

**Universidad Nacional de La Plata**

**Facultad de Informática**

**ACTIVIDAD 2 – AWS IoT**

**Cloud Computing y Cloud Robotics**

**Deluca, Tadeo**

**Domé, Valentín**

**6 de Octubre de 2023**

# Índice

Índice.....	2
Sección 1: Introducción.....	3
Sección 2: Lanzamiento dos instancias EC2 en AWS.....	4
Sección 3: Conexión a la Instancia EC2 Utilizando PuTTY en Windows.....	7
Sección 4: Instalación de Node-RED en ambas instancias.....	10
Sección 5: Instalación del nodo “node-red-dashboard”.....	14
Sección 6: Creación de objetos AWS IoT.....	16
Sección 7: Edición de políticas de objetos IoT.....	20
Sección 8: Configuración de nodo MQTT.....	23
Sección 9: Configuración del dashboard en el objeto A.....	29
Sección 10: Configuración del dashboard en el objeto B.....	39
Sección 11: Validaciones de funcionamiento.....	43
Sección 12: Terminación de instancias.....	44
Sección 13: Conclusión.....	44
Sección 14: Fuentes.....	45

## **Sección 1: Introducción**

En este informe, se detallarán los pasos seguidos para llevar a cabo la actividad. El objetivo principal de esta actividad es la implementación de un sistema de control remoto de iluminación, donde la Instancia A actúa como controlador y la Instancia B como el dispositivo controlado.

Para lograr este propósito, se ha empleado la plataforma Node-RED, una herramienta versátil y poderosa que facilita la creación de aplicaciones de Internet de las Cosas (IoT) de manera visual y sencilla. Además, se ha utilizado el nodo "node-red-dashboard" para diseñar interfaces gráficas que permiten el control y la supervisión de la iluminación en tiempo real.

La conectividad entre las instancias se ha logrado a través del protocolo MQTT (Message Queuing Telemetry Transport), utilizando el servicio AWS IoT Core como broker de mensajes. Este enfoque de comunicación en tiempo real permite que la Instancia A y la Instancia B intercambien información y controlen el estado de la iluminación de manera eficiente y confiable.

El informe detallará paso a paso cómo se configuraron las instancias, se instaló Node-RED y el nodo "node-red-dashboard", y se estableció la comunicación MQTT mediante AWS IoT Core.

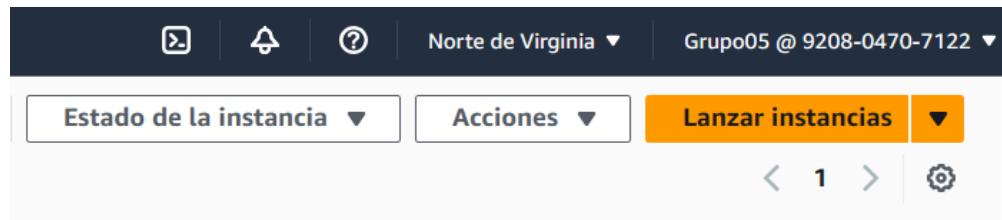
En el marco de esta actividad, se ha desarrollado una implementación parcial de la función del "TIMER de 5s". Aunque se ha creado un toggle en la instancia A para activar esta función y transmitir su estado a la instancia B, lamentablemente, no se ha logrado completar la funcionalidad de apagado automático del LED después de 5 segundos. Este inconveniente se debe a limitaciones de tiempo durante la realización de la actividad en clase.

A pesar de esta limitación, el sistema permite el control manual del LED y proporciona una representación precisa del estado del LED y del temporizador en la instancia B.

## Sección 2: Lanzamiento dos instancias EC2 en AWS

Para comenzar con la actividad, el primer paso es lanzar una instancia EC2 en AWS siguiendo estos pasos para la instancia A y repitiendo los mismos pasos para la instancia B:

1. Inicie sesión en su cuenta de AWS y navegue a la página de EC2 en la consola de AWS, haga clic en "Instancias" en el panel de navegación de la izquierda, luego, en la parte superior derecha, haga clic en el botón "Lanzar instancia":



Botón Lanzar instancias en AWS console

2. Se abrirá el asistente de creación de instancia EC2. Ingresamos el nombre de la instancia que en nuestro caso es “**grupo5-A**” para la primera instancia y “**grupo5-B**” para la segunda, luego en la sección “Inicio rápido”, seleccione el sistema operativo “Debian” y, en la lista de imágenes de máquina Amazon (AMI), elija la opción correspondiente a “Debian 12”, además configure la arquitectura en “64 bits” tal como se muestra en la siguiente imagen:

**Inicio rápido**

Ubuntu Windows Red Hat SUSE Linux Debian

Buscador de AMI

Inclusión de AMI de AWS, Marketplace y la comunidad

**Amazon Machine Image (AMI)**

**Debian 12 (HVM), SSD Volume Type**  
ami-06db4d78cb1d3bbf9 (64 bits (x86)) / ami-0d3eda47adff5e44b (64 bits (Arm))  
Virtualización: hvm Habilitado para ENA: true Tipo de dispositivo raíz: ebs

Apto para la capa gratuita

**Descripción**  
Debian 12 (20230711-1438)

**Arquitectura** 64 bits (x86) **ID de AMI** ami-06db4d78cb1d3bbf9 **Proveedor verificado**

#### Selección del SO y la Arquitectura

- En la sección "Par de claves" para el inicio de sesión en la consola vía SSH, seleccione la clave .pem previamente generada, en nuestro caso nombrada como "grupo05"

**Par de claves (Inicio de sesión)**

Puede utilizar un par de claves para conectarse de forma segura a la instancia. Asegúrese de que tiene acceso al par de claves seleccionado antes de lanzar la instancia.

**Nombre del par de claves - obligatorio**

grupo05

Crear un nuevo par de claves

- Luego, en la sección "Tipo de instancia", elija "t2.micro" la cual pertenece a la capa gratuita:

**Tipo de instancia** [Información](#)

**t2.micro** Apto para la capa gratuita

Familia: t2 1 vCPU 1 GiB Memoria Generación actual: true  
 Bajo demanda Windows base precios: 0.0162 USD por hora  
 Bajo demanda SUSE base precios: 0.0116 USD por hora  
 Bajo demanda RHEL base precios: 0.0716 USD por hora  
 Bajo demanda Linux base precios: 0.0116 USD por hora

Todas las generaciones

[Comparar tipos de instancias](#)

hell Comentarios Idioma

Selección de tipo de instancia

- En el apartado "Configuración de red", utilice la VPC proporcionada por la cátedra y, en la sección "Firewall", seleccione la opción "Seleccionar un grupo de seguridad existente":

Subred [Información](#)  
 Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

Asignar automáticamente la IP pública [Información](#)

Habilitar

**Firewall (grupos de seguridad)** [Información](#)  
 Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

Crear grupo de seguridad  Seleccionar un grupo de seguridad existente

Grupos de seguridad comunes [Información](#)  
 Seleccionar grupos de seguridad

sggrupo05 sg-0ec0328da94086b14 X  
 VPC: vpc-0157552403fa567fd

Compare reglas de grupo de seguridad

Configuración del grupo de seguridad

- Deje el resto de la configuración de la instancia en los valores predeterminados y haga clic en el botón "Lanzar instancia".

Una vez haya terminado de configurar y lanzar las dos instancias deberían poder verse en el apartado de instancias en el estado “En ejecución”, tal y como se muestra en la siguiente imagen:

Name	ID de la instancia	Estado de la i...	Tipo de inst...	Comprobación ...	Estado de la ...	Zona de dispon...
grupo5-A	i-0392c15cc13d89f0	En ejecución	t2.micro	Inicializando	Sin alarmas	us-east-1b
Grupo06instanciaA	i-06d32ff9ff4aced145	En ejecución	t2.micro	2/2 comprobacion	Sin alarmas	us-east-1b
grupo5-B	i-0ac0fea0b56f83a79	En ejecución	t2.micro	-	Sin alarmas	us-east-1b
Grupo06instanciaB	i-027543724cfdead7f9	En ejecución	t2.micro	Inicializando	Sin alarmas	us-east-1b

Instancias grupo5-A y grupo5-B en ejecución

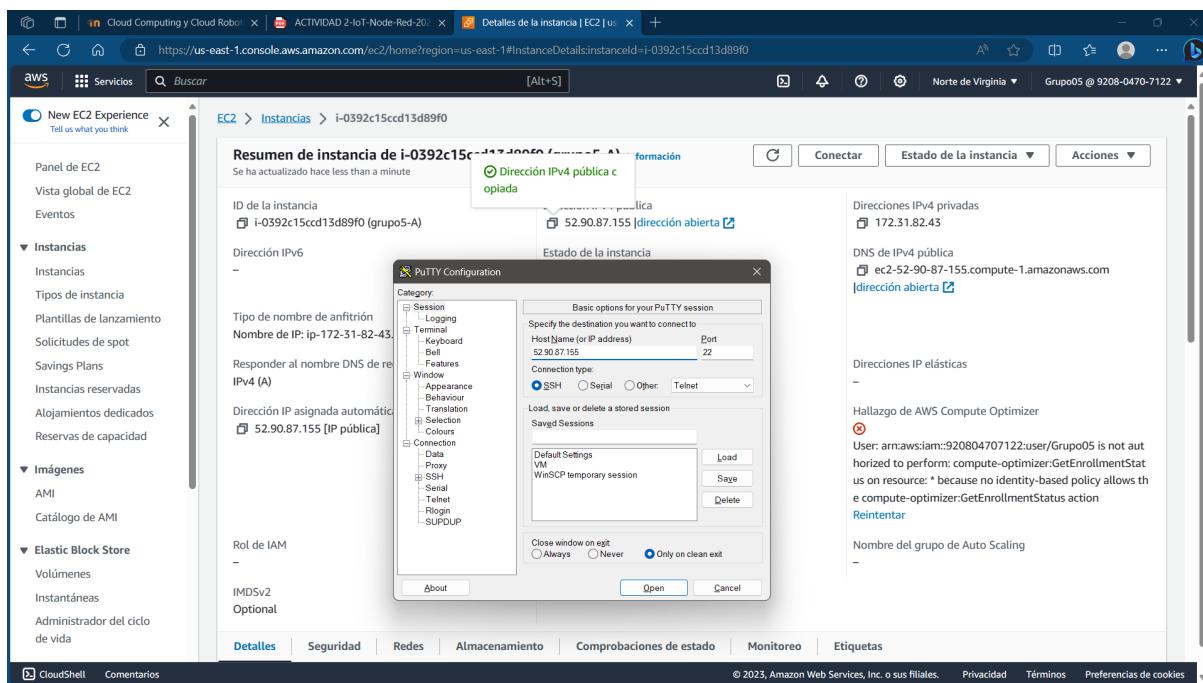
## Sección 3: Conexión a la Instancia EC2 Utilizando PuTTY en Windows

En esta sección, explicaremos cómo conectarte a la instancia EC2 utilizando PuTTY en un sistema operativo Windows ya que de este disponemos. Teniendo en cuenta que estos pasos fueron descritos con anterioridad en la actividad 1 solo desarrollaremos los pasos de forma general.

A continuación, describiremos los pasos que se siguieron para lograr esta conexión.

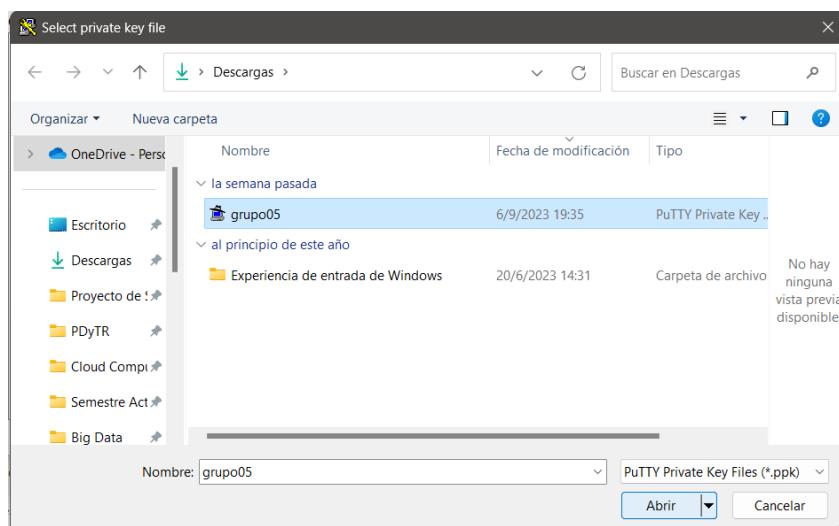
### Paso 1: Conexión a la Instancia Utilizando PuTTY

1. Suponiendo que previamente se generó el archivo .pem necesario para acceder mediante el programa PuTTY, abre PuTTY y en el campo "Host name (or IP address)", ingresa la dirección IP pública o el nombre de host de la instancia EC2 que se encuentra en el resumen de la instancia generada. Mantén el número de puerto en 22, que es el puerto SSH por defecto.

