	•
rak-edge-wis-gateway		Gateways > rak-edge-wis-ga	ateway			
Overview Uve data		rak-edge-w ID: rak-edge-wis-ga	is-gateway ^{ateway}			
9 Location		 Disconnected (2) 			1 Collaborator O O API keys	
Collaborators		General information Gateway ID	rak-edge-wis-gateway	Live data 15:39:87 Create gateway	See all activity →	
OT API keys		Gateway EUI	•			
General settings		Gateway description	None			
		Created at	Nov 17, 2022 15:39:07			
		Last updated at Gateway Server address	Nov 17, 2022 15:39:07			
		Global configuration	EU_863_870 Download global_conf.json	Location No location informatio	Change location settings	
< Hide sidebar						

Figure 22: Successfully added a gateway

TTNv3 supports TLS server authentication and Client token, which requires a trust file and a key file to configure the gateway to successfully connect it to the network.

Generating the Token

1. To generate a key file, from the **Overview page** of the registered gateway navigate to **API keys**.

THE THINOS STACK Community Edition	Sverview D Applications	🛃 Gateways 🛛 🚢 Organiz	ations		(Definition of the second seco							
		Gateways > rak-edge-wis-ga	teway									
rak-edge-wis-gateway												
Overview		rak-edge-wis-gateway										
Uve data		Disconnected		<u></u>	Collaborator 🛛 🗛 0 API keys							
Location												
Collaborators		General information		Live data	See all activity →							
		Gateway ID	rak-edge-wis-gateway	D15:39:87 Create gateway								
Ov API keys		Gateway EUI	•									
🕸 General settings		Gateway description	None									
		Created at	Nov 17, 2022 15:39:07									
		Last updated at	Nov 17, 2022 15:39:07									
		Gateway Server address	eui.cloud.thethings.network	Location	Change location settings →							
		LoRaWAN information										
		Frequency plan	EU_863_870									
		Global configuration	▲ Download global_conf.json									
				No location information wallable								
				No location mormation available								
< Hide sidebar												

Figure 23: Overview page

2. On the API keys page, choose + Add API key.

THE THINGS NET WORK	THE THINGS STACK Community Edition	Overview	Applications	🚽 Gateways	Organizations					EU1 Community to SLA applicable ⑦	3	
a rake	edge-wis-gateway			Gateways 🔪 i	- rak-edge-wis-gateway > API key	ys						
• 14. 1	coge mo gaterioj			API keys (0)					+ Add	NPI key		
Uver	view			Key ID 💠	Name	¢	Granted Rights		Create	iat 🔺		
💷 Live o	data						No items found					
Q Locat	tion											
🚜 Colla	borators											
Or API k	eys											
🏟 Gene	eral settings											
< Hide side	bar											
© 2022 The Th	nings Stack by The Things Networl	c and The Things Inc	dustries					 ⊕ en	v3.22.2 (3203f63b2)	Documentation	Status page	Get support

Figure 24: API key page

3. In the Name field, type the name of your key (for example - mykey). Choose Grant individual rights and select Link as Gateway to a Gateway for traffic exchange, i.e. read uplink and write downlink.

	uateways > Tak-euge-wis-gateway > ArTkeys > Au	
rak-edge-wis-gateway		
	Add API key	
Overview		
Live data	Name	
	тукеу	
Cocation	Expiry date	
Collaborators	mm/dd/yyyy	
	Rights *	
Ov API keys	 Grant all current and future rights 	
General settings	 Grant individual rights 	
	Select all	
	Delete gateway	
	View gateway information	
	☑ Link as Gateway to a Gateway Server for traffic exchange, i.e. write uplink and read downlink	
	View gateway location	
	Retrieve secrets associated with a gateway	
	View and edit gateway API keys	
	Edit basic gateway settings	
	View and edit gateway collaborators	
	View gateway status	
	Write downlink gateway traffic	
	Read gateway traffic	
	Store secrets for a gateway	
< Hide sidebar	Crisate API key	
© 2022 The Things Stack by The Things Networ	k and The Things Industries	In the second se

Figure 25: Generating an API key

4. To generate the key, choose **Create API key**. The following window will pop up, telling you to copy the key you just generated.

Your API key has been created successfully. Note: After closing this window, the value of the key secret will not be accessible anymore. Make sure to copy and store it in a safe place now. API key

Figure 26: Copying the generated key

WARNING:

```
Copy the key and save it in a .txt file (or other), because you won't be able to view or copy your key after that.
```

5. Click I have copied the key to proceed.

Configuring the Gateway

- 1. To configure the gateway, access it via the Web UI. To learn how to do that, refer to the Quick Start Guide for each gateway.
- 2. Navigate to LoRa > Configuration > Work mode and select Basics station.

WS Gate	RAK7289C	Gateways	
	• •		
± ∿ ≎	Work mode Log Level	Packet forwarder Basics station Built-in network server Log Level NOTICE	
	Basics station server setup	Configure Basics Station server setup.	~ •
	©2022 RANwireless Technology Limited. All Rights Reserved. WoGateOS 2.1		Privacy Policy · Terms and Conditions

Figure 27: Changing the working mode

3. Expand the Basics Station settings by clicking **Configure Basics Station server setup.**

WIS Gate	RAK7289C
	Overview Configuration Applications Gateways
	Work mode Packet forwarder Packet forwarder Basics station Built-in network server
*	Log Level Log Level NOTICE
	Balcs station server setup Configure Baacs Station server setup. Configure Baacs Station server setup. </th
	©2022 RAKwireless Technology Limited. All Rights Reserved. WisGeteOS 2.0 Privacy Policy - Terms and Conditions

Figure 28: Expanded Basics Station settings

- 4. To connect the gateway to TTNv3, configure the following parameters:
- Basics Station Server Type For server type, choose LNS Server.
- Server URL This is the link to The Things Stack server. Note that, for this tutorial, the gateway is connected to the European cluster. For Europe fill in the following:

wss://eu1.cloud.thethings.network

- Server Port The LNS Server uses port 8887. Type in 8887.
- Authentication Mode Choose TLS server authentication and Client token. When selected, the Trust (CA Certificate) and Client token fields will show up.
- Trust (CA Certificate) For trust, upload the Let's Encrypt ISRG ROOT X1 Trust certificate by clicking choose file. The file with the certificate can be downloaded directly □.
- Client Token This is the generated API key. The key must start with Authorization:.

For example:

Authorization: YOUR_API_KEY

NOTE:

Replace **YOUR_API_KEY** with the key generated previously. Have in mind that there should be a **space** between **Authorization:** and **YOUR_API_KEY**, as shown in the example.

Wis Gate	RAK7289C	
	Overview Configuration Applications Gateways	
₽ ∴ ^⁄	Work mode Packet forwa Basics statio Built-in netwo	der k server
*	Log Level Log Level NOTICE	•
	Basics station server setup Configure Basics Basics Station Server Configure Basics Configure	state server setup. wr Type server Port dthelhings network B867 ieiter Token Authentication Ieiter Token Authentication NIXXSX UPRUMBIKFH7327VMFEN55HTSQRDMIPFRQQT
	02022 RAKwizeless Technology Limited, All Rights Reserved, WisGeteOS 2.0	Privacy Policy - Terms and Conditions

Figure 29: Basics Station settings

THE THINOS STACK Community Edition	Uverview D Applications	🕳 Gateways 🛛 🗮 Or	rganizations		EU1 Community No support plan (2)	•
nak.edge.wic.gateway		Gateways > rak-edge-v	wis-gateway			
Takeoge-wis-gateway		nak oda				
Overview		ID: rak-edge-	e-wis-gateway wis-gateway			
Elive data		↑0 ↓0 • Last act	tivity 20 seconds ago 💿		🚜 1 Collaborator 🛛 🗣 1 API key	
Cocation		General information		• Live data	Son all activity	
Collaborators		Gateway ID	rak-edge-wis-gateway	↑ 15:49:13 Receive uplink message DevAddr:	: FPort: (
OT API keys		Gateway EUI		↓ 15:49:89 Send downlink message Tx Power: ↑ 15:49:87 Deceive unlink message JoinEUT;	: 16.15 Data rate: SF7BW125	
General settings		Gateway description	None	↑ 15:49:80 Receive uplink message JoinEUI:		
		Created at	Nov 17, 2022 15:39:07	↑ 15:48:53 Receive uplink message JoinEUI:		
		Last updated at	Nov 17, 2022 15:39:07	"F 15.46.49 Receive Saleway status versions	s. 2 5141101. 2.6.4-9-8305066	
		Gateway Server address	eul.cloud.thethings.network	Location	Change location settings \rightarrow	
		LoRaWAN information Frequency plan Global configuration	EU_863_870	No location information a	vallable	
< Hide sidebar						

Now, you can see that their gateway is connected to TTNv3 as Basics Station.

