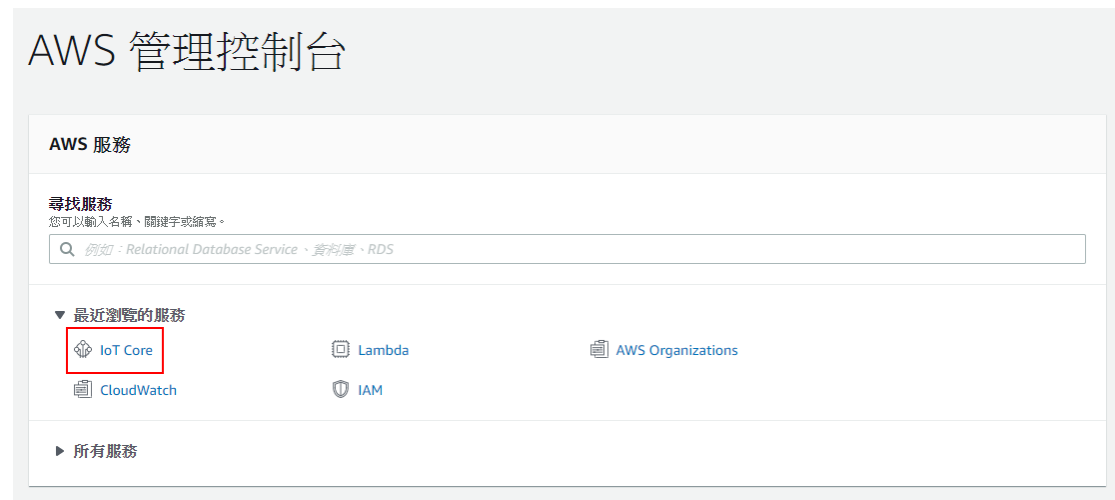


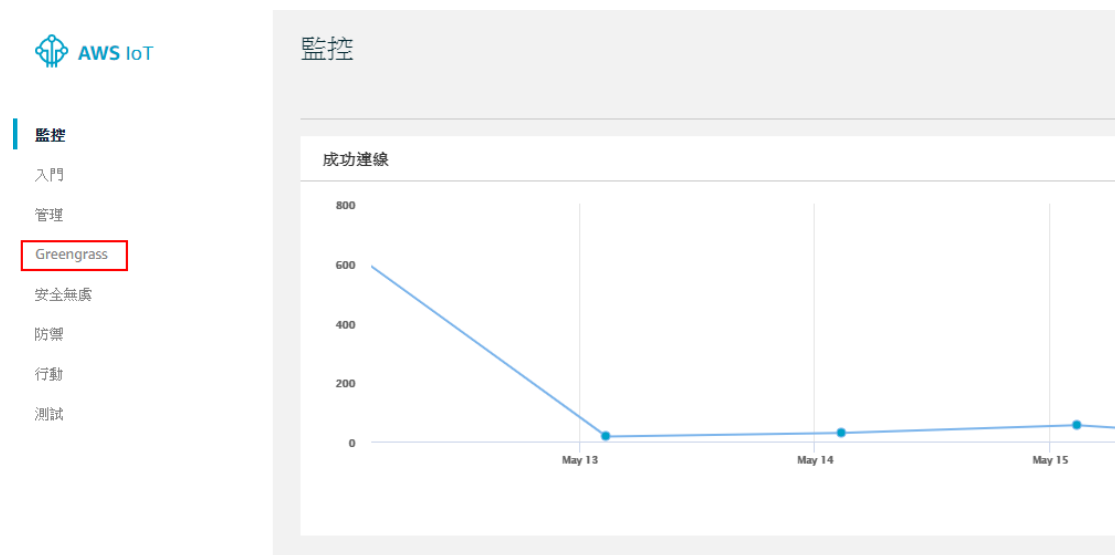
01.

登入 AWS 平台後，搜尋並選擇下圖紅框項目：



02.