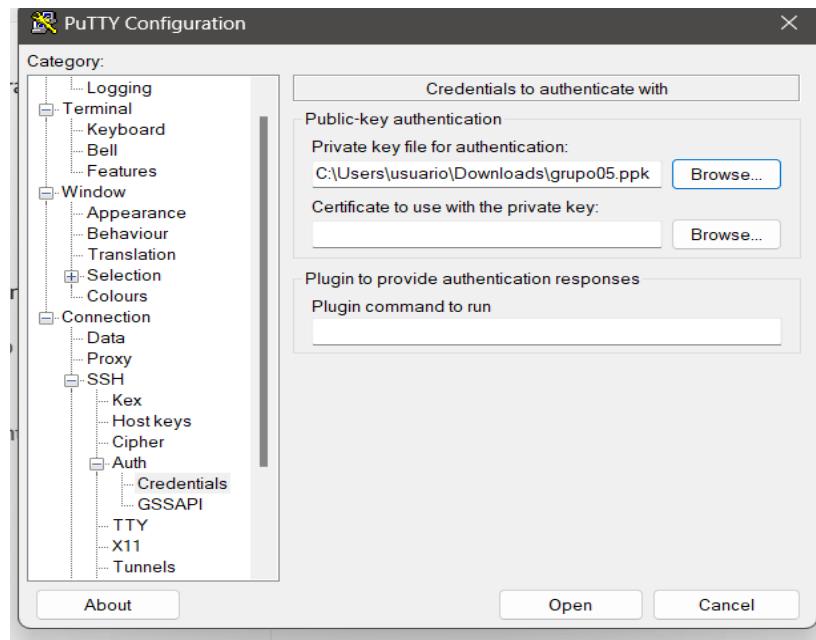


Ventana inicial de PuTTY y dirección IPv4 pública de la instancia grupo5-A

2. Luego, en la sección **Connection > SSH > Auth**, en el campo **Private key file for authentication**, selecciona la ubicación del archivo .ppk de tu clave privada

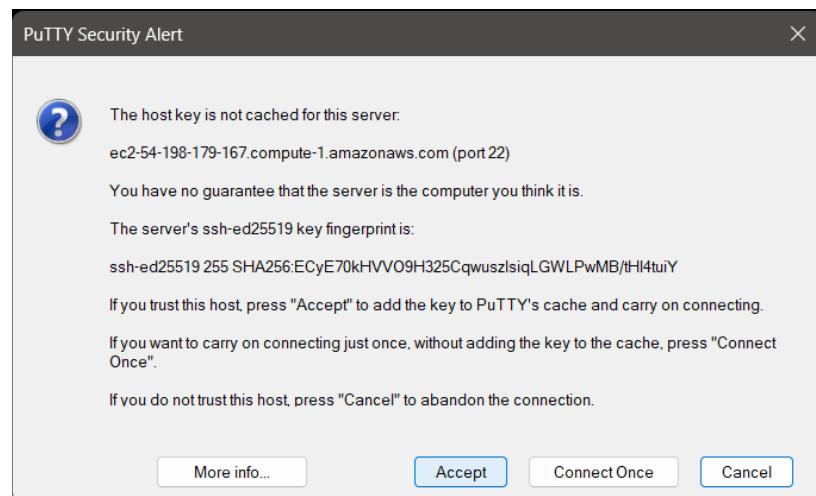


Búsqueda del archivo .ppk



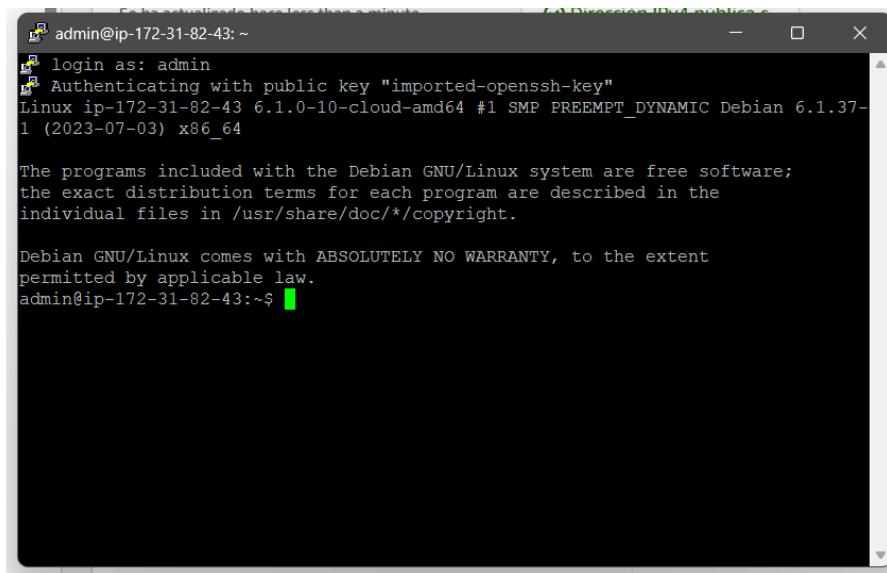
Selección de la clave ppk

3. Haz clic en el botón "Open".
4. Aparecerá una ventana emergente de seguridad de PuTTY indicando que la host key no está en la caché. Haz clic en el botón "Accept" para continuar:



Mensaje de seguridad

- Finalmente, se abrirá una ventana de línea de comandos donde deberás ingresar el nombre de usuario de la instancia EC2, que es "admin". Una vez que lo ingreses, habrás establecido una conexión SSH exitosa.



```
admin@ip-172-31-82-43: ~
login as: admin
Authenticating with public key "imported-openssh-key"
Linux ip-172-31-82-43 6.1.0-10-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.37-1 (2023-07-03) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin@ip-172-31-82-43:~$
```

Consola corriendo desde de SSH

## Sección 4: Instalación de Node-RED en ambas instancias

Node-RED es un entorno de programación visual basado en JavaScript que facilita la creación de aplicaciones IoT (Internet de las cosas) y flujos de trabajo automatizados. Esta plataforma de código abierto es ampliamente utilizada en proyectos de automatización, control de dispositivos y recopilación de datos.

Para instalar Node-RED en ambas instancias, siga los siguientes pasos detallados a continuación:

### Paso 1: Actualización de los Paquetes

Comienza actualizando los paquetes en la instancia con el siguiente comando:

```
$ sudo apt-get update
```

## **Paso 2: Instalación de Snapd**

Snapd es una herramienta de administración de paquetes y un sistema de sandboxing que permite la instalación y gestión de aplicaciones de forma segura en sistemas Linux. Este sistema de empaquetado ofrece una forma confiable de distribuir software, proporcionando aislamiento de aplicaciones para garantizar la seguridad del sistema.

Para poder instalar la herramienta ejecuta en la terminal el siguiente comando:

```
$ sudo apt install snapd
```

## **Paso 3: Instalación del núcleo con Snapd**

El paquete "core" de Snapd proporciona las bibliotecas y componentes esenciales necesarios para que las aplicaciones Snap se ejecuten y funcionen de manera aislada y segura en el sistema operativo.

Para poder instalarlo ingresa el siguiente comando:

```
$ sudo snap install core
```

## **Paso 4: Instalación del Node-RED**

Finalmente para poder instalar Node-RED ejecute en su terminal el siguiente comando:

```
$ sudo snap install node-red
```

## **Paso 5: Verificación de instalación de Node-RED**

Para poder comprobar que se instaló correctamente Node-RED verifica que el puerto 1880 está en estado de “LISTEN”, indicando que el servicio de Node-RED está activo en ese puerto.

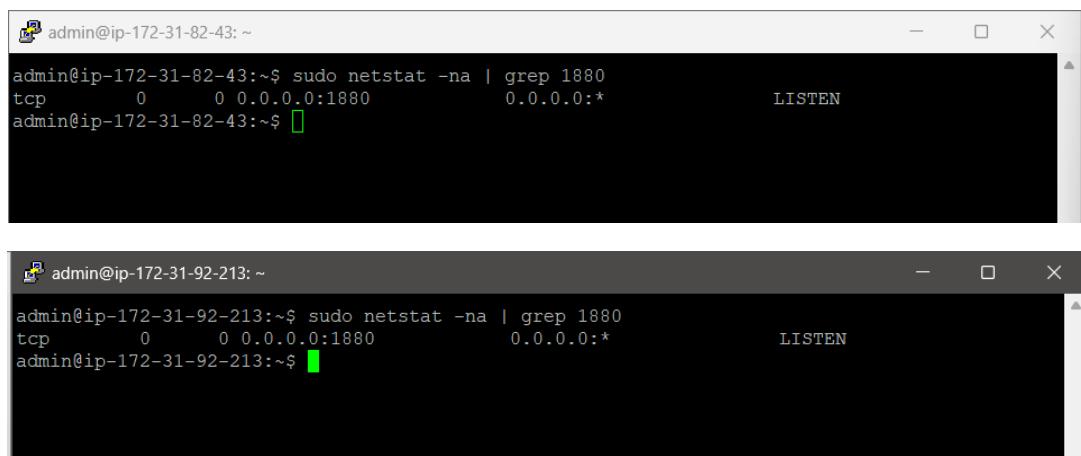
Primero instala net-tools, que es un conjunto de herramientas para trabajar con redes desde consola, con el siguiente comando:

```
$ sudo apt-get install net-tools
```

Acto seguido ejecuta el siguiente comando:

```
$ sudo netstat -na | grep 1880
```

Debería obtener una salida similar a la siguiente para ambas instancias:



The image shows two separate terminal windows. Both windows have a title bar with the user 'admin' and IP address. The first window's title is 'admin@ip-172-31-82-43: ~'. The second window's title is 'admin@ip-172-31-92-213: ~'. Both windows contain the same command output: 'sudo netstat -na | grep 1880'. The output shows a single line: 'tcp 0 0 0.0.0.0:1880 0.0.0.0:\* LISTEN'. The windows are side-by-side, indicating two different physical or virtual machines.

```
admin@ip-172-31-82-43:~$ sudo netstat -na | grep 1880
tcp        0      0 0.0.0.0:1880          0.0.0.0:*          LISTEN
admin@ip-172-31-82-43:~$ [green highlight]

admin@ip-172-31-92-213:~$ sudo netstat -na | grep 1880
tcp        0      0 0.0.0.0:1880          0.0.0.0:*          LISTEN
admin@ip-172-31-92-213:~$ [green highlight]
```

Servicio de Node-RED escuchando en el puerto 1880 para las instancias A y B

## Paso 6: Configuración de las reglas de entrada para el puerto 1880.

Para poder acceder desde tu navegador al servicio de Node-RED en tu instancia debes configurar las reglas de entradas de las mismas para permitir el tráfico TCP a través del puerto 1880 a las mismas, para esto debe para ambas instancias:

1. Diríjase a la sección de instancias, seleccione su instancia y en el resumen de la instancia, ingrese al apartado “Seguridad”:

The screenshot shows the AWS EC2 instance summary page. On the left, there's a sidebar with options like 'Panel de EC2', 'Vista global de EC2', 'Eventos', and 'Instancias'. Under 'Instancias', there are sub-options: 'Instancias', 'Tipos de instancia', 'Plantillas de lanzamiento', 'Solicitudes de spot', 'Savings Plans', and 'Instancias reservadas'. The main content area has tabs at the top: 'Detalles', 'Seguridad' (which is selected), 'Redes', 'Almacenamiento', 'Comprobaciones de estado', 'Monitoreo', and 'Etiquetas'. Under 'Seguridad', it shows 'Detalles de seguridad' with 'Rol de IAM' set to '-' and 'ID del propietario' as 920804707122. It also lists 'Grupos de seguridad' with 'sg-0ec0328da94086b14 (sggrupo05)' selected. Below that is the 'Reglas de entrada' section, which is currently empty.

Apartado de seguridad en el resumen de la instancia

- En este apartado haga clic en el botón “Editar Reglas de Entrada” y agregue la siguiente regla:

Nombre	ID de la regla del grupo d...	Intervalo de pu...	Protocolo	Origen	Grupos de seg...
-	sgr-0fddc6e34f89fa1a2	1880	TCP	0.0.0.0/0	sggrupo05

Regla de entrada para permitir tráfico TCP al puerto 1880

- Una vez hecho esto haga clic en el botón “Agregar regla de entrada” y obtendrá una ventana similar a esta:

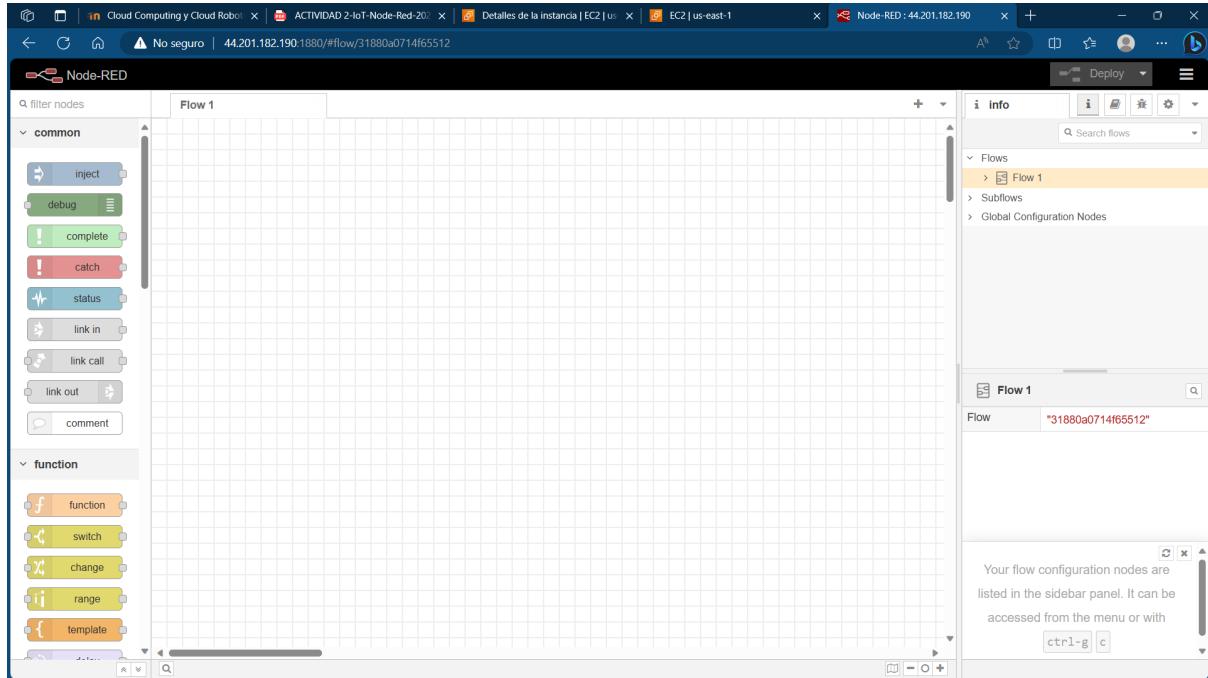
The screenshot shows the AWS Security Groups details page for the security group sg-0ec0328da94086b14 - sggrupo05. At the top, there's a green banner stating: "Las reglas del grupo de seguridad de entrada se han modificado correctamente en el grupo de seguridad (sg-0ec0328da94086b14 | sggrupo05)". Below that, there's a breadcrumb navigation: EC2 > Grupos de seguridad > sg-0ec0328da94086b14 - sggrupo05. On the right, there's an "Acciones" button. The main content area has a "Detalles" section with fields: Nombre del grupo de seguridad (sggrupo05), ID del grupo de seguridad (sg-0ec0328da94086b14), Descripción (launch-wizard-2 created 2023-09-06T22:13:54.33Z), and ID de la VPC (vpc-0157552403fa567fd). Below that is a "Reglas de entrada" section with a table showing four rules. The table columns are: Name, ID de la regla del g..., Versión de IP, Tipo, Protocolo, and Intervalo de puerto. The first rule is for port 1880 with TCP protocol.