Figure 30: Successful connection

ChirpStack

This guide will show you how to connect the RAKWireless Commercial V2 gateway running WisGateOS 2 to a ChirpStack Network server, whether it is installed in the local or external network.

The guide is not about how to install the ChirpStack, but how to configure the gateway to send data to it.

Configuring the Edge Gateway V2 to ChirpStack

As mentioned before, the guide is for all RAK Edge V2 Series gateways running WisGateOS 2. It will be separated into two main sections based on where the ChirpStack Network server is installed:

- Local ChirpStack
- External ChirpStack

Different methods on how to connect the gateway to the server will be shown.

Local ChirpStack

In this case, the ChirpStack is installed in the local network. Three options will be considered here:

- Connecting the Gateway via Packet Forwarder
- Connecting the Gateway via MQTT Bridge
- · Connecting the Gateway via Basics Station

Each option is explained in a separate section.

Connecting the Gateway via Packet Forwarder

In this method, you will configure the gateway's packet forwarder to send data to the ChirpStack Gateway Bridge.

1. Start by accessing the gateway. To see how to access the gateway, refer to the Access the Gateway section.

Wis Gate	
Login root Pasword Login	Since connect, monitor, analyza and run gateways.
02022 RAKwireless Technology Limited. All Rights Reserved. WisGateOG 2.0	

Figure 31: Login page

2. Login using the set credentials you have set in the Access the Gateway section.

3. On the left side, head to **LoRa**. By default, the gateway is configured to work as a Built-in network server.

WsGate	RAK7289C	
::	Overview Configuration Applications Gateways	
.≟ .∿ ♥	Work mode Packet forwarder Basics station Built-in network server	
*	Log Level DEBUG ~	
	Frequency Plan Region EU868 -	
	View detailed regional parameters of the frequency plan.	~ Ø
	Network Server Parameters Network server parameters are used to configure general setup for your LoRa built-in server section is required for filling-in.	r. This 🗸 💿
	Gateway backend Configure the Gateway Backend to allow the central gateway and extenders to communicate MQTT.	e via 🗸 🖉
	Class B Settings Configure the beacon period and ping slots of class B devices to use time-sync beacons ser the gateways.	nt by 🗸 🖉
	Integration Interface Parameters Configure the Integration Interface to forward all received data to an external network serve	_{ir.} ~ ©
	Save changes	
	@2022 RAKwiteless Technology Limited. All Rights Reserved. WIsOateOS 2.0	Privacy Policy - Terms and Conditions

Figure 32: Network server settings

4. From **Work Mode**, select **Packet forwarder**. Click **Choose from the available protocols** to expand the Packet forwarder settings.

Wis Gate	RAK7289C	
	Overview Configuration Applications Gateways	
	Work mode Packet forwarder Basics station Built-in network server	
*	Log Level DEBUG ~	
	Frequency Plan EU868	
	View detailed regional parameters of the frequency plan.	v 0
	Protocol Choose from the available protocols. Protocol © Semtech UDP GWMP Protocol ChRa Gateway MQTT Bridge Statistic interval (s) 30	^ ◎
	UDP Protocol parameters Server address rak eu1.cloud thethings industries Server port up Server port down 1700 1700 Push timeout (ms) Keepative interval (s) 200 5 Auto-restart threshold 30	
	Class B Settings Configure the beacon period and ping slots of class B devices to use time- the gateways.	sync beacons sent by V
	GPS Information Add your GPS info manually.	~ 0
	Packet Filter Allows to optimize bandwidth by filtering and forwarding packets from cho	sen end devices.
	Save changes	
	@2022 RAKwireless Technology Limited. All Rights Reserved. WisGateOS 2.0	Privacy Policy - Terms and Conditions

Figure 33: Setting Packet Forwarder Mode

- 4. By default, when Packet Forwarder mode is chosen, the **Semtech UDP GWMP Protocol** is selected.
- 5. To point the gateway to the ChirpStack network using the packet forwarder, you only need to set the **Server address** of the ChirpStack.

In this case, the ChirpStack is installed locally on an Ubuntu machine on IP 192.168.0.130 (yours will be different). The other fields are filled with default parameters and can be left by default.

Protocol	Choose from the avail	able protocols.	
	Protocol		
	Semtech UDP GV	VMP Protocol	
	LoRa Gateway M	QTT Bridge	
	Statistic interval (s)		
	30		
UDP Protocol parameters	Server address		
	192.168.0.130		
	Server port up	Server port down	
	1700	1700	
	Push timeout (ms)	Keepalive interval (s)	
	200	5	
	Auto-restart threshold		

Figure 34: Configuring Packet Forwarder to ChirpStack

6. Click **Save changes** to save the changes.

Now you need to register the gateway in ChirpStack. The steps are the same for all options.

Registering the Gateway in ChirpStack Network Server

1. To register the gateway in the ChirpStack Network server, access the ChirpStack UI. To do that, open a web browser and type the server address of the ChirpStack with port 8080.

<IP address of ChirpStack>:8080

In this case, the ChirpStack is installed on a local Ubuntu machine with IP 192.168.0.130. The server address will be 192.168.0.130:8080.

ChirpStack Application Server × +			– o ×
← → C O A 192.168.0.130.8080/#/login		ជ	∞ IN © <u>5</u> =
	ChirpStack Login		
	Username / email *		
	Password *		
	LOGIN		

Figure 35: ChirpStack Login page

- 2. Login using the following credentials:
- Username/email: admin
- Password: admin

	ChirpStack				? e admin
^	Dashboard Network-servers	Dashboard			
R	Gateway-profiles	Active devices	Active gateways	Device data-rate usage	
	Organizations	No data available.	No data available.	No data available.	
-	All users				
٩	API keys	Gateways			
chirp	ostack 👻	No data available			
A	Org. dashboard				
*	Org. users				
٩	Org. API keys				
* =	Service-profiles				
프는	Device-profiles				
R	Gateways				
	Applications				

Figure 36: ChirpStack dashboard

3. Head to $\ensuremath{\textbf{Gateways}},$ on the left pane.

∉	ChirpStack				Q Sea	rch organization, application, gateway or device		9	admin
↑	Dashboard Network-servers	Gateways						+ cr	EATE
@ #1	Gateway-profiles Organizations	Last seen	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users					Rows per page: 10 👻	0-0 of 0	<	>
٩	API keys								
chirp	stack 👻								
ŧ	Org. dashboard								
•	Org. users								
٩	Org. API keys								
<u>≞</u> ≡	Service-profiles								
븊	Device-profiles								
R	Gateways								
	Applications								

Figure 37: Gateway list

- 4. By default, no gateways are registered. To register one, click + Create.
- 5. In the **General menu**, you need to set the gateway parameters.