進入 IoT Core 後，選擇下圖內左側紅框項目：



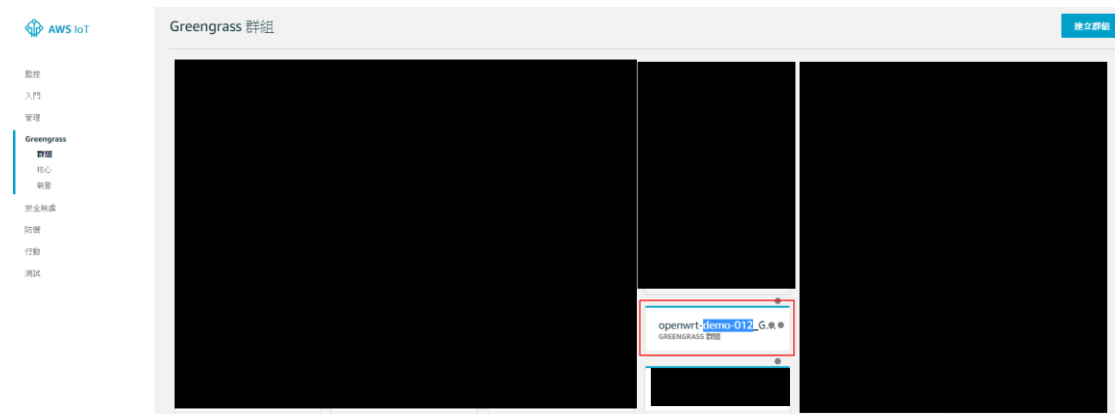
03.

進入 Greengrass 後，選擇下圖左側紅框項目：



04.

選擇群組內欲設定的 group:



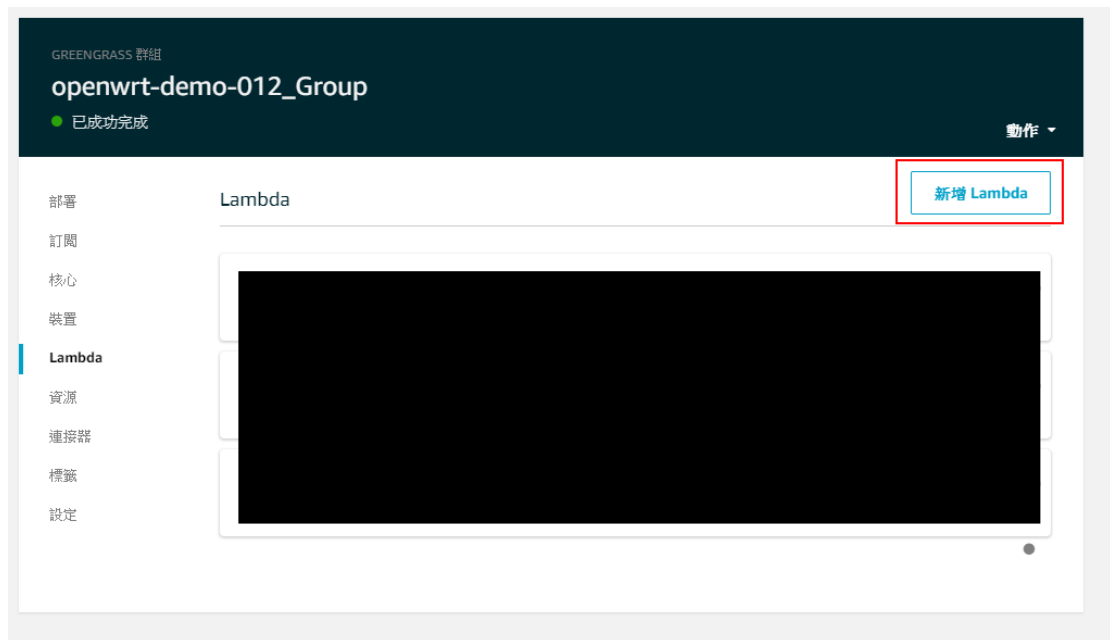
05.