Name	ID de la regla del g...	Versión de IP	Tipo	Protocolo	Intervalo de puerto
-	sgr-0fddc6e34f89fa1a2	IPv4	TCP personalizado	TCP	1880
-	sgr-0fddc6e34f89fa1a2	IPv4	TCP personalizado	TCP	1880
-	sgr-0fddc6e34f89fa1a2	IPv4	TCP personalizado	TCP	1880
-	sgr-0fddc6e34f89fa1a2	IPv4	TCP personalizado	TCP	1880

Mensaje de éxito de la edición de las reglas de entrada

## Paso 6: Acceso desde una navegador

Por último, accede desde tu navegador web a la dirección IPv4 pública de la instancia configurada en el puerto 1880, de esta forma debería poder obtener un resultado similar a este:



Servicio de Node-RED funcionando

## Sección 5: Instalación del nodo “node-red-dashboard”

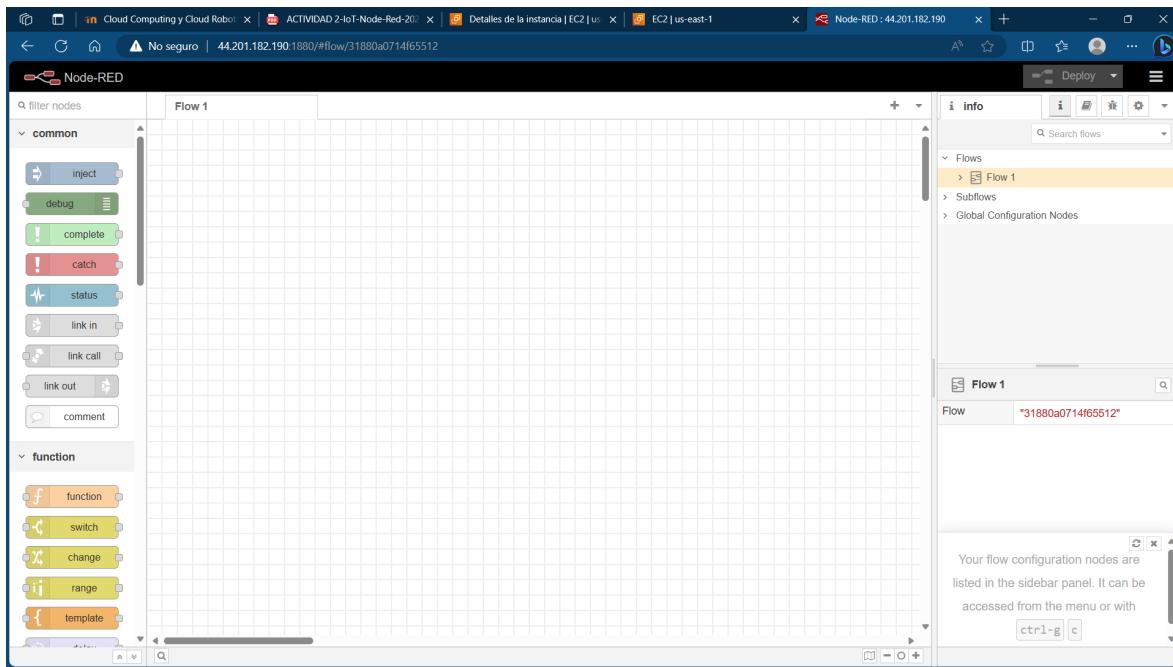
### ¿Qué es el nodo node-red-dashboard?

Este nodo, propio de Node-RED, proporciona una interfaz de usuario gráfica que permite crear paneles de control personalizados para controlar y monitorear dispositivos y datos en tiempo real. "node-red-dashboard", le permite a los usuarios diseñar fácilmente paneles de control web interactivos sin necesidad de conocimientos de programación web. Esto hace que la visualización y el control de datos IoT sean accesibles y amigables para una variedad de aplicaciones y proyectos.

Para poder instalar node-red-dashboard, se siguieron los siguientes pasos:

### Paso 1: Conéctese desde su navegador a Node-RED.

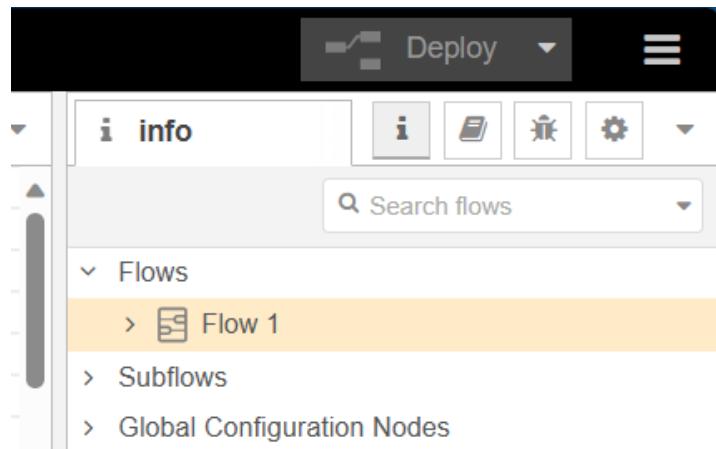
Tal y como se mostró en la sección anterior, conéctese al servicio de Node-RED de la instancia creada desde tu navegador web:



Servicio de Node-RED funcionando

## Paso 2: Abrir el menú de Node-RED.

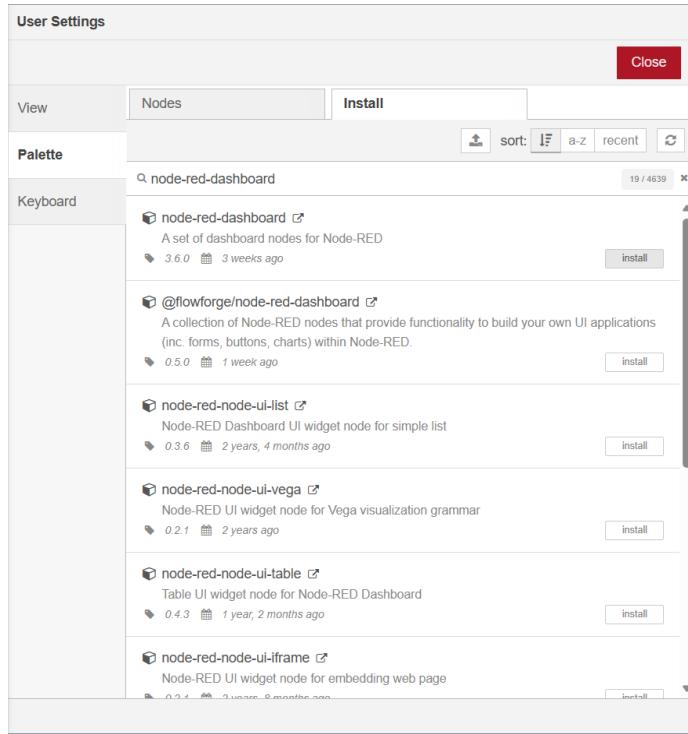
Haz clic en el menú de Node-RED ubicado en la parte superior derecha de la pantalla:



Esquina superior derecha de interfaz de Node-RED

## Paso 3: Abrir la pestaña install.

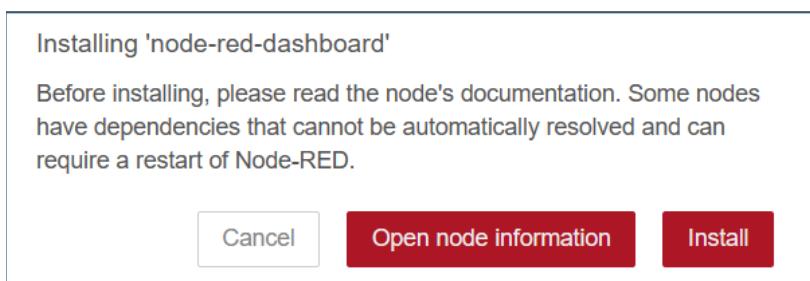
Diríjase a la pestaña “Install” en donde podrá ver todos los nodos que están disponibles para instalar en Node-RED, escriba “node-red-dashboard” en el buscador y luego oprima el botón “Install” sobre el primer nodo de la búsqueda:



Resultado de búsqueda de nodos

#### Paso 4: Seleccione instalar.

Una vez haya seleccionado “install”, se abrirá un cuadro de diálogo como el que se muestra abajo, en donde debe oprimir “Install”:



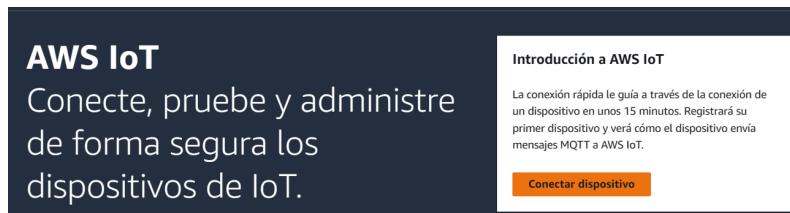
Cuadro de diálogo para confirmar la instalación de “node-red-dashboard”

## Sección 6: Creación de objetos AWS IoT

En esta sección, se detallará el proceso de creación de dos objetos AWS IoT que nos permiten simular objetos virtuales que envían y reciben datos a través de Internet.

## Paso 1: Creación de Objeto

Seleccione la opción de servicios de AWS que se encuentra en el panel superior y haga clic sobre el apartado de “IoT Core”, una vez hecho esto seleccione la opción “Conectar Dispositivo”

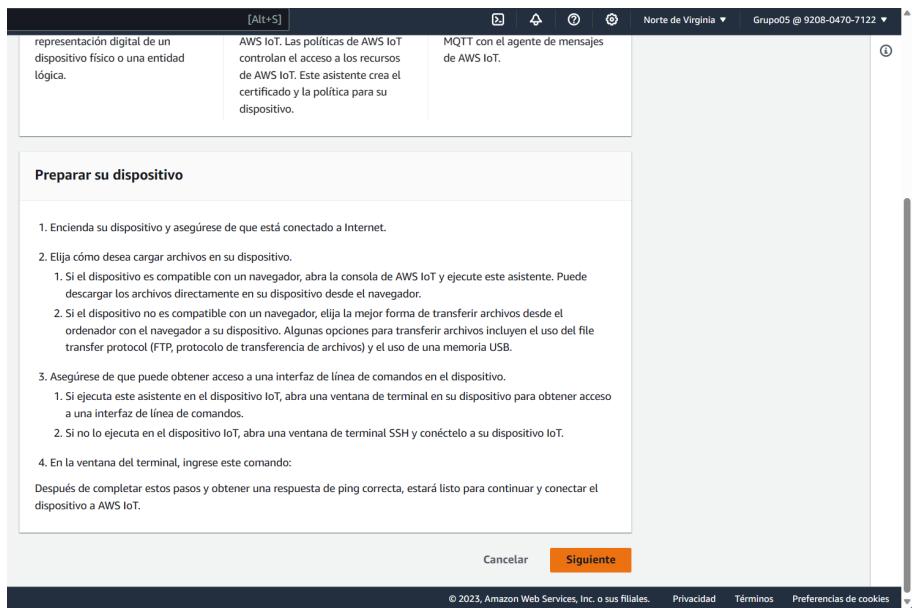


Apartado de IoT Core

## Paso 2: Configuración del objeto IoT

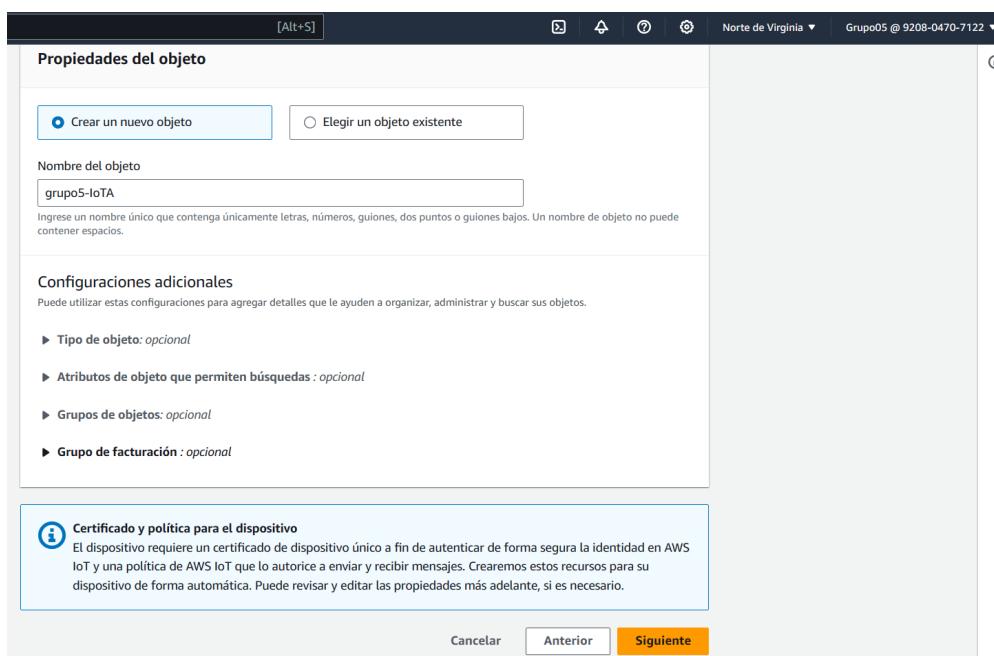
Al momento de la configuración de los objetos IoT se siguieron los siguientes pasos para ambos objetos creados:

1. Simplemente se seleccionó el botón “siguiente”



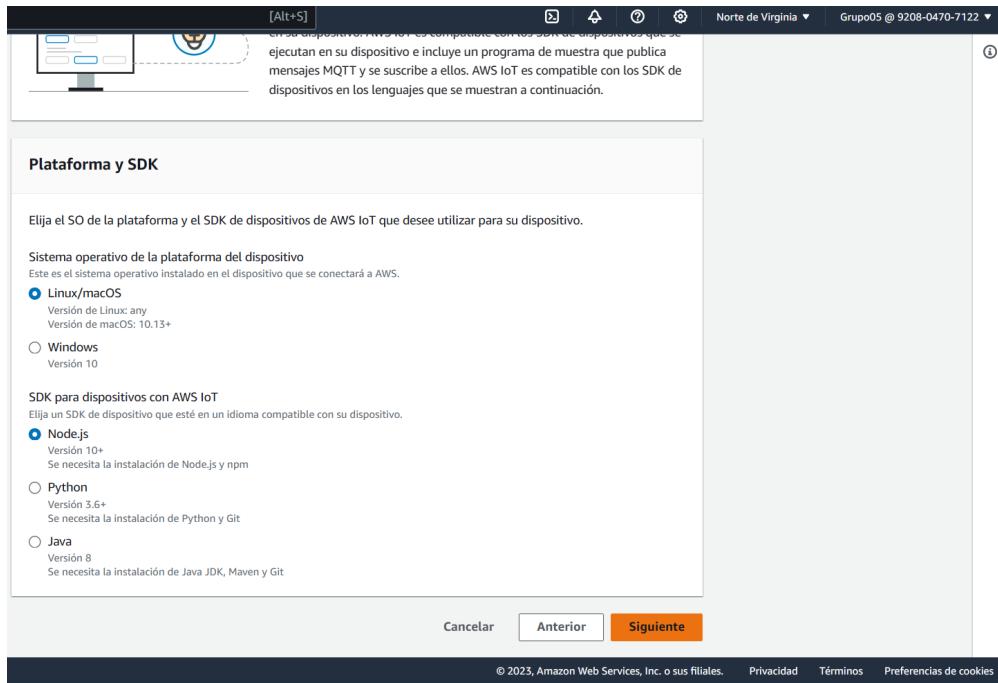
### Apartado de IoT Core

2. Se seleccionó la opción “Crear un nuevo objeto”, se ingresó el nombre del nuevo objeto, que en la primera ocasión fue “grupo5-IoTA” y para el segundo objeto fue “grupo5-IoTB”, en cuanto a las configuraciones solo se utilizaron las que están por defecto. Una vez hecho esto se oprimió el botón “Siguiente”:



### Creación de un nuevo objeto IoT

3. Luego se eligió el tipo de sistema operativo dejando las opciones por defecto y se seleccionó el botón “Siguiente”. Una vez hecho esto se creó el nuevo objeto AWS IoT:



Selección del S.O

4. Despues de haber creado el nuevo dispositivo se procedió a descargar el kit de conexión que se utilizará más adelante para poder acceder al objeto creado.

The screenshot shows the AWS IoT Device Management console. In the center, there's a large green banner stating: "AWS IoT ha creado correctamente el recurso de objeto grupo5-IoTA y ha generado el kit de conexión." Below this, there are three columns of files:

- Certificado**: grupo5-IoTA.cert.pem
- Clave privada**: grupo5-IoTA.private.key
- SDK para dispositivo**: Node.js
- Script para enviar y recibir mensajes**: start.sh
- Política**: grupo5-IoTA-Policy
- Ver política**

A "Descargar" (Download) button is present. To the right, a tooltip explains: "Si lo ejecuta desde un navegador en el dispositivo, después de descargar el kit de conexión, estará en la carpeta de descarga del navegador. Si no lo está ejecutando desde un navegador en su dispositivo, tendrá que transferir el kit de conexión desde la carpeta de descarga de su navegador a su dispositivo utilizando el método que probó cuando preparó su dispositivo en el paso 1." A "Descargar kit de conexión" button is also shown.

Below this, a section titled "Descomprima el kit de conexión en su dispositivo" (Extract the connection kit on your device) provides instructions: "Una vez que el kit de conexión esté en su dispositivo, descomprimalo con este comando:" followed by "unzip connect\_device\_package.zip" and a "Copiar" (Copy) button.

Descarga del kit de conexión

5. Finalmente se oprimió el botón “Siguiente” y luego en “Continuar”:

The screenshot shows the final step of creating an IoT object. The top banner says: "AWS IoT ha creado correctamente el recurso de objeto grupo5-IoTA y ha generado el kit de conexión." Below it, there's a command line interface with a copy button: ".start.sh". A note says: "Paso 3: Volver a esta pantalla para ver los mensajes de su dispositivo. Despues de ejecutar el script de inicio, vuela a esta pantalla para ver los mensajes entre el dispositivo y AWS IoT. Los mensajes del dispositivo aparecen en la siguiente lista." A "Suscripciones" table shows one entry: "sdk/test/js" with status "Esperando mensajes". Buttons for "Poner en pausa" (Pause) and "Borrar" (Delete) are also visible. At the bottom, there are "Cancelar" (Cancel), "Anterior" (Previous), and a prominent orange "Continuar" (Continue) button.

Creación del objeto IoT

## Sección 7: Edición de políticas de objetos IoT

En esta sección, se detalla el proceso de edición de permisos sobre la política de los objetos

IoT creada con anterioridad. Cada política es una acción o conjunto de acciones que se pueden realizar sobre determinado objeto creado.

Para poder configurar las políticas que vienen por defecto en las instancias de los objetos IoT creados y agregar las políticas necesarias para utilizar el servicio de Node-RED debe de seguir los siguientes pasos, para ambos objetos, aunque por simplicidad sólo se mostrará cómo hacerlo con un solo objeto.

## Paso 1: Abrir el resumen del objeto IoT

En el panel lateral izquierdo, en la sección de “Administrar”, en el apartado de “Políticas” seleccione la opción “Políticas” y elija el objeto que desea editar, en nuestro caso fue “grupo5-IoTA”, y haga clic en el botón “Editar versión activa”:

The screenshot shows the AWS IoT Policies page. On the left, there's a sidebar with navigation options like 'Todos los dispositivos', 'Objetos', 'Grupos de objetos', etc. Under 'Seguridad', 'Políticas' is selected. In the main area, it shows the policy 'grupo5-IoTA-Policy'. A table provides details: ARN (arn:aws:iot:us-east-1:920804707122:policy/grupo5-IoTA-Policy), Versión activa (1), Creado (September 27, 2023, 18:38:27 (UTC-03:00)), and Actualización más reciente (September 27, 2023, 18:38:27 (UTC-03:00)). Below this, a table lists the active version's rules:

Efecto de la política	Acción de la política	Recurso de la política
Allow	iot:Publish	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/java
Allow	iot:Publish	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/python
Allow	iot:Publish	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/js
Allow	iot:Receive	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/java
Allow	iot:Receive	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/python
Allow	iot:Receive	arn:aws:iot:us-east-1:920804707122:topic/sdk/test/js

Políticas del objeto grupo5-IoTA

## Paso 2: Eliminar los permisos actuales para el objeto IoT

Una vez haya realizado el paso anterior, verá las políticas habilitadas por defecto en el

objeto creado, las cuales se deben eliminar haciendo clic sobre el botón “Eliminar”:

The screenshot shows the AWS IoT Policy Editor interface. At the top, there are tabs for "Instrucciones de política" and "Ejemplos de políticas". Below this, a section titled "Documento de política" contains three policy statements. Each statement has three dropdown menus: "Efecto de la política" (set to "Permitir"), "Acción de la política" (set to "iot:Publish", "iot:Receive", and "iot:PublishRetain" respectively), and "Recurso de la política" (set to "arn:aws:iot:us-east-1:920804707122:tr"). To the right of each statement is a blue "Eliminar" button. At the bottom of this section is a button labeled "Agregar nueva instrucción".

Políticas del objeto grupo5-IoTA

### Paso 3: Agregar nueva política

Acto seguido deberá seleccionar la opción “Aregar nueva instrucción”, con el efecto de “**Permitir**” y tanto la acción como el recurso deberá tener es símbolo “\*”, haciendo referencia a todos los recursos y símbolos. Una vez hecho esto haga clic en el botón “Guardar como nueva versión”, esto creará una nueva versión de las políticas de este objeto. Es importante señalar que esta medida, si bien simplifica la usabilidad, plantea cuestionamientos en términos de seguridad.

The screenshot shows the AWS IoT Policy Editor interface. At the top, the navigation bar includes 'AWS IoT > Seguridad > Políticas > grupo5-IoTA-Policy > Editar'. The main title is 'Editar política: grupo5-IoTA-Policy (versión 1)'. Below the title, there are two tabs: 'Instrucciones de política' (selected) and 'Ejemplos de políticas'. A section titled 'Documento de política' contains an 'Información' link and a note: 'Una política de AWS IoT contiene una o varias instrucciones de política. Cada instrucción de política contiene acciones, recursos y un efecto que concede o deniega las acciones por parte de los recursos.' There are three dropdown menus: 'Efecto de la política' (set to 'Permitir'), 'Acción de la política' (set to '\*'), and 'Recurso de la política' (set to '\*'). Buttons for 'Creador' and 'JSON' are at the top right. Below these are buttons for 'Eliminar' and 'Agregar nueva instrucción'. Another section titled 'Estado de la versión de la política' shows 'Política activa' with a checkbox 'Establecer la versión editada como la versión activa de esta política'. A note says 'Puede cambiar esta configuración más adelante en la página de detalles de la política.' At the bottom right are 'Cancelar' and 'Guardar como nueva versión' buttons.

Nueva política agregada

## Paso 4: Establecer la versión creada como activa.

Luego de haber creado la nueva versión de las políticas es necesario habilitarlas, para esto debe seleccionar la versión recién creada, en nuestro la versión número 3, y dar clic en el botón “Establecer como activa”:

The screenshot shows the AWS IoT Policy Versions page. At the top, a green success message says 'Política grupo5-IoTA-Policy actualizada correctamente, creando la versión 3 de la política.' Below it, another message says 'La política se ha actualizado correctamente y se ha creado la versión 3 de la política.' A table lists the versions: Version 3 (arn:aws:iot:us-east-1:920804707122:policy/grupo5-IoTA-Policy), Version 2, and Version 1. The 'Versiones' tab is selected. A section titled 'Versión activa: 3' shows the policy details: 'Efecto de la política' (Allow), 'Acción de la política' (\*), and 'Recurso de la política' (\*). Below this is a table titled 'Todas las versiones (1/3)'. It shows three rows: Version 3 (Active, created September 27, 2023, 19:22:51 UTC-03:00), Version 2 (Inactive, created September 27, 2023, 19:10:30 UTC-03:00), and Version 1 (Inactive, created September 27, 2023, 18:38:27 UTC-03:00). Buttons for 'Creador' and 'JSON' are at the top right of the version table. Below the table are buttons for 'Eliminar', 'Establecer como activa' (highlighted in blue), 'Editar versión', and 'Ver JSON'.

Activación de nueva política

## **Sección 8: Configuración de nodo MQTT**

En esta sesión se va a detallar una serie de pasos para configurar los nodos MQTT desde el navegador para poder comunicar los diferentes nodos ubicados en cada una de las instancias creadas previamente.

### **¿Qué es MQTT?**

MQTT (Message Queuing Telemetry Transport) es un protocolo de comunicación ligero y eficiente diseñado para facilitar la transmisión de mensajes entre dispositivos en redes con ancho de banda limitado o conexiones inestables. MQTT es ampliamente utilizado en el contexto de la Internet de las cosas (IoT) y otras aplicaciones de comunicación máquina a máquina (M2M).

Este protocolo se basa en el principio de publicación y suscripción, donde los dispositivos pueden actuar como editores que envían mensajes a temas específicos, y otros dispositivos pueden suscribirse a estos temas para recibir los mensajes relevantes.

### **Paso 1: Obtener el conjunto de permisos**

Como previamente se mencionó, al momento de configurar el objeto IoT se debe descargar el kit de conexión el cual contiene permisos que serán utilizados para establecer la conexión con el protocolo de comunicación MQTT.