€	ChirpStack	Q. Search organization, application, gateway or device 📀 😝 admin
ŧ	Dashboard	Cotourue / Croote
8 - 8 - 8 -	Network-servers	Sateways / Steate
\bigcirc	Gateway-profiles	GENERAL TAGS METADATA
	Organizations	Gateway name *
•	All users	WisGate_Edge_gateway
٩	API keys	nie name nay ony comani words, numera ano dasires. Gateway description *
chirp	ostack -	Connect to <u>ChirpStack</u>
♠	Org. dashboard	
•	Org. users	Gateway ID * AC 1F 09 FF FE
٩	Org. API keys	Network-server *
. ≡	Service-profiles	build_in_ns select the network-server to which the gateway will connect. When no network-servers are available in the dropdown, make sure a service-profile exists for this organization.
	Device-profiles	Service-profile
\bigcirc	Gateways	service-profile-build-in Select the service-profile under which the gateway must be added. The available service-profiles depend on the selected network-server, which must be selected first.
	Applications	Gateway-profile Select gateway-profile Optional. When assigning a gateway-profile to the gateway. ChirpStack Network Server will attempt to update the gateway according to the gateway-profile. Note that this does require a gateway with ChirpStack Concentratord.
		Gateway discovery enabled
		When enabled (and ChirpStack Network Server is configured with the gateway discover feature enabled), the gateway will send out periodical pings to test its coverage by other gateways in the same network.
		0 0
		When the gateway has an on-board GPS, this value will be set automatically when the network has received statistics from the gateway.
		0
		Leafer © OpenStreetNap contributors Drag the marker to the location of the gateway. When the gateway has an on-board GPS, this value will be set automatically when the network receives statistics from the gateway.
		ADD BOARD CONFIGURATION CREATE GATEWAY

Figure 38: Register the gateway

- **Gateway name** unique name for the gateway on the Network server. The name may only contain words, numbers, and dashes.
- Gateway description a brief description of the gateway.

- **Gateway ID** the Extended Unique Identifier (EUI) of the gateway. The EUI can be found, on the Overview page of the Dashboard menu of the web UI of the gateway.
- **Network-server** the network server to which the gateway will connect. When no network servers are available in the dropdown, make sure a service profile exists for this organization.
- Service profile the service profile under which the gateway must be added. The available service profiles depend on the selected network server, which must be selected first.
- **Gateway profile** this field is optional. When assigning a gateway profile to the gateway, ChirpStack Network Server will attempt to update the gateway according to the gateway profile. Note that this does require a gateway with ChirpStack Concentratord.
- Gateway discovery enabled When enabled (and ChirpStack Network Server is configured with the gateway
 discover feature enabled), the gateway will send out periodical pings to test its coverage by other gateways in
 the same network.
- **Gateway attitude** When the gateway has an onboard GPS, this value will be set automatically when the network has received statistics from the gateway.
- **Gateway location** you can drag the marker to the location of the gateway. When the gateway has an onboard GPS, this value will be set automatically when the network receives statistics from the gateway.
- 6. Once, everything is set, click **Create gateway** to register the gateway. You will see the registered gateway in the **Gateway list**.

∉	ChirpStack					Q Sear	ch organization, application, gateway or device	6	•	admin
^	Dashboard Network-servers	Ga	teways						+ CRE	EATE
@ #	Gateway-profiles Organizations		Last seen	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users API keys		Never	WisGate_Eoge_gateway	ac1109me	build_in_ns	Rows per page: 10 👻	1-1 of 1	<	>
chirp	ostack 👻									
↑	Org. dashboard Org. users									
م .≡	Org. API keys									
	Device-profiles									
®	Gateways Applications									

Figure 39: Registered gateway

7. If everything is set correctly, the Last seen status will state a few seconds ago in a while.

€	ChirpStack					Q Search organ	ization, application, gateway or device		2 6	admin
↑	Dashboard Network-servers	Gateways	5						+ 0	REATE
@ #	Gateway-profiles	Last seer	n	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users	a few sec	conds ago	WisGate_Edge_gateway	ac1f09fffe	build_in_ns				
٩	API keys						Rows per page: 10 👻	1-1 of 1	<	>
chirp	ostack 👻									
f	Org. dashboard									
•	Org. users									
٩	Org. API keys									
≞ ≡	Service-profiles									
1 1 1	Device-profiles									
\bigcirc	Gateways									
	Applications									

Figure 40: Connect the gateway

You can click the gateway name to inspect the gateway traffic.

€	ChirpStack				Q Search organization, application, gateway or device	? 🔒 admin
♠ ■ @	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY LIVE LORAWAN FRAMES			DELETE
•	All users	Gateway details		+		
chirp	estack 👻	Gateway ID Altitude	ac1f09fffe		0	
↑ ±	Org. dashboard Org. users	GPS coordinates	0,0		*	
۹.	Org. API keys Service-profiles	Last seen at	Apr 13, 2022 10:38 AM			
	Device-profiles				Leafer	OpenStreetMap contributors
	Applications	Received		Transmitted		
				10 03 04 07 05 04 05 04 03 02 01 0 0 05 04 05 04 05 04 05 04 05 05 05 05 05 05 05 05 05 05	ى ، ئە ^ى ى ، ئە ، ئە ، ئە ، ئە ، ئو بى بى جى بى بى بى بى بى بى بى .	· · · · · · · · · · · · · · · · · · ·



Connect the Gateway via MQTT Bridge

In this method, you will configure the gateway's built-in gateway bridge to send data to the ChirpStack Broker.

1. Start by accessing the gateway. To access the gateway check the Access the gateway.

WsGate	
Login ^{Login} root	
Password Ø	
	Simple access to your gateways Set up, connect, monitor, analyze and run gateways.
€2022 RAtiviteless Technology Limited. All Rights Reserved. WisQuteOS 2.0	

Figure 42: Login page

2. Login using the set credentials you have set in the **Access the gateway**.

3. On the left side, head to **LoRa**. By default, the gateway is configured to work as a **Built-in network server**.

WSGate	RAK7289C		
::	Overview Configuration Applicatio	ns Gateways	
.≞ .∿ ≎	Work mode	Packet forwarder Basics station Built-in network server	
*	Log Level	Log Level TEBUG	
	Frequency Plan	Region EU868 -	
		View detailed regional parameters of the frequency plan.	~ 0
	Network Server Parameters	Network server parameters are used to configure general setup for your LoRa built-in server. This section is required for filing-in.	~ 0
	Gateway backend	Configure the Gateway Backend to allow the central gateway and extenders to communicate via MOTT.	~ 0
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0
	Integration Interface Parameters	Configure the Integration Interface to forward all received data to an external network server.	~ 0
	@2022 RAVwireless Technology Limited. All Rights Reserved. WisGateO	520	Privacy Policy - Terms and Conditions

Figure 43: Network server settings

4. From **Work Mode**, select **Packet forwarder**. Click **Choose from the available protocols** to expand the Packet forwarder settings.

WsGate	RAK7289C		
::	Overview Configuration Applicat	tions Gateways	
.≟ .∿ ≎	Work mode	Packet forwarder Basics station Built-In network server	
*	Log Level	Log Level DEBUG	
	Frequency Plan	Region EU368	
		View detailed regional parameters of the frequency plan.	× 0
	Protocol	Choose from the available protocols. Protocol	∧ ©
		Semtech UDP GWMP Protocol LoRa Gateway MQTT Bridge Statistic interval (s) 30	
	UDP Protocol parameters	Server address rak.eu1.cloud thethings.industries Server port up Server port down 1700 1700 Push timeout (ms) Keepalive interval (s) 200 5	
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0
	GPS Information	Add your GPS info manually.	~ 0
	Packet Filter	Allows to optimize bandwidth by filtering and forwarding packets from chosen end devices.	~ 0
		Save changes	
	©2022 RAKwireless Technology Limited. All Rights Reserved. WIsGe	NOS 2.0	Privacy Policy - Terms and Conditions

Figure 44: Setting Packet Forwarder Mode

5. By default, when Packet Forwarder mode is chosen, the **Semtech UDP GWMP Protocol** is selected. To use the built-in gateway bridge, from the **Protocol** select **LoRa Gateway MQTT Bridge**.

Protocol	
	Choose from the available protocols.
	Protocol
	Semtech UDP GWMP Protocol
	LoRa Gateway MQTT Bridge
	Statistic interval (s)
	30
LoRa Gateway MQTT Bridge	MQTT Protocol
Parameters	MQTT for Build-in LoRa Network Server 👻
	MOTT Broker Address
	127.0.0.1
	MQTT Broker Port MQTT Version
	1883 v3.1 👻
	QoS Keepalive interval (s)
	1 - At Least Once 👻 10
	Clean session Retain
	Enable User Authentication
	SSL/TLS Mode
	None -

Figure 45: LoRa Gateway MQTT Bridge

The latest ChirpStack version supports both **MQTT for ChirpStack 3.x (JSON)** and **MQTT for ChirpStack 3.x** (**PROTOBUF**).