### **Paso 2: MQTT en Node-RED**

El nodo MQTT out en Node-RED se utilizará para enviar mensajes y datos entre las dos instancias EC2 corriendo Node-RED.

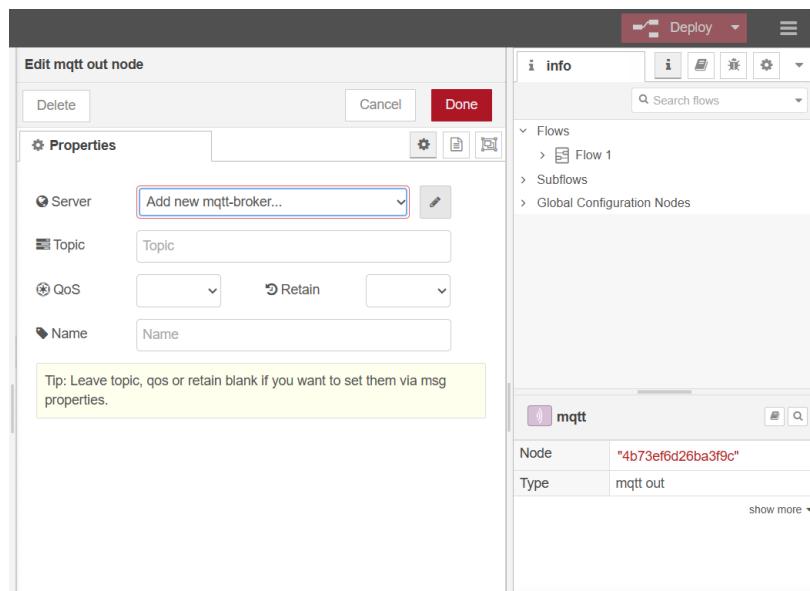
Diríjase al panel lateral izquierda de Node-RED, desde allí busque y seleccione el nodo llamado “mqtt in”.



Nodo mqtt en Node-RED

### Paso 3: Configurar nodo mqtt

1. Seleccionar el nodo “mqtt out” y luego en la opción de “add new mqtt-broker”:



Panel de configuración de nodo mqtt

2. Esto desplegará un nuevo panel de configuración similar al siguiente:

The screenshot shows a configuration interface for adding a new MQTT broker node. The left pane is titled "Edit mqtt out node > Add new mqtt-broker config node". It contains a "Properties" section with fields for "Name", "Connection" (with tabs for "Security" and "Messages"), "Server" (set to "e.g. localhost" with port "1883"), "Protocol" (set to "MQTT V3.1.1"), "Client ID" (set to "Leave blank for auto generated"), "Keep Alive" (set to "60"), and "Session" (with "Use clean session" checked). The right pane shows a summary of the added node: "undefined:1883", "Node": "b2eee4c7e3bd5ef6", "Type": "mqtt-broker".

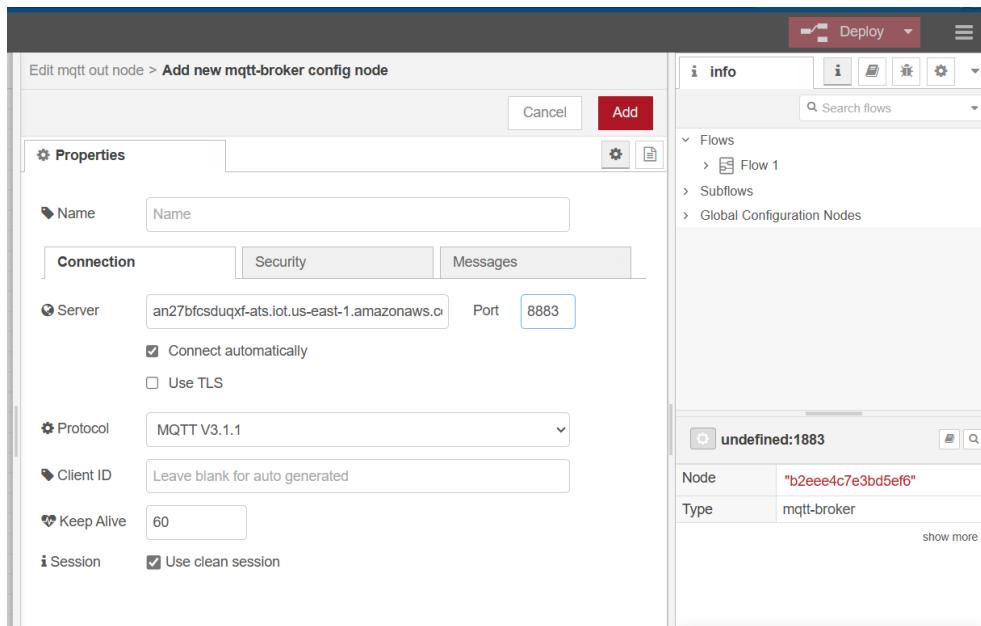
Panel de configuración de nodo mqtt, sección “add-broker”

3. Diríjase a la sesión de configuración del objeto creado en su sesión de AWS y copie el “punto de enlace”:

The screenshot shows the AWS IoT Configuration section. It includes the "Punto de enlace de datos de dispositivo" (Device Data Link endpoint) which is "an27bfcsduqxf.ats.iot.us-east-1.amazonaws.com" and the "Configuraciones de dominio" (Domain Configuration) section which is currently empty.

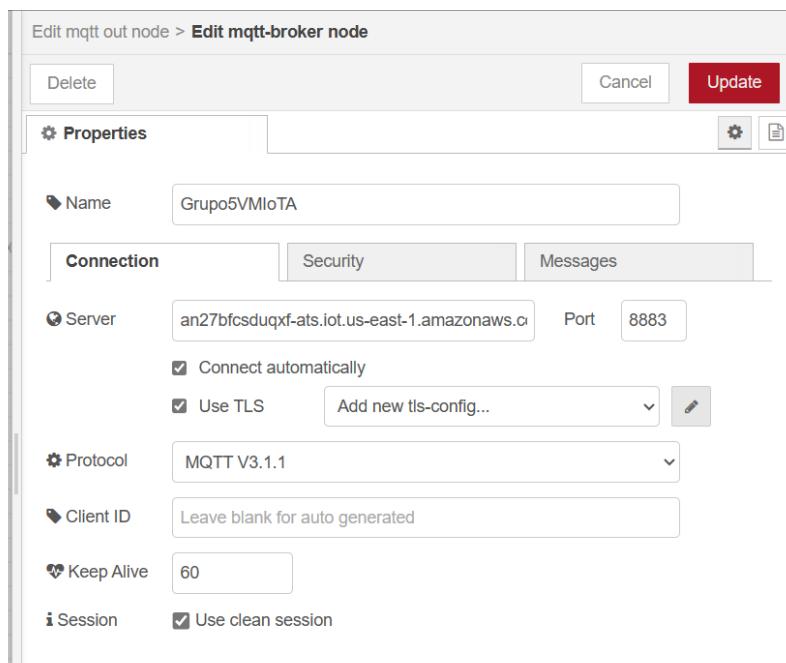
Sección de configuración del objeto IoT

4. Pegue el punto de enlace copiado en la opción de “Server” en el panel de configuración del nodo mqtt y establezca el puerto de la salida al 8883:



Panel de configuración de nodo mqtt, sección “add-broker”

5. Marque la opción de “Use TLS”, al habilitar TLS (Transport Layer Security) o Seguridad de la Capa de Transporte, que es un protocolo de seguridad utilizado para proteger las comunicaciones en línea.



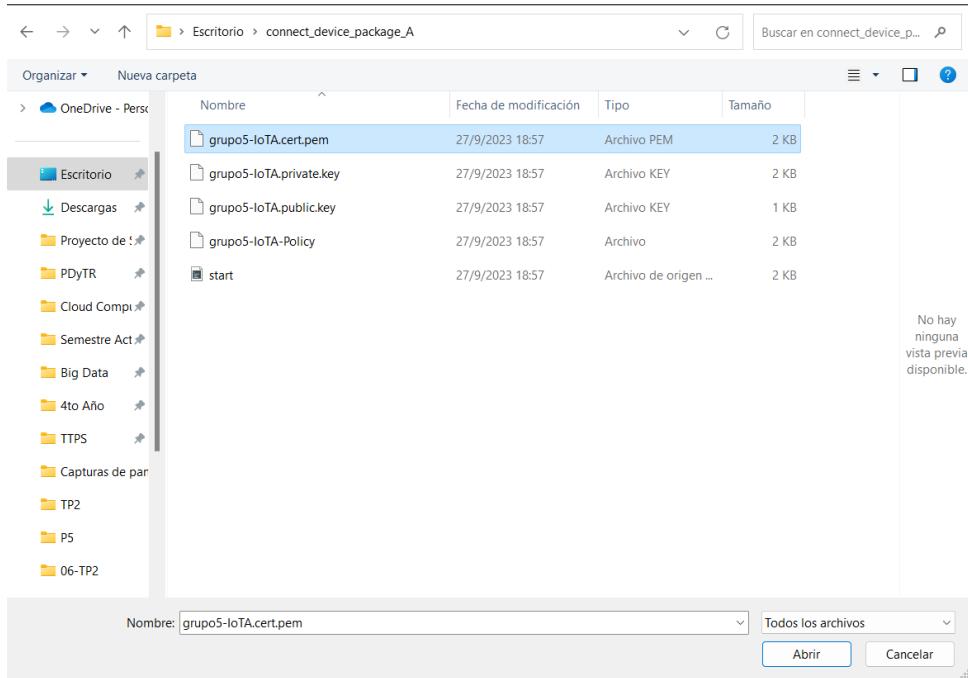
Panel de configuración de nodo mqtt, sección “add-broker”

6. Luego debe proceder a ingresar las claves necesarias para poder utilizar el nodo mqtt utilizando el protocolo TLS, haciendo clic en el botón “Add new tls-config”:

The screenshot shows a configuration dialog titled "Edit mqtt out node > Edit mqtt-broker node > Add new tls-config config node". The dialog has a "Properties" tab selected. It includes fields for "Certificate" (with an "Upload" button), "Private Key" (with an "Upload" button), "Passphrase" (with a placeholder "private key passphrase (optional)"), "CA Certificate" (with an "Upload" button), and a checked "Verify server certificate" checkbox. There are also fields for "Server Name" (placeholder "for use with SNI") and "ALPN Protocol" (placeholder "for use with ALPN"). At the bottom is a "Name" field with the placeholder "Name". A red "Add" button is located at the top right.

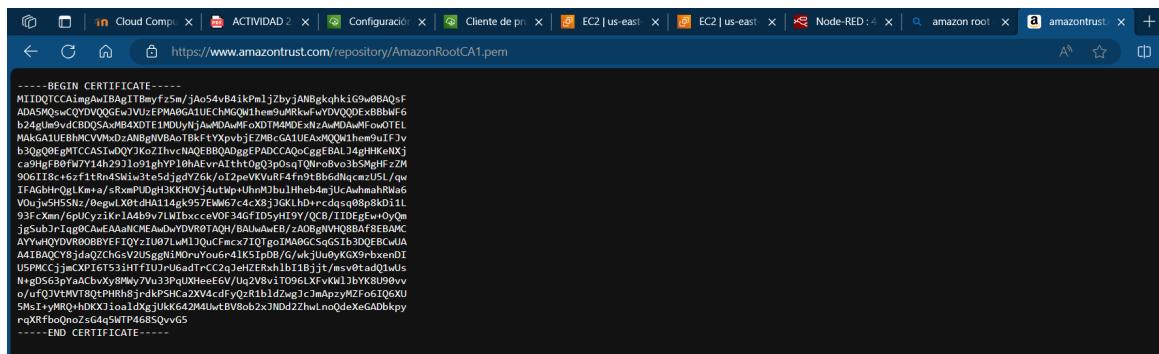
Configuración de protocolo TLS

7. Una vez hecho esto debe proceder a cargar los certificados descargados previamente en el kit de conexión, solo serán necesarios los archivos con extensión .cert.pem para el campo “Certificate” y el archivo con “private” en su nombre y extensión .pem para el campo “Private Key”:

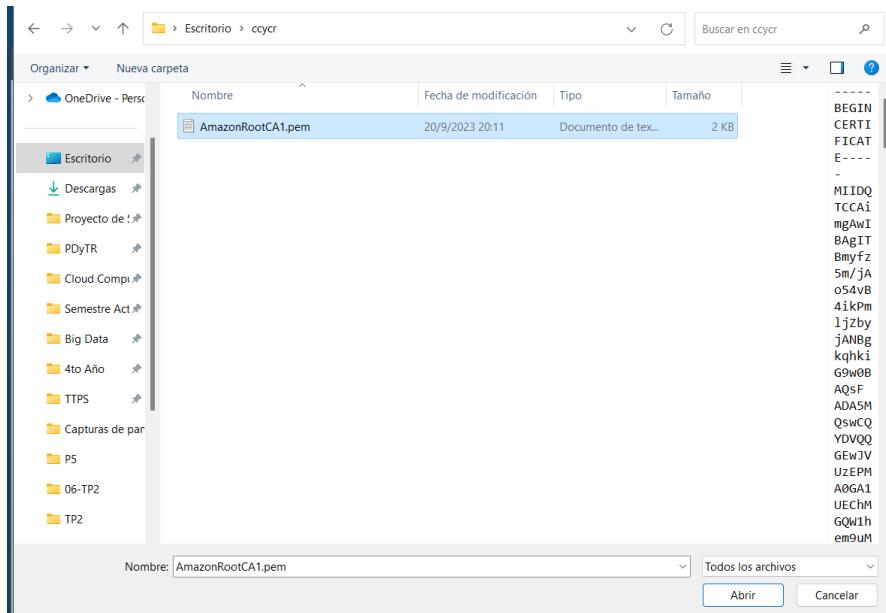


## Kit de conexión

8. En el caso del archivo para el campo “CA certificate”, un certificado de CA es un certificado digital emitido por una autoridad de certificación (CA), por lo que los clientes SSL (como los navegadores web) pueden usarlo para verificar los certificados SSL firmados por esta CA. Deberá dirigirse al enlace número 4 que se especifica en la última sección y descargar el archivo y cambiar tanto el nombre como el formato del archivo por “AmazonRootCA1.pem”:



Certificado CA de AWS



Carga del archivo para el campo CA certificate

9. Finalmente debe dar clic a la opción “Update” para actualizar la nueva configuración y tendrá el nodo mqtt preparado para utilizar.

En las propiedades del nodo mqtt out establezca el QoS (Quality of Service) en 0.

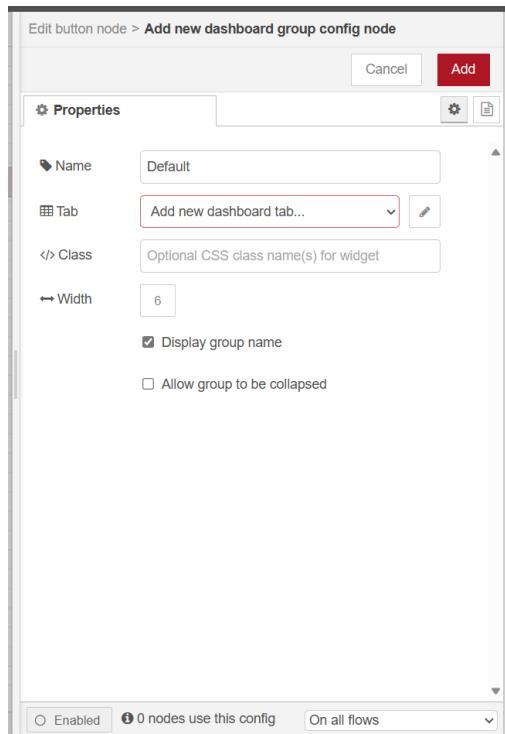
## Sección 9: Configuración del dashboard en el objeto A

En esta sección, vamos a explorar cómo hemos configurado y diseñado el dashboard en la instancia EC2 A. Este dashboard es el corazón de nuestro sistema de control remoto de iluminación basado en IoT. Veamos los pasos que seguimos:

### Paso 1: Creación de los botones

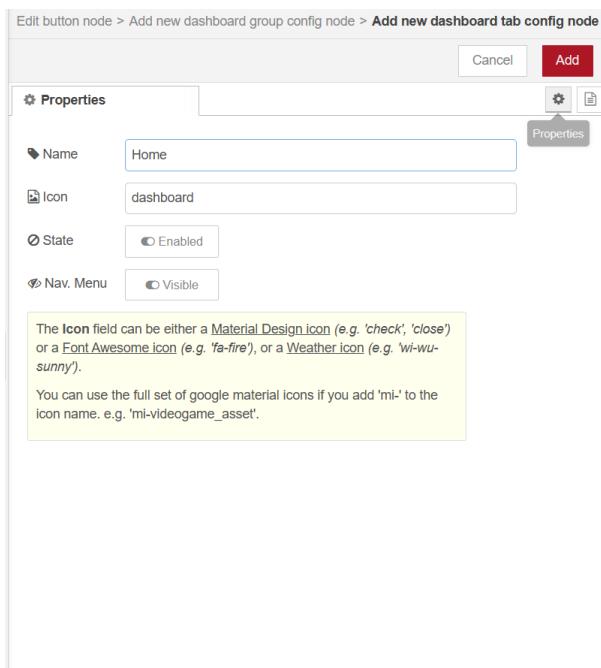
En el panel de nodos a la izquierda, encontramos el nodo "Botón" que utilizaremos bajo la solapa “dashboard”.

Hacemos clic en el nodo "Botón" y configuramos un nuevo panel (tab) en la parte superior. Para hacerlo, seleccionamos la opción "Add new dashboard tab..." en el campo "Tab":



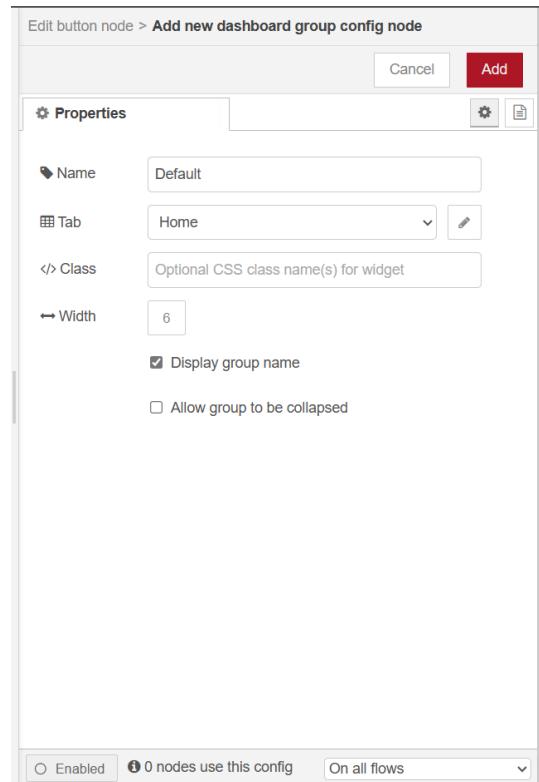
Creación del dashboard

Aparecerá una ventana de configuración para la nueva pestaña (tab). Dejamos los valores predeterminados. Hacemos clic en "Propiedades".



Configuración del dashboard

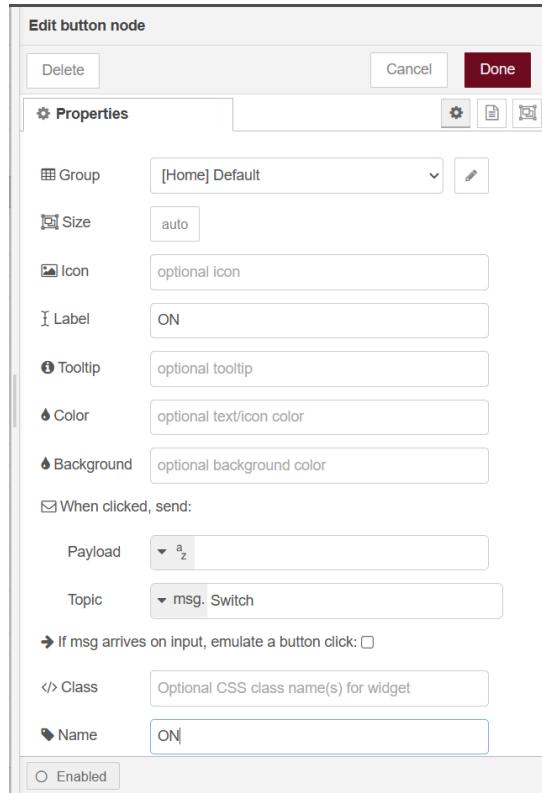
En la ventana de configuración de la pestaña (tab), dejamos el nombre por defecto o, si se prefiere, se puede cambiar el nombre por uno más descriptivo, como "Control Remoto" y hacemos clic en "Add" para crear el panel:



Configuración del grupo

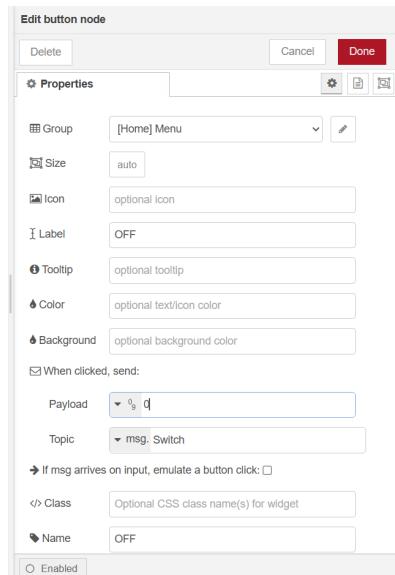
Regresamos a la configuración del nodo "Botón" y en el campo "Group", seleccionamos el nuevo panel "Home" que acabamos de crear.

A su vez, configuramos el botón de encendido. Le damos el nombre "ON" en el campo "Label" y establecemos que, al hacer clic, envíe el mensaje "1" como una cadena por el tema "Switch":



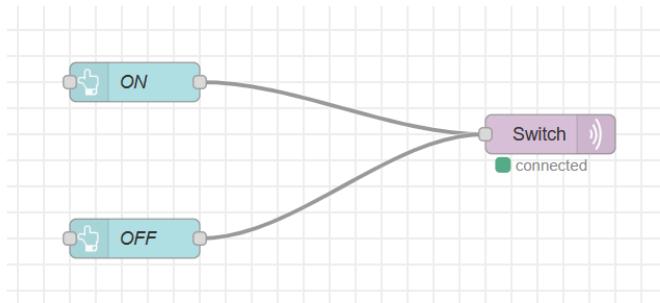
Configuración del botón "ON"

Creamos otro botón, esta vez para apagar el LED. Le asignamos el nombre "OFF" en el campo "Label" y configuraremos para que, al hacer clic, envíe el mensaje "0" como una cadena por el tópico "Switch":



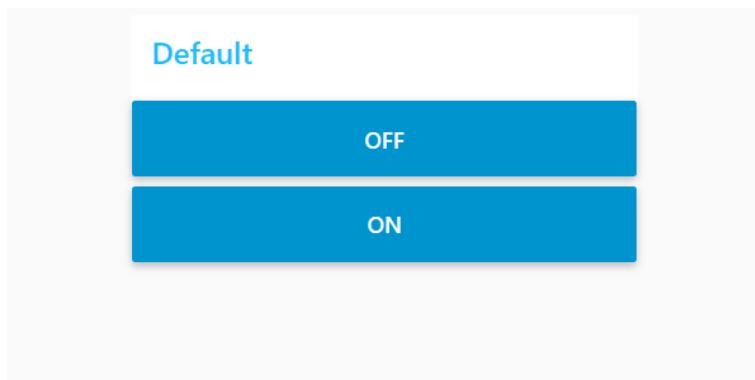
Configuración del botón "OFF"

Conectamos estos dos nuevos nodos de botones al nodo MQTT de salida que hemos configurado previamente:



Conexión de los nodos de botones al nodo MQTT de salida

Finalmente, observamos cómo queda el dashboard hasta este punto:

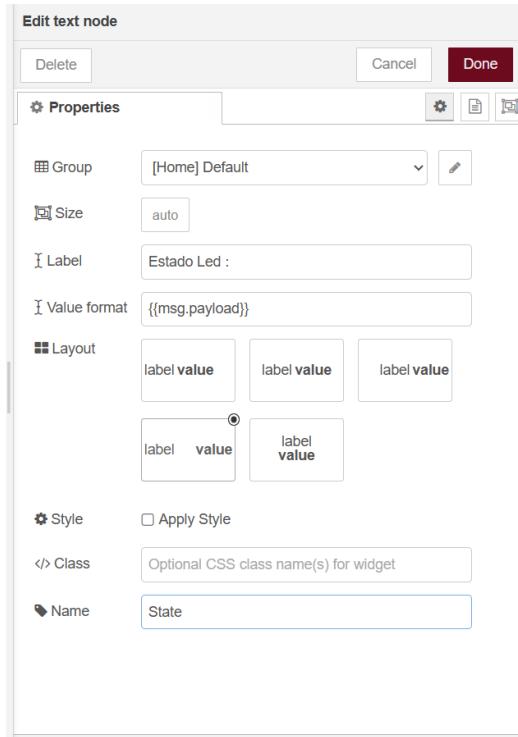


Vista del dashboard con los botones creados

## Paso 2: Creación del texto de estado del LED

Para mostrar el estado del LED en el dashboard, seleccionamos un nodo del tipo "text" de la lista de nodos y lo arrastramos al área de trabajo de Node-RED.

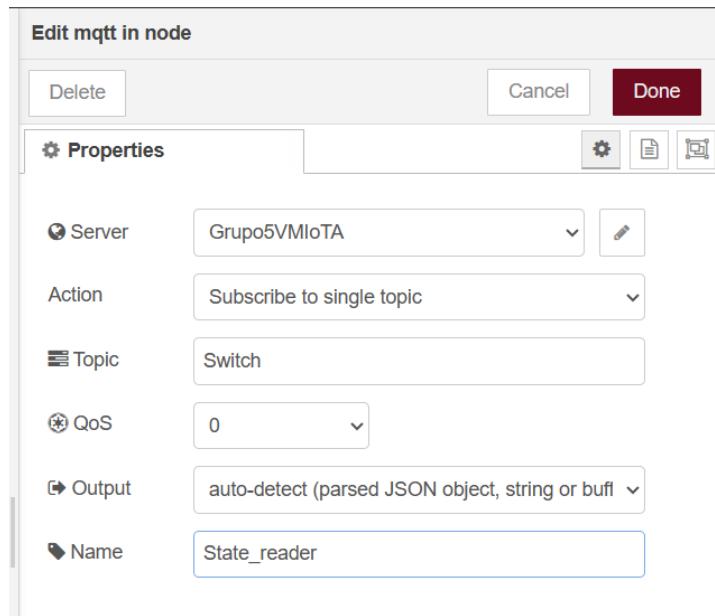
En la ventana de configuración del nodo tipo "text", en el campo "Group", seleccionamos el dashboard "Home" que hemos creado previamente. Luego, en el campo "Label", ingresamos "Estado Led: ". Dejamos el campo "Value format" tal como está por defecto, ya que queremos que muestre el payload del mensaje, que será "ON" o "OFF". En la sección "Layout" de la configuración del nodo de texto, seleccionamos la opción que mejor se ajuste a los requisitos de diseño de la actividad:



Configuración del nodo para la label

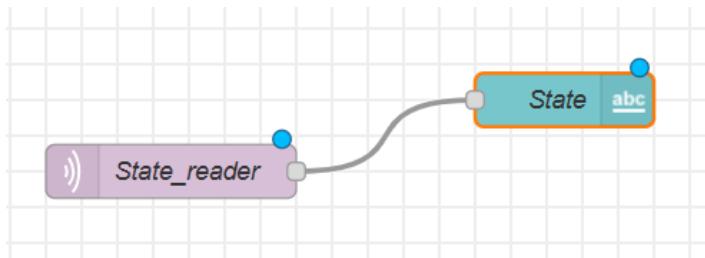
Establecemos el nombre del nodo como "State".