📝 NOTE

If you want to use JSON protocol, you need to change the payload marshaler in the gateway bridge .toml file to "JSON".

To configure the payload marshaler you need to access the configuration file of the gateway bridge. You can access it with the configuration file of the gateway bridge. You can also access it with an SSH connection. In this case, you will use the SSH client PuTTY to access the configuration files.

6. Access the ChirpStack via PuTTY. You need to type the address of the ChirpStack server in the **Host name** field and click **Open**. In this case on IP 192.168.0.130.

🕵 PuTTY Configuration		? ×	
Category: Session Logging Category: Session Logging Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Colours Connection Proxy Telnet Rlogin SSH Serial	Basic options for your PuTTY sess Specify the destination you want to connect Host Name (or IP address) 192.168.0.130 Connection type: Raw Telnet Raw Strings AWS_Martinskey AWS_VPN test Pi Zero aws aws_aws_cloud awstestagain Close window on exit: Always Never Only on clear	<pre>sion tto Port 22 O Serial Load Save Delete an exit</pre>	
About Help	Open	Cancel	

Figure 46: PuTTY client

7. In the /etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge.toml , find the **Integration** section and change the marshaler to JSON.



Figure 47: Payload Marshaler

8. Save and exit the file.

However, if you are using an earlier version of ChirpStack (V2), you will need to select MQTT for ChirpStack 2.
 x. The option MQTT for Embedded RAK Network Server is for a mesh network, where one gateway plays the role of a network server.

For this example, you will choose MQTT for ChirpStack 3. x (PROTOBUF).

IVIII Gate	Protocol	Choose from the available protocols.
		Protocol Semtech UDP GWMP Protocol C LoRa Gateway MQTT Bridge
↔ •		Statetic interval (s) 30
*	LoRa Gateway MQTT Bridge Parameters	MOTT Protocol MOTT for ChirpStack 3.x (Protobul) MOTT Groker Address MOTT External Content and Conte
		127.0.0.1 MGTT Broker Port MGTT Version 1883 v3.1
		QoS Koopalino interval (s) 1 - At Least Once + 10
		Clean session Retain Enable User Authentication
		SSL/TLS Mode None
		Lider Tope

Figure 48: MQTT for Chirpstack Protocol

10. By default, the built-in gateway bridge is pointed to the local Broker (127.0.0.1). To point the gateway to the ChirpStack network, you need to set the ChirpStack Broker address in the **MQTT Broker Address** field.

In this case, the ChirpStack is installed locally on an Ubuntu machine on IP 192.168.0.130 (yours will be different). The default port that the MQTT Broker uses is 1883.

LoRa Gateway MQTT Bridge	MQTT Protocol	
Parameters	MQTT for ChirpStack	3.x (Protobuf) 👻
	MQTT Broker Address	
	192.168.0.130	
	MQTT Broker Port	MQTT Version
	1883	v3.1 👻
	QoS	Keepalive interval (s)
	1 - At Least Once 👻	10
	Clean session	Retain
	Enable User Auth	nentication
	SSL/TLS Mode	
	None	•

Figure 49: Configuring Packet Forwarder to ChirpStack

- 11. Click **Save changes** to save the changes.
- 12. Now you need to register the gateway in the ChirpStack Network server. When Packet Forwarder mode is chosen, the **Semtech UDP GWMP Protocol** is selected by default. To register the gateway in ChirpStack, see **Registering gateway in ChirpStack Network server** section.
- 13. If everything is set correctly, the Last seen status will state a few seconds ago.

€	ChirpStack					Q Search	n organization, application, gateway or	device	6		admin
↑	Dashboard Network-servers	Gateways								+ c	REATE
@ 	Gateway-profiles	Last seen		Name	Gateway ID	Network server	Gateway activity (30d)				
	All users	a few second	s ago	WisGate_Edge_gateway	ac1f09fffe	build_in_ns					L
chirp	API keys						Rows per page:	10 👻	1-1 of 1	<	``````````````````````````````````````
A	Org. dashboard										
* «	Org. users Org. API keys										
1	Service-profiles										
₩ @	Device-profiles Gateways										
	Applications										

Figure 50: Registered gateway

You can click the gateway's name to inspect the gateway traffic.

∉	ChirpStack				Q Search organization, application, gateway or device	? e admin
↑ ■ <i>©</i>	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY LIVE LORAWAN FRAMES			DELETE
•	Organizations All users API keys	Gateway details		+ -		
chirp	org. dashboard Org. users Org. API keys Service-profiles Device-profiles	Gateway ID Altriude GPS coordinates Last seen at	ac 1109fffe 0 meters 0, 0 Apr 13, 2022 10:38 AM		9	e CoperStreetWag contributors
@ 	Gateways Applications	Received	<u><u><u></u></u></u>	Transmitted	ى ئى ئى تى تى ئى ئى ئى ئى ئى ئى ئى تى	0 0 0 0 0

Figure 51: Gateway details

Now your gateway is connected to the ChirpStack Network server.

Connecting the Gateway via Basics Station

1. In this method, you will connect the gateway to the **ChirpStack** via **Basics Station**. The **LoRa Basics™ Station** is an implementation of a LoRa packet forwarder.

Wis Gate	
Login ^{Login} root	
Password 🖉	
	Simple access to your gateways Set up, connect, monitor, analyze and run gateways.
Ø2022 RAKeinineless Technology Limited. AB Rights Reserved. WaStateO3 2.0	

Figure 52: Login page

2. Login using the set credentials you have set in the **Access the gateway**.

3. n the left side, head to LoRa. By default, the gateway is configured to work as a Built-in network server.

Wis Gate	RAK7289C		
::	Overview Configuration	Applications Gateways	
.≞ .∿ ≎	Work mode	Packet forwarder Basics station Built-in network server	
*	Log Level	Log Level DEBUG	
	Frequency Plan	EU868	
		View detailed regional parameters of the frequency plan.	~ 0
	Network Server Parameters	Network server parameters are used to configure general setup for your LoRa built-in server. This section is required for filling-in.	• O
	Gateway backend	Configure the Gateway Backend to allow the central gateway and extenders to communicate via MQTT.	~ 0
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0
	Integration Interface Parameters	Configure the Integration Interface to forward all received data to an external network server.	~ •
	@2022 RAXwinaless Technology Limited. All Rights	eserved. WsGateOS 2.0	Privacy Policy - Terms and Conditions

Figure 53: Network Server Settings

4. For **Work Mode**, select **Basics station** and click **Configure Basics Station** server setup to expand the Basics Station settings.

Wis Gate	R	AK7289C			
		Overview Configuration Applications Gate	eways		
	Lo Ba	Overview Configuration Applications Gate ork mode Pack Pack Basi rg Level Log Level DEBUC asics station server setup Configure Configure asics station server setup Configure Configure Muthentical None Server UR	eways cket forwarder sics station lit-in network server si J J G re Basics Station server setup. Station Server PIPS BOOT Server PIPS Server RL s cation Mode s Serve changes Serve changes	ever Pot	
	62022	RAKwireless Technology Limited. All Rights Reserved. WisGeteOS 2.0			Yivacy Policy - Terms and Conditions

Figure 54: Setting Basics Station mode

Here you need to point the gateway to the ChirpStack Network server:

- Server For server, choose LNS Server.
- URI the address of the ChirpStack server. In this case, the ChirpStack is installed locally on an Ubuntu machine on IP 192.168.0.130 (yours will be different). The URI will be ws://192.168.0.130.

NOTE

The URL starts with ws:// in case a plain text connection is used. Using the wss:// scheme will trigger a TLS connection based on the tc.{cert,key,trust} credentials set.

- Port the port to which the Websocket listens. The port is 3001.
- Authentication Mode authentication for the ChirpStack server. For this case, you will use no authentication.

Cate	RAK	7289C	Catavara		
	Overv	ew Configuration Applications	Gateways		
₩ ふ *	Work me Log Leve	ation server setup	Currenty 3 Packet forwarder Basics station Built-In network server Log Level DEBUG COnfigure Basics Station server setup. Basics Station Server Type CUPS Server CUPS Server CUPS Server ENL Server Server URL Ws://192.168.0.130 Authentication Mode None Server LNL Server LNL None Server LNL Server LNL None	Server Port	∧ 0
	©1012 RAXwind	ss Technology Limited. All Rights Reserved. WisGateOS 2	10		Privacy Policy - Terms and Conditions

Figure 55: Configuring Basics Station to ChirpStack

5. Click **Save changes** to save the changes.

Now your gateway is configured to work as Basics Station, and it is pointed to the ChirpStack gateway bridge. The default installation of the ChirpStack setups the backend configuration of the ChirpStack gateway bridge to semtech_udp .

6. To configure the backend of the ChirpStack gateway bridge, you need to access the configuration file of the bridge. To access it, you will need an SSH terminal. In this case, you will use a PuTTY client.

🕵 PuTTY Configuration		?	×
Putty Configuration Category: □ Logging □ Terminal □ Keyboard □ Bell □ Features □ Window □ Appearance □ Behaviour □ Translation □ Selection □ Colours □ Colours □ Proxy □ Telnet □ Rlogin □ SSH □ Serial	Basic options for your PuTTY set Specify the destination you want to connect Host Name (or IP address) 192.168.0.130 Connection type: Raw Telnet Rlogin SSH Load, save or delete a stored session Saved Sessions Default Settings AWS_Martinskey AWS_VPN test Pi Zero aws aws_cloud awstestagain Close window on exit: Always Never Only on cl	? ssion ct to Port 22 Conse Load Save Delet	rial
About Help	Open	Cance	əl

Figure 56: PuTTY client

Copy the configuration file's text in the ChirpStack Gateway bridge webpage 🖸 and place it in /etc/chirpstack-gateway-bridge.toml .

7. In the file, find the **Gateway backend configuration** paragraph and replace the type with basic_station.



Figure 57: Configure gateway bridge type

8. Now scroll down until you find the **Concentrator configuration** paragraph and uncomment the following text as shown below. Uncommenting the text, enables the configuration for the SX1301 concentrator chips.



Figure 58: Configure gateway bridge backend

9. Save and exit the .toml file and restart the gateway bridge service to apply the changes by restarting the gateway bridge service with the following command:



Now the ChirpStack backend configuration is set to basics station.

10. Then, you need to register the gateway in the ChirpStack Network server. To register the gateway in ChirpStack, see **Registering gateway in ChirpStack Network server** section.

If everything is set correctly, the **Last seen** status will state a few seconds ago. You can click the gateway name to inspect the gateway traffic.

Schementation Center

€	ChirpStack					Q Search orga	nization, application, gateway or device		? 6	admin
•	Dashboard Network-servers	G	ateways						+ (CREATE
® ₽	Gateway-profiles Organizations		Last seen	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users		a few seconds ago	WisGate_Edge_gateway	ac1f09fffe	build_in_ns				
٩	API keys						Rows per page: 10 👻	1-1 of 1	<	>
chir	pstack 👻									
ŧ	Org. dashboard									
•	Org. users									
٩	Org. API keys									
. ≣	Service-profiles									
	Device-profiles									
$\widehat{\mathbb{N}}$	Gateways									
	Applications									

Figure 59: Registered gateway

∉	ChirpStack				Q Search organization, application, gateway or device	? 🔒 admin
^	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY LIVE LORAWAN FRAMES			DELETE
•	Organizations All users API keys	Gateway details		* -		
chirp	istack 👻	Gateway ID Altitude	0 meters			
↑	Org. dashboard Org. users	GPS coordinates	0, 0 Aor 13, 2022 10:38 AM		8	
∝ .≓	Org. API keys Service-profiles					
₩ ®	Device-profiles Gateways				Leafer	OpenStreetMap contributors
	Applications	Received		Transmitted		
		. کمی کمی کمی کمی کمی کمی کمی کمی می کمی ک			^٢ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠ ٢٠	



Now your gateway is connected to the ChirpStack Network server.

External ChirpStack

In this case, the ChirpStack is installed on an external network. In the following example, the ChirpStack Network server is installed on the AWS cloud. A guide on how to install it on AWS can be found in the Knowledge Hub section.

To access the ChirpStack web UI, you need to enable **TCP port 8080** and to make the gateway to communicate with the Network server you need to enable the following ports in the inbound rules of the instance:

- The Semtech Packet Forwarder needs UDP port 1700.
- MQTT Bridge (unsecured) needs TCP port 1883.
- MQTT Bridge (secured) needs TCP port 8883.
- Basics Station needs TCP port 3001.

📝 NOTE

A guide on how to open the above ports can be found in the guide 🗹 on how to install ChirpStack on AWS.

Three options will be considered here:

- Connecting the Gateway via Packet Forwarder
- Connecting the Gateway via MQTT Bridge
- Connecting the Gateway via Basics Station

Each option is explained in its own separate section.

Connecting the Gateway via Packet Forwarder

In this method, you will configure the gateway's packet forwarder to send data to the ChirpStack Gateway Bridge.

NOTE

When connecting the gateway to the ChirpStack, you will need to open ports 1700 and 8080 to enable the communication between the gateway and the server and be able to access the ChirpStack.

sgr-02d2022a8cd0b1dc7	Custom TCP	•	тср	1700	Custom	•	Q	[Delete
							0.0.0.0/0 ×		
sgr-051383d50cb9fa500	Custom TCP	•	ТСР	8080	Custom	•	Q	[Delete
							0.0.0.0/0 ×		

Figure 61: Opened 1700 UDP port

1. Start by accessing the gateway.

Wis Gate	
Login Login Password Login	
02022 RAXwireless Technology Limited. All Rights Reserved	Simple access to your gateways Set up, connect, monitor, analyze and run gateways.

Figure 62: Login page

- 2. Login using the set credentials you have set in the Access the gateway.
- 3. On the left side, head to LoRa. By default, the gateway is configured to work as a Built-in network server.

Gate	RAK7289C		
	Overview Configuration Application:	s Gateways	
± ∧ o	Work mode	Packet forwarder Basics station Built-in network server	
*	Log Level	Log Level VEBUG V	
	Frequency Plan	Region EU868	
		View detailed regional parameters of the frequency plan.	~ 0
	Network Server Parameters	Network server parameters are used to configure general setup for your LoRa built-in server. This section is required for filling-in.	~ 0
	Gateway backend	Configure the Gateway Backend to allow the central gateway and extenders to communicate via MQTT.	 ♥
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ •
	Integration Interface Parameters	Configure the Integration Interface to forward all received data to an external network server.	~ 0
	@2022 RAKwireless Technology Limited. All Rights Reserved. WisGateOS	20	Privacy Policy - Terms and Conditions

Figure 63: Network server settings

4. From **Work Mode**, select **Packet forwarder**. Click **Choose from the available protocols** to expand the Packet forwarder settings.

Wis Gate		RAK7289C		
::		Overview Configuration Applications	Gateways	
≟ ∿ ≎		Work mode	Packet forwarder Basics station Built-in network server	
		Log Level	Log Level DEBUG	
		Frequency Plan	Region EU868 -	
			View detailed regional parameters of the frequency plan.	~ 0
		Protocol	Choose from the available protocols. Protocol Semtech UDP GWMP Protocol LoRa Gateway MQTT Bridge Statistic interval (s)	~ 0
		UDP Protocol parameters	30 Server address rak eu1.cloud thethings industries Server port up 1700 1700 Push timeout (ms) Keepalive interval (s) 200 5 Auto-restart threshold 30	
		Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0
		GPS Information	Add your GPS info manually.	~ 0
		Packet Filter	Allows to optimize bandwidth by filtering and forwarding packets from chosen end devices.	~ 0
			Save changes	
	٥	2022 RAKwireless Technology Limited. All Rights Reserved. WisGeteOS 2.0		Privacy Policy - Terms and Conditions

Figure 64: Setting packet forwarder settings

When Packet Forwarder mode is chosen, the Semtech UDP GWMP Protocol is selected by default.

To point the gateway to the ChirpStack network using the packet forwarder, you only need to set the When Packet Forwarder mode. The **Semtech UDP GWMP Protocol** is selected by default of the ChirpStack.