Creamos un nodo MQTT de entrada llamado "State\_reader" que se encargará de recibir la información del tópico "Switch":



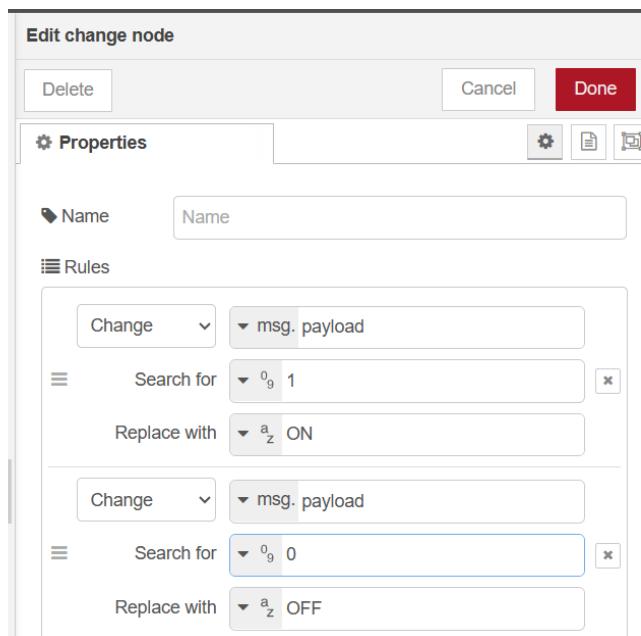
Configuración del nodo "State\_reader"

Conectamos los dos nodos, de manera que el nodo "State" reciba la información del tópico "Switch" y se actualice automáticamente con el estado del LED:



Conexión de los nodos "State\_reader" y "State"

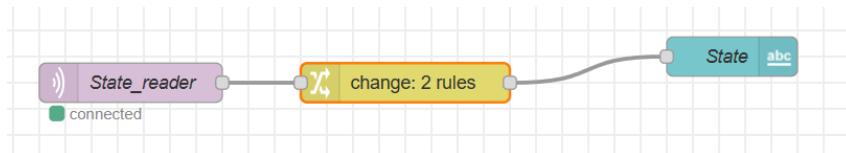
Aunque la configuración hasta este punto funcionaría correctamente, se presenta un problema en el tópico "Switch". El payload del mensaje contiene un "0" o un "1" para representar el estado del LED, pero deseamos que la etiqueta muestre "Estado Led: ON" o "Estado Led: OFF". Para solucionar esto, creamos un nodo intermedio del tipo "change" que tomará el "1" y el "0" del tópico "Switch" y los cambiará por "ON" y "OFF". Este nodo "change" se conecta entre los dos nodos anteriores.



Configuración del nodo "change"

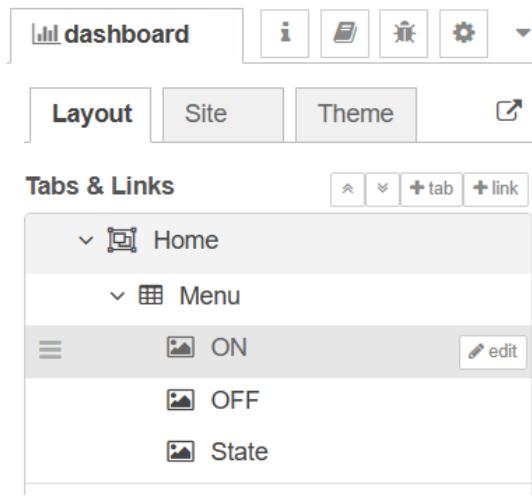
En la configuración del nodo "change", se pueden ver dos reglas agregadas que se encargan de cambiar el "1" por "ON" y el "0" por "OFF".

La conexión en Node-RED quedaría de la siguiente manera, con el nodo "change" actuando como intermediario entre el nodo "State\_reader" y el nodo "State":



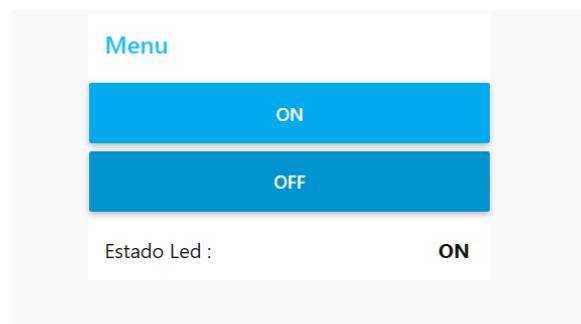
Conexión de los nodos "State\_reader", "change" y "State"

En la parte derecha de la ventana de Node-RED, encontramos una jerarquía en la pestaña "Layout" de nuestro dashboard. En esta sección, podemos reorganizar individualmente cada uno de los componentes del layout para que se muestren en el orden deseado. En este caso, colocamos los dos botones primero y luego el nodo "State" para mantener un diseño ordenado:



Jerarquía del dashboard

Finalmente, el dashboard quedaría completamente funcional, ya que el nodo "State" se actualiza automáticamente de acuerdo al estado del LED.

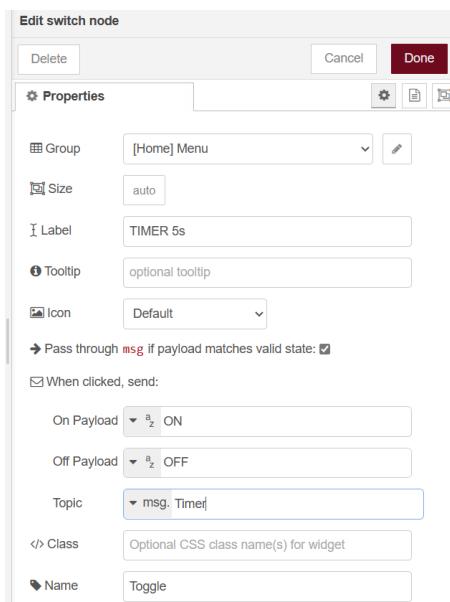


Dashboard final con el label de estado del LED

### Paso 3: Implementación de la funcionalidad “TIMER de 5s”

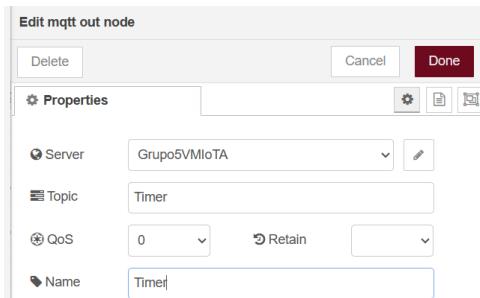
Ahora procedemos a crear el toggle o switch que se utilizará para habilitar la funcionalidad del "TIMER de 5s". Creamos un nuevo nodo del tipo "switch" que se encuentra en la lista de nodos bajo el apartado "dashboard".

En la configuración de este nodo "switch", establecemos su etiqueta como "TIMER de 5s". Configuramos el nodo de tal manera que cuando esté en estado activado, el toggle envíe el mensaje "ON" al tópico "Timer", y cuando se desactive, envíe el mensaje "OFF" al mismo tópico. Además, nombramos el nodo como "Toggle":



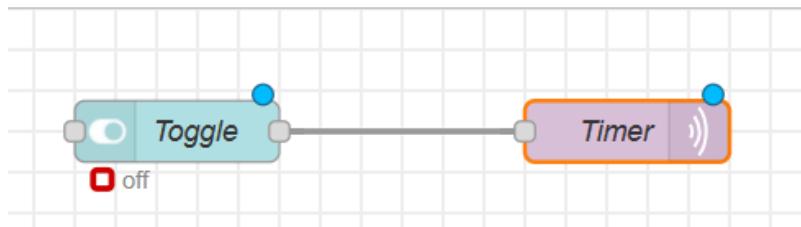
Configuración del nodo “Toggle”

Una vez creado el nodo "Toggle", debemos conectarlo a un nodo MQTT de salida que publique el tópico "Timer". Configuramos el nodo MQTT de salida como se muestra en la siguiente imagen:



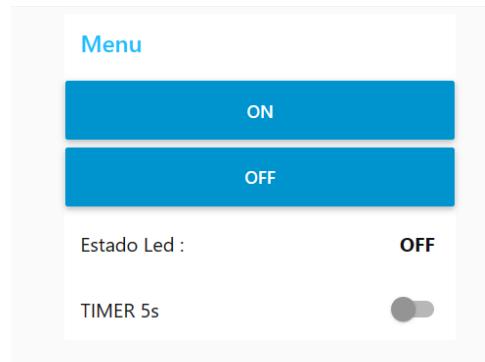
Configuración del nodo MQTT de salida "Timer"

La conexión entre los nodos "Toggle" y "Timer" se realiza de la siguiente manera:



Conección entre los nodos "Toggle" y "Timer"

Finalmente, al agregar el nuevo toggle en el dashboard, el aspecto se vería de la siguiente forma:

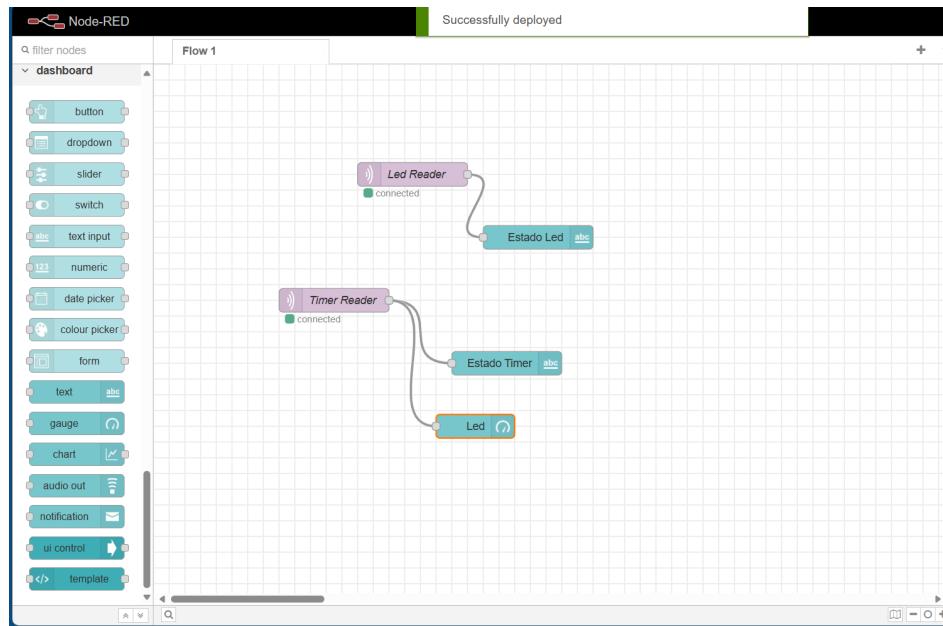


Dashboard con el nuevo toggle "TIMER de 5s"

A pesar de que la funcionalidad completa del "TIMER de 5s" aún no se ha implementado, se ha logrado configurar el nodo toggle para actualizar el tópico "Timer" y enviar su estado a la instancia B, lo que representa un paso importante hacia la implementación completa de esta funcionalidad en futuras iteraciones del proyecto.

## Sección 10: Configuración del dashboard en el objeto B

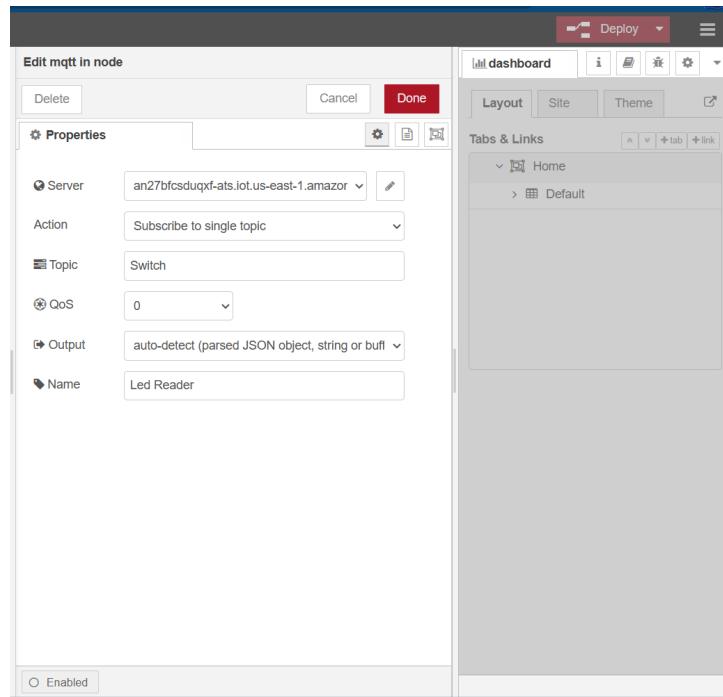
En esta sección se explicará el funcionamiento de los nodos, creados y conectados en el dashboard del objeto B, para simular el “ENCENDIDO/APAGADO” de la luminaria, informando el estado de la “LUMINARIA” y el estado del “TIMER”.