In this case, the ChirpStack is installed on the AWS cloud instance with public IP **18.156.176.220** (yours will be different). The default ports that the packet forwarder is using are 1700.

UDP Protocol parameters	Server address	Server address			
	18.156.176.220				
	Server port up	Server port down			
	1700	1700			
	Push timeout (ms)	Keepalive interval (s)			
	200	5			
	Auto-restart threshold				
	30				

Figure 65: Configure packet forwarder to ChirpStack

5. Click **Save changes** to save the changes.

Now you need to register the gateway in the ChirpStack Network server.

Registering the Gateway in ChirpStack

The steps for registering the gateway in ChirpStack are the same for all options.

1. To register the gateway in the ChirpStack Network server, access the ChirpStack UI. To do that, open a web browser and type the server address of the ChirpStack with port 8080.

<IP address of ChirpStack>:8080

2. In this case, the ChirpStack is installed on the AWS cloud with the public IP address 18.156.176.220.

ChirpStack Application Server × +		
← → C ▲ Not secure 18.156.176.220.8080/#/login		순 🏚 🛤 🖬 🔘 :
	ChirpStack Login	
	chinpstack Login	
	Username / email *	
	Password *	
	LOGIN	

Figure 66: ChirpStack login page

- 3. Login using the following credentials:
- Username/email: admin
- Password: admin

	ChirpStack		ঀ	Search organization, application, gateway or device	? 🔒 admin
^	Dashboard Network-servers	Dashboard			
\bigcirc	Gateway-profiles	Active devices	Active gateways	Device data-rate usage	
•	Organizations	No data available.	No data available.	No data available.	
٩	API keys	Gateways			
chirp	ostack 👻	No data available.			
f	Org. dashboard				
•	Org. users				
٩	Org. API keys				
* =	Service-profiles				
	Device-profiles				
\bigcirc	Gateways				
	Applications				

Figure 67: ChirpStack dashboard

4. On the left pane, head to **Gateways**.

	ChirpStack						? 🔒 admin
^	Dashboard	Gateways					+ CREATE
®	Gateway-profiles	Last seen	Name	Gateway ID	Network server	Gateway activity (30d)	
•	Organizations All users			-		Rows per page: 10 👻	0-0 of 0 < >
٩	API keys						
chirp	ostack 👻						
f	Org. dashboard						
<u>*</u>	Org. users						
٩	Org. API keys						
å≡	Service-profiles						
	Device-profiles						
R	Gateways						
	Applications						
_							

Figure 68: Gateway list

- 5. By default, no gateways are registered. To register one, click + Create.
- 6. In the **General** menu, you need to set the gateway parameters.

€	ChirpStack	Q Search organization, application, gateway or device ? \varTheta	admin
ŧ	Dashboard	Gateways / Create	
••	Network-servers		
\bigcirc	Gateway-profiles	GENERAL TAGS METADATA	
	Organizations	Galauze nama*	
•	All users	Valeney Iname WISGate_Edge_gateway	_
٩	API keys	The name may only contain words, numbers and dashes.	
chirr	octack -	Gateway description * Connect to <u>ChirpStack</u>	
Ching			
f	Org. dashboard	Gatavara (h.)	
•	Org. users	AC 1F 09 FF FE MSB	C
٩	Org. API keys	Network-server*	
<u>_</u> ≡	Service-profiles	Durg_in_ins Select the network-server to which the gateway will connect. When no network-servers are available in the dropdown, make sure a service-profile exists for this organization.	-
11	Device-profiles	Service-profile	
\bigcirc	Gateways	service-profile-build-in Select the service-profile under which the gateway must be added. The available service-profiles depend on the selected network-server, which must be selected first.	-
	Applications	Gateway-profile	
		Select gateway-profile Optional. When assigning a gateway-profile to the gateway, ChirpStack Network Server will attempt to update the gateway according to the gateway-profile. Note that this does require a gateway with ChirpStack Concentratord.	-
		Gateway discovery enabled	
		When enabled (and ChirpStack Network Server is configured with the gateway discover feature enabled), the gateway will send out periodical pings to test its coverage by other gateways in the same network.	
		Gateway altitude (meters) * 0	
		When the gateway has an on-board GPS, this value will be set automatically when the network has received statistics from the gateway.	
		Gateway location (set to current location)	- 1
		+	
		Q	
		Leafet © OpenStreeMap cont Drag the marker to the location of the gateway. When the gateway has an on-board GPS, this value will be set automatically when the network receives statistics from the gateway.	ributors
		ADD BOARD CONFIGURATION CREATE GATE	EWAY

Figure 69: Register the gateway

- **Gateway name** unique name for the gateway on the Network server. The name may only contain words, numbers, and dashes.
- Gateway description a brief description of the gateway.
- Gateway ID the Extended Unique Identifier (EUI) of the gateway. The EUI is in the Overview menu of the Dashboard page of the web UI of the gateway.
- **Network server** the network server to which the gateway will connect. When no network servers are available in the dropdown, make sure a service profile exists for this organization.
- **Service-profile** the service profile under which the gateway must be added. The available service profiles depend on the selected network server, which must be selected first.
- **Gateway profile** this field is optional. When assigning a gateway profile to the gateway, ChirpStack Network Server will attempt to update the gateway according to the gateway profile. Note that this does require a gateway with ChirpStack Concentrator.
- Gateway discovery enabled When enabled (and ChirpStack Network Server is configured with the gateway
 discover feature enabled), the gateway will send out periodical pings to test its coverage by other gateways in
 the same network.
- **Gateway attitude** When the gateway has an onboard, this value will be set automatically when the network has received statistics from the gateway.
- **Gateway location** you can drag the marker to the location of the gateway. When the gateway has an onboard GPS, this value will be set automatically when the network receives statistics from the gateway.
- 7. Once everything is set, click **Create gateway** to register the gateway. You will see the registered gateway in the Gateway list.

If everything is set correctly, the Last seen status in the ChirpStack will state a few seconds ago.

AK Documentation Center

€	ChirpStack						Q Search org	anization, application, gateway or device		9 6	admin
^	Dashboard Network-servers	Ga	iteways							+ 0	DREATE
®	Gateway-profiles		Last seen	Name	Gatew	ay ID	Network server	Gateway activity (30d)			
•	All users		a few seconds ago	WisGate_Edge_gateway	ac1f0	fffe	build_in_ns				T.
٩	API keys							Rows per page: 10 👻	1-1 of 1	<	>
chir	ostack -										
f	Org. dashboard										
•	Org. users										
٩	Org. API keys										
<u>*</u> =	Service-profiles										
	Device-profiles										
\bigcirc	Gateways										
	Applications										

Figure 70: Registered gateway

8. You can click the gateway name to inspect the gateway traffic.

∉	ChirpStack				Q Search organization, application, gateway or device	? 🕒 admin
↑ ■ <i>©</i>	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY LIVE LORAWAN FRAMES			DELETE
» •	Organizations All users API keys	Gateway details		-		
chirp A S C C C C C C C C C C C C C	org. dashboard Org. users Org. API keys Service-profiles	Gateway ID Altrude QPS coordinates Last seen at	ac 1109ffe 0 meters 0, 0 Apr 13, 2022 10:38 AM		Q	
# @	Gateways Applications	Received		Transmitted		



Now your gateway is connected to the ChirpStack Network server.

Connecting the Gateway via MQTT Bridge

In this method, you will configure the gateway's built-in bridge to publish the data to the ChirpStack MQTT broker.



1. Start by accessing the gateway.

WsGate	
Login roat Pasward Login	
	Simple access to your gateways Set up, connect, monitor, analyze and run gateways.

Figure 73: Login page

2. Login using the set credentials you have set in the Access the gateway.

3. On the left side, head to **LoRa**. By default, the gateway is configured to work as **Built-in network server**.

Wis Gate	RAK7289C			
	Overview Configuration Application	s Gateways		
.≟ ^∕	Work mode	Packet forwarder Basics station Built-In network server		
*	Log Level	Log Level DEBUG		
	Frequency Plan	Region EU868		
		View detailed regional parameters of the frequency plan.	~ 0	
	Network Server Parameters	Network server parameters are used to configure general setup for your LoRa built-in server. This section is required for filling-in.	~ 0	
	Gateway backend	Configure the Gateway Backend to allow the central gateway and extenders to communicate via MQTT.	× 0	
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0	
	Integration Interface Parameters	Configure the Integration Interface to forward all received data to an external network server.	× 0	
	@2022 RAKwireless Technology Limited, All Rights Reserved. WisGateOS	2.0	Privacy Policy · Ter	rms and Conditions

4. From **Work Mode**, select **Packet forwarder**. Click **Choose from the available protocols** to expand the Packet forwarder settings.

W8 Gate	RAK7289C		
	Overview Configuration Applications	Gateways	
.≟ .∿ ≎	Work mode	Packet forwarder Basics station Built-In network server	
*	Log Level	og Level TEBUG T	
	Frequency Plan	legion EUS68 -	
	v	Fiew detailed regional parameters of the frequency plan.	~ 0
	Protocol P S	Protocol Semtech UDP GWMP Protocol LoRa Gateway MQTT Bridge tatistic interval (s) 30	∧ ©
	UDP Protocol parameters	erver address rek eu1.cloud thethings industries erver port up 1700 1700 1700 1700 10 1700 10 10 10 10 10 10 10 10 10 10 10 10 1	
	Class B Settings	Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by he gateways.	~ 0
	GPS Information	Add your GPS info manually.	~ 0
	Packet Filter	Allows to optimize bandwidth by filtering and forwarding packets from chosen end devices.	~ 0
		Save changes	
	@2022 RAMvinless Technology Limited. All Rights Reserved. WisGateOS 2.0		Privacy Policy - Terms and Conditions

Figure 75: Setting packet forwarder mode

5. By default, when Packet Forwarder mode is chosen, the **Semtech UDP GWMP Protocol** is selected. To use the built-in gateway bridge, from the **Protocol** select **LoRa Gateway MQTT Bridge**.

Protocol					
	Choose from the available p	rotocols.			
	Protocol				
	Semtech UDP GWMP P	rotocol			
	LoRa Gateway MQTT B	ridge			
	Statistic interval (s)				
	30				
LoRa Gateway MQTT Bridge	MQTT Protocol				
Parameters	MQTT for Build-in LoRa N	letwork Server 👻			
	MQTT Broker Address				
	127.0.0.1				
	MQTT Broker Port	MQTT Version			
	1883	v3.1 -			
	QoS	Keepalive interval (s)			
	1 - At Least Once 👻	10			
	Clean session	Retain			
	Enable User Authen	tication			
	SSL/TLS Mode				
	None	~			

Figure 76: LoRa Gateway MQTT bridge

The latest ChirpStack version supports both **MQTT for ChirpStack 3.x (JSON)** and **MQTT for ChirpStack 3.x** (**PROTOBUF**).

📝 NOTE

If you want to use JSON protocol, you need to change the payload marshaler in the gateway bridge .toml file to json. By default, the marshaler is protobuf.

6. To configure the payload marshaler, you need to access the configuration file of the gateway bridge. You can access it with an SSH connection. In this case, you will use the SSH client PuTTY to access the configuration files. How to do this is explained in the Knowledge Hub 🖄 section.

🕵 PuTTY Configuration		?	×
Category: Session Logging Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Connection Data Proxy Telnet Rlogin SSH Serial About Help	Basic options for your PuTTY sess Specify the destination you want to connect Host Name (or IP address) ubuntu@ec2-18-156-176-220.eu-central- Connection type:	sion t to Port 22 O Se Load Save Delete	rial
, book Thop	open	Carlot	

Figure 77: PuTTY client

7. In the /etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge.toml , find the Integration section and change the marshaler to json .



Figure 78: Payload Masher

8. Save and exit the file.

However, if you are using an earlier version of ChirpStack (V2), you will need to select **MQTT for ChirpStack 2.x.** The option **MQTT for Embedded RAK Network Server** is for a mesh network, where one gateway plays the role of a network server. For this example, you will choose **MQTT for ChirpStack 3.x (PROTOBUF)**.

Santo	Protocol	Choose from the available protocols.
₩ ∴ *		LoRa Gateway MQTT Bridge Statistic interval (a) 30
*	LoRa Gateway MQTT Bridge Parameters	MOTT Protocol MOTT for ChirpStack 3.x (Protobuf) MOTT Boker Address
		127.0.0.1 MOTT Version 1883 V3.1 GoS Koppative interval (s)
		1 - At Least Once • 10 Clean session Retain
		SSL/TLS Mode None
		Linke Tree

Figure 79: MQTT for ChirpStack protocol

By default, the built-in gateway bridge is pointed to the local Broker (127.0.0.1). To point the gateway to the ChirpStack network, you need to set the ChirpStack Broker address in the **MQTT Broker Address** field.

In this case, the ChirpStack is installed on an AWS cloud instance with public IP 18.156.176.220 (yours will be different). The default port that the MQTT Broker uses is 1883.

LoRa Gateway MQTT Bridge	MQTT Protocol					
Parameters	MQTT for ChirpStack	MQTT for ChirpStack 3.x (Protobuf)				
	MQTT Broker Address					
	18.156.176.220					
	MQTT Broker Port	MQTT Version				
	1883	v3.1 👻				
	QoS	Keepalive interval (s)				
	1 - At Least Once 👻	10				
	Clean session	Retain				
	Enable User Auth	nentication				
	SSL/TLS Mode					
	None	•				

Figure 80: Configuring packet forwarder to ChirpStack

9. Click **Save changes** to save the changes.

If everything is set correctly, the **Last seen status** will state a few seconds ago. You can click the gateway name to inspect the gateway traffic.

€	ChirpStack				Q Search orga	nization, application, gateway or device		9 6	admin
↑	Dashboard Network-servers	Gateways						+ 0	REATE
@ #	Gateway-profiles Organizations	Last seen	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users	a few seconds ago	WisGate_Edge_gateway	ac1f09fffe	build_in_ns				11
٩	API keys					Rows per page: 10 👻	1-1 of 1	<	>
chirp	ostack 👻								
ŧ	Org. dashboard								
•	Org. users								
٩	Org. API keys								
å≡	Service-profiles								
	Device-profiles								
\bigcirc	Gateways								
	Applications								

Figure 81: Registered gateway

æ	ChirpStack				Q Search organization, application, gateway or device	? 🕒 admin
↑ ■ ©	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY LIVE LORAWAN FRAMES			DELETE
■ •	Organizations All users API keys	Gateway details		+ -		
chirp	ostack 👻	Gateway ID Altitude	0 meters			
♠ • •	Org. dashboard Org. users Org. API keys Service-profiles	GPS coordinates	0, 0 Apr 13, 2022 10:38 AM		\$ *	
₩ @ Ⅲ	Device-profiles Gateways Applications	Received		Transmitted	Laster j	OpenStreetMsp contributors
				10 05 07 06 05 04 03 02 01 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ى. ئى ئى ئى ئى ئى ئى ئى ئى ئى ² ى ئى ² ى ئى ئى ئى ئى ئى ² ى ئى	

Figure 82: Gateway details

Now your gateway is connected to the ChirpStack Network server.

Connecting the Gateway via Basics Station

In this method, you will connect the gateway to the ChirpStack via Basics Station. The LoRa Basics™ Station is an implementation of a LoRa packet forwarder.

VOTE					
When connecting the communication the communication the communication of	ng the gateway to t tion between the g	the Chirps ateway ai	Stack, you w nd the serve	vill need er and be	to open TCP ports3001 and 8080 to enable e able to access the ChirpStack.
sgr-051383d50cb9fa500	Custom TCP V	ТСР	8080	Custom 🔻	Q Delete
sgr-02cb9d4a6ca3b49e4	Custom TCP 🔹	TCP	3001	Custom 🔻	Q Delete

Figure 83: Enable 8080 and 3001 port

WIS Gate	
Login Login root Reservoird	
Ø2012 RAKiwietiess Technology Limited. All Rights Reserved. V	Simple access to your gateways Set up, connect, monitor, analyze and run gateways.

Figure 84: Login page

- 2. Login using the set credentials you have set in the **Access the gateway**.
- 3. On the left side, head to LoRa. By default, the gateway is configured to work as a Built-in network server.