Dashboard final del objeto B

### Paso 1: Configuración de nodo MQTT “Led Reader”

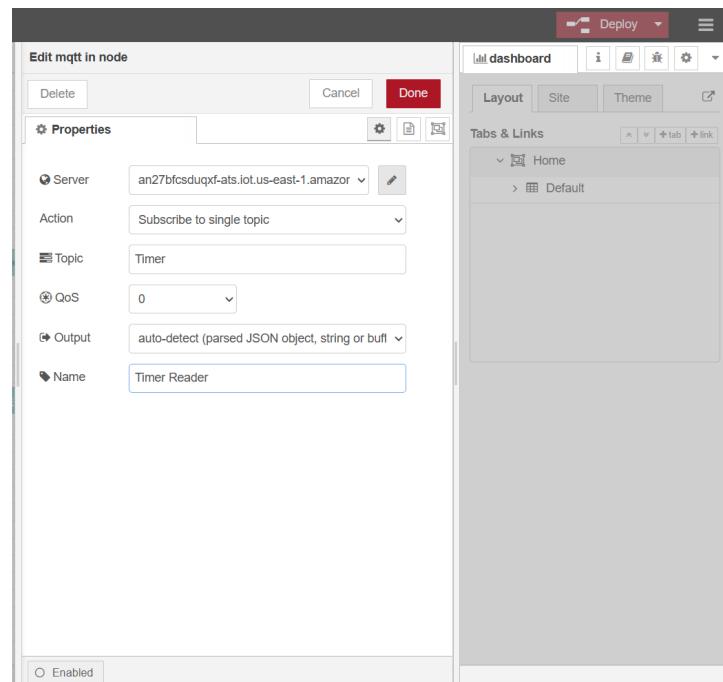
Este nodo es el encargado de recibir la información transmitida por el canal “Timer”, luego esta información es pasada al nodo “Estado Led” para informar el dato que recibe. La propiedad QoS fue configurada en el nivel más bajo, además de configurar la etiqueta “Name” para darle un nombre descriptivo a este nodo.



Edición del node MQTT Led Reader

## Paso 2: Configuración de nodo MQTT “Timer Reader”

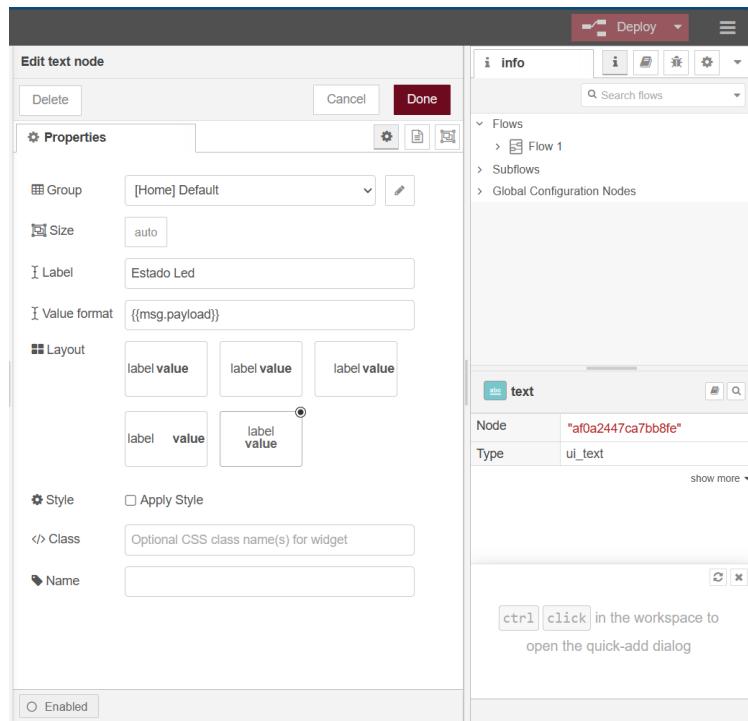
Este nodo es el encargado de recibir la información transmitida por el canal “Switch”, luego esta información es pasada al nodo “Estado Timer” para informar según corresponda, el estado del timer ubicado en el objeto A.



Edición del node MQTT Led Reader

### Paso 3: Configuración de nodo text input “Estado Led”

La funcionalidad de este nodo es simplemente mostrar el estado del led, según el valor que reciba desde el nodo anterior.



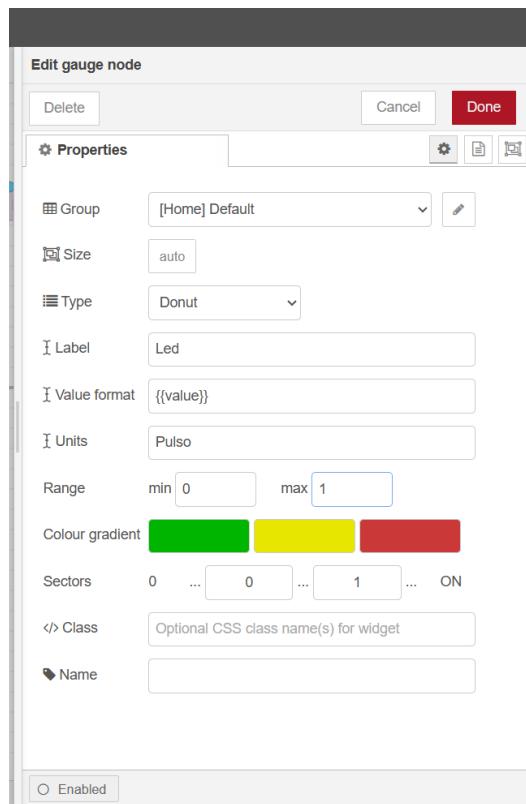
Edición del node textinput Estado Led

### Paso 4: Configuración de nodo text input “Estado Timer”

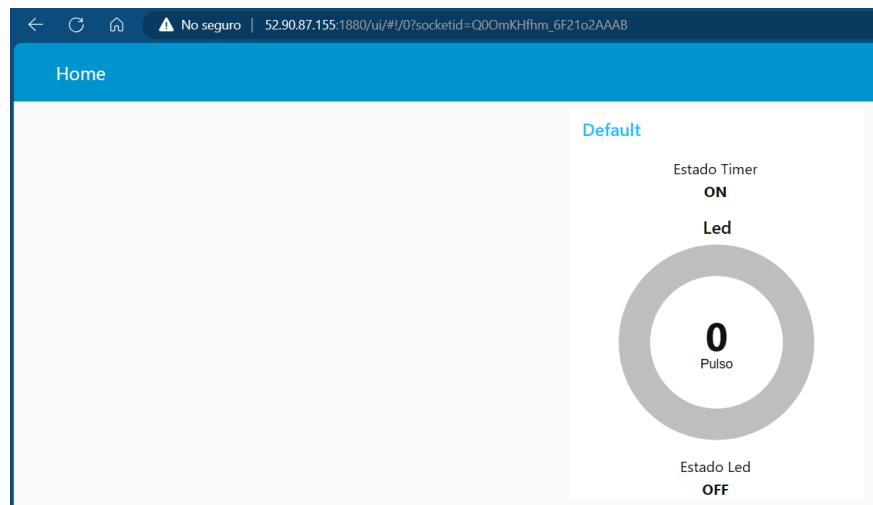
Este nodo se configuró de la misma forma que el nodo “Estado Led”, solo cambiando la etiqueta “Name”.

### Paso 5: Configuración de nodo gauge “Led”

El último nodo se configuró de manera tal que a partir del nodo que reciba desde su nodo predecesor, se encienda una luz roja si es que el valor es correcto y en caso de no recibir el valor correcto se mantendrá en un color gris, representando que la luz led está apagada:



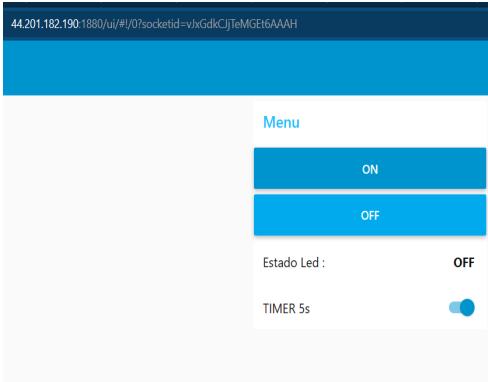
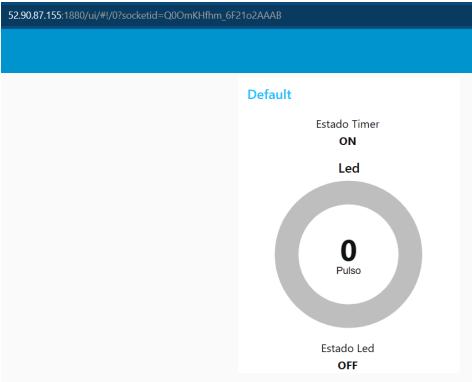
Edición del node gauge Led



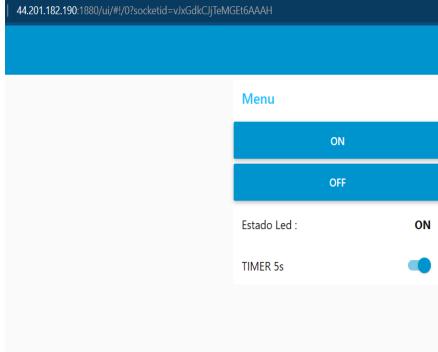
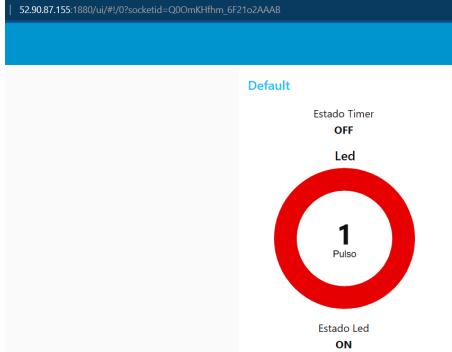
Resultado del Dashboard con led apagado

## Sección 11: Validaciones de funcionamiento

Estado inicial de la aplicación APAGADO:

Objeto A	Objeto B
	

Operación de ENCENDIDO:

Objeto A	Objeto B
	

## Sección 12: Terminación de instancias

Una vez realizada la actividad se deben de eliminar los objetos e instancias creados en AWS, para evitar el uso innecesario de sus recursos.

Para esto diríjase al apartado de sus instancia, para cada instancia que desee terminar, seleccione la instancia, en el botón desplegable seleccione la opción “Terminar instancia”.

Resumen de instancia de i-0392c15ccd13d89f0 (grupo5-A) <a href="#">Información</a>		<a href="#">C</a>	<a href="#">Conectar</a>	<a href="#">Estado de la instancia ▲</a>	<a href="#">Acciones ▼</a>
Se ha actualizado hace less than a minute					
ID de la instancia <a href="#">i-0392c15ccd13d89f0 (grupo5-A)</a>	Dirección IPv4 pública <a href="#">52.90.87.155 [dirección abierta □]</a>	Direcci on <a href="#">17</a>	Estado de la instancia <a href="#">En ejecución</a>	Detener instancia <a href="#">Iniciar instancia</a> <a href="#">Reiniciar instancia</a>	
Dirección IPv6 -	Estado de la instancia <a href="#">En ejecución</a>	DNS d ominio <a href="#">ed</a>	Nombre DNS de IP privada (solo IPv4) <a href="#">ip-172-31-82-43.ec2.internal</a>	Hibernar instancia <a href="#">Terminar instancia</a> <a href="#">[dirección abierta □]</a>	zonaws.com
Tipo de nombre de anfitrión Nombre de IP: ip-172-31-82-43.ec2.internal	Nombre DNS de IP privada (solo IPv4) <a href="#">ip-172-31-82-43.ec2.internal</a>	Tipo de instancia t2.micro	ID de VPC <a href="#">vpc-0157552403fa567fd □</a>	Direcciones IP elásticas -	Hallazgo de AWS Compute Optimizer <a href="#">Reintentar</a>
Responder al nombre DNS de recurso privado IPv4 (A)	ID de subred <a href="#">subnet-0bd313524362217f1 □</a>	ID de VPC <a href="#">vpc-0157552403fa567fd □</a>			User: arn:aws:iam::920804707122:user/Grupo05 is not authorized to perform: compute-optimizer:GetEnrollmentStatus us on resource: * because no identity-based policy allows th e compute-optimizer:GetEnrollmentStatus action
Dirección IP asignada automáticamente <a href="#">52.90.87.155 [IP pública]</a>					
Rol de IAM -					
IMDSv2 Optional					

Finalización de la instancia creada.

## Sección 13: Conclusión

Esta actividad ha proporcionado a los participantes una valiosa introducción a conceptos clave de IoT, comunicación en tiempo real a través de MQTT y la configuración de servicios en la nube.

A medida que avanzamos en este campo, se abre la oportunidad de explorar la implementación completa del temporizador y la expansión de este proyecto para abordar desafíos más complejos. En última instancia, esta experiencia sienta las bases para futuros proyectos de IoT y el desarrollo de soluciones más avanzadas en el ámbito de la tecnología de la nube.

## **Sección 14: Fuentes**

1. Amazon Web Services (AWS). [Documentación de Amazon EC2](#).
2. PuttyGen. [Descarga de PuttyGen](#).
3. Guía oficial de AWS. [Connect to Your Linux Instance from Windows Using PuTTY](#).
4. Archivo Root CA [AmazonRootCA1](#)
5. Actividad anterior [Act.1 CC y CR - Documentos de Google](#)