Wis Gate	RAK7289C	
	Overview Configuration Applications Gateways	
.≞ .∿ ≎	Work mode Packet forwarder Basics station Built-in network server	
*	Log Level DEBUG ~	
	Frequency Plan Region EU868 -	
	View detailed regional parameters of the frequency plan.	~ 0
	Network Server Parameters Network server parameters are used to configure general setup for your LoRa built-in server. This section is required for filling-in.	~ 0
	Gateway backend Configure the Gateway Backend to allow the central gateway and extenders to communicate via MQTT.	~ 0
	Class B Settings Configure the beacon period and ping slots of class B devices to use time-sync beacons sent by the gateways.	~ 0
	Integration Interface Parameters Configure the Integration Interface to forward all received data to an external network server.	~ 0
	©2022 RAKwireless Technology Limited. All Rights Reserved. WisGateOS 2.0	Privacy Policy - Terms and Conditions

Figure 85: Network server settings

4. For **Work Mode**, select **Basics station** and click **Configure Basics Station** server setup to expand the Basics Station settings.

WB Gate	Overview Configuration Applications Gateways	
₫ ~ ¢	Work mode Packet forwarder Packet forwarder Basics station Built-in network server	
*	Log Level DEBUG *	
	Basics station server setup. CuPS-BOOT Server Server Server Cups-Boot Server Server Server	~ 0
	©2022 RAKelvelveless Technology Limited. All Rights Reserved. WbOolaOS 2.0	Privacy Policy - Terms and Conditions

Figure 86: Setting Basics Station mode

- Server For server, choose LNS Server.
- URI the address of the ChirpStack server. In this case, the ChirpStack is installed on an AWS cloud instance with public IP 18.156.176.220 (yours will be different). The URI will be: ws://18.156.176.220.

📝 NOTE

The URL starts with ws:// in case a plain text connection is used. Using the wss:// scheme will trigger a TLS connection based on the ``tc.{cert,key,trust} `credentials set.

- **Port** the port to which the Websocket listens. The port is 3001.
- Authentication Mode Authentication for the ChirpStack server. For this case, you will use no authentication.

Wis Gate	RAK7289C	Gateways		
± .∿ *	Work mode	Packet forwarder Basics station Built-in network server Log Level DEBUG		
	Basics station server setup	Configure Basics Station server setup. Basics Station Server Type CUPS-BOOT Server CUPS Server Ent Server Server URL Ws://192.168.0.130 Authentication Mode	Server Port	∧ 0
		None Save changes	•	
	D2022 RAKwireless Technology Limited. All Rights Reserved. WisQateOS 2.			Privacy Policy - Terms and Conditions

Figure 87: Configuring Basics Station to ChirpStack

5. Click **Save changes** to save the changes.

Now your gateway is configured to work as Basics Station and it is pointed to the ChirpStack gateway bridge. The default installation of the ChirpStack setups backend of the ChirpStack gateway bridge as semtech_udp .

To configure the backend of the ChirpStack gateway bridge, you need to access the configuration file of the bridge. To access it, you will need an SSH terminal. In this case, you will use the PuTTy client.

To access the ChirpStack configuration files, you need to access the instance. How to do this is explained in the Knowledge Hub 🖾 section.

🕵 PuTTY Configuration	? ×
Category:	Basic options for your PuTTY session Specify the destination you want to connect to Host Name (or IP address)
Keyboard Bell Features Window	ubuntu@ec2-18-156-176-220.eu-central· 22 Connection type: Raw Telnet Rlogin SSH Serial
···· Appearance ···· Behaviour ···· Translation ⊕·· Selection	Load, save or delete a stored session Saved Sessions awstestagain
Colours Connection Data Proxy	Default Settings AWS_Martinskey AWS_VPN test Pi Zero
Telnet Rlogin SSH Serial	aws aws_cloud awstestagain
Jenar	Close window on exit: Always Never Only on clean exit
About Help	Open Cancel

Figure 88: PuTTY client

- 6. In the ChirpStack Gateway bridge webpage, copy the configuration file text and place it in /etc/chirpstack-gateway-bridge.toml .
- 7. In the file, find the gateway backend configuration paragraph and replace the type with basic_station .

/etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge.toml	Modified	/etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge.toml	Modified
<pre># join_euis=[</pre>		<pre># join_euis=[</pre>	
<pre># ["00000000000000", "0000000000000ff"],</pre>		<pre># ["00000000000000", "0000000000000ff"],</pre>	
<pre># ["0000000000ff00", "0000000000ffff"],</pre>		<pre># ["0000000000ff00", "0000000000ffff"],</pre>	
#]			
join_euis=[join_euis=[
1			
# Gateway backend configuration.		Gateway backend configuration.	
[backend]		[backend]	
# Backend type.		# Backend type.	
		ŧ	
# Valid options are:		# Valid options are:	
<pre>* semtech_udp</pre>		<pre># * semtech_udp</pre>	
# * concentratord		<pre># * concentratord</pre>	
<pre># * basic_station</pre>		<pre># * basic_station</pre>	
type="semtech_udp"		type="basic_station"	
AG Get Help "O write Out "W where is "W Cut lext "O Justiny "C Cu	r Pos	A Get Help "O write Out "W where is "A Cut lext "J Justing "C (Jur Pos
A LAIL K REAG FILE A REPLACE TO Paste Text I To Spell Go	o io Line	A LAIL A READ FILE A REPLACE TO Paste Text I To Spell T	o to Line

Figure 89: Configure gateway bridge type

8. Now scroll down until you find the **Concentrator configuration** paragraph and uncomment the following text as shown below.



Figure 90: Configuring gateway bridge backend

9. Save and exit the .toml file and restart the gateway bridge service to apply the changes by restarting the gateway bridge service with the following command:

sudo	syst	emct	:l re	start	chir	pstac	k-ga1	teway	∕-br:	idge	.ser	vice	ę					
	<u>.</u>								_									

Now the ChirpStack backend configuration is set to Basics station.

If everything is set correctly, the Last seen status will state a few seconds ago. You can click the gateway name to inspect the gateway traffic.

	ChirpStack						•	• •	admin
^	Dashboard Network-servers	Gateways						+ cr	REATE
®	Gateway-profiles Organizations	Last seen	Name	Gateway ID	Network server	Gateway activity (30d)			
•	All users	a few seconds ago	WisGate_Edge_gateway	ac1f09fffe	build_in_ns				1
٩	API keys					Rows per page: 10 👻	1-1 of 1	<	>
chirp	ostack 👻								
f	Org. dashboard								
•	Org. users								
٩	Org. API keys								
<u>*</u> ≡	Service-profiles								
	Device-profiles								
\bigcirc	Gateways								
	Applications								

Figure 91: Registered gateway

€	ChirpStack					Q Search organization, application, gateway or device	? 🖯 admin
* 	Dashboard Network-servers Gateway-profiles	Gateways / WisGate_Edge_gateway	CERTIFICATE GATEWAY DISCOVERY	LIVE LORAWAN FRAMES			DELETE
	All users	Gateway details			<u>+</u> -		
chirp	ostack - Org. dashboard	Gateway ID Altitude	ac1f09fffe 0 meters			9	
•1 •	Org. users Org. API keys	Last seen at	Apr 13, 2022 10:38 AM				
The Diff	Service-profiles Device-profiles					Let	aflet @ OpenStreetMap contributors
	Applications	Received	· · · · · · · · · · · · · · · · · · ·		Transmitted		

Figure 92: Gateway details

Now your gateway is connected to the ChirpStack Network server.

ThingPark by Actility

In this section, you will learn how to add RAK7289V2/RAK7289CV2 WisGate Edge Pro in ThingPark.

ThingPark is Actility's platform, in which you can register your LoRaWAN gateway and end devices. ThingPark offers a user-friendly dashboard, in which you can monitor different information about the gateway/end device like status, radio traffic, statistics, and more. Together with HTTPS integration, you can send the data received from the end nodes to an Application server for post-processing and visualizing.

For the complete step-by-step tutorial, refer to the How to Add RAK WisGate Edge Gateway V2 in ThingPark - Actility ThingPark Guide

Last Updated: 9/19/2023, 2:53:54 AM