進入 group 後，選擇下圖左側紅框項目：



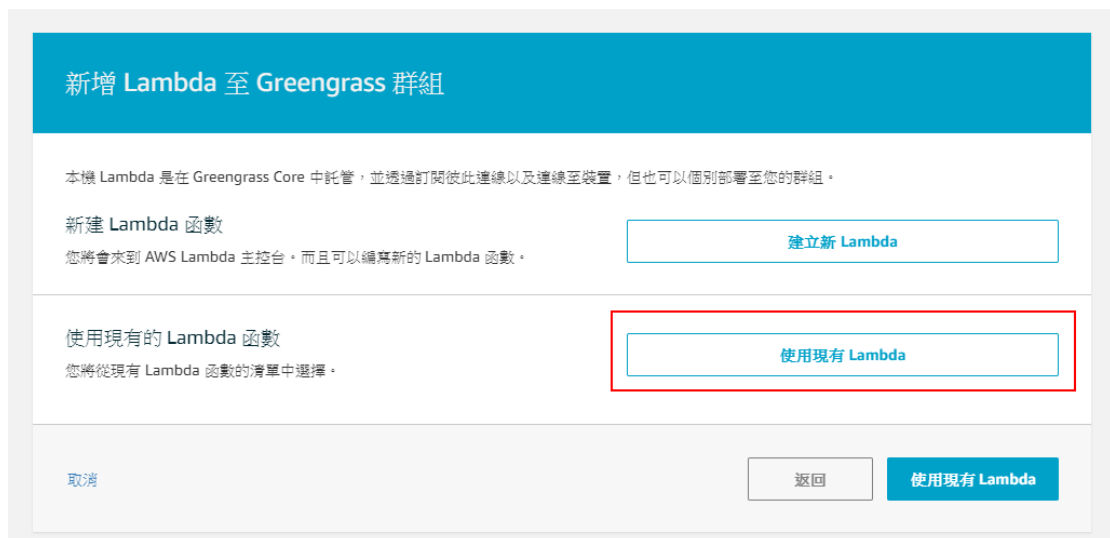
06.

進入 Lambda 管理畫面後，選擇下圖紅框按鈕：



07.

進入新增 Lambda 畫面後，選擇紅框項目：



08.

進入 Lambda 清單後, 選擇欲部屬的 Lambda 項目, 並進行下一步:

新增 LAMBDA 至 GREENGRASS 群組

使用現有 Lambda

選擇 Lambda

搜尋所有 Lambda 函數及標籤

<input type="radio"/>	[Redacted]	Node.js 12.x
<input type="radio"/>	[Redacted]	Node.js 12.x
<input checked="" type="radio"/>	rogerTestMqttPublish	Node.js 12.x
<input type="radio"/>	[Redacted]	Greengrass Executable
<input type="radio"/>	[Redacted]	Greengrass Executable
<input type="radio"/>	[Redacted]	Node.js 12.x
<input type="radio"/>	[Redacted]	Java 8
<input type="radio"/>	[Redacted]	Node.js 12.x

取消 返回 下一步

09.

進入 Lambda 版本清單後, 選擇欲部屬的 Lambda 版本, 並完成:

新增 LAMBDA 至 GREENGRASS 群組

選擇 Lambda 版本

選擇 Lambda 版本

搜尋 Greengrass Lambda 版本

<input type="radio"/>	別名: publishDemo-1
<input checked="" type="radio"/>	別名: publishDemo-2
<input type="radio"/>	第 2 版
<input type="radio"/>	第 1 版

取消 返回 完成

10.

完成後，點選剛才新增的 **Lambda** 程式旁的編輯按鈕，如下圖右側紅框處：
(需先點擊“兩點”圖示，才會出現選單)



11.

進入編輯選單後，下圖紅框處按照截圖設定，下圖藍框處依照 **Lambda** 程式需求設定：

群組專屬 Lambda 組態

rogerTestMqttPublish

檢視 AWS Lambda 中的函數

別名 publishDemo-2 移除版本

執行身分 ?

- ☒ 使用群組預設值 (目前：ggc_user/ggc_group)
- ☐ 其他使用者 ID/群組 ID

容器化 ?

- ☒ 使用群組預設值 (目前：Greengrass 容器)
- ☐ Greengrass 容器 (一律)
- ☐ 沒有容器 (一律)

記憶體限制

16

MB

逾時

3

秒

Lambda 生命週期

- ☐ 隨需函數
- ☒ 將此函數設定為長時間存留且無限期持續執行

12.

設定完成後，選擇更新：

Lambda 生命週期

☐ 隨需函數

☒ 將此函數設定為長時間存留且無限期持續執行

對 /sys 目錄的讀取存取

☒ 停用

☐ 啟用

輸入承載資料類型

☒ JSON

☐ 二元

環境變數

您可以將環境變數定義成可從您的函數程式碼存取的鍵值組。這些可用來儲存組態設定，而不需要變更函數程式碼。

索引鍵

值

例如顏色

例如藍色

×

新增此 Lambda 函數的另一個版本

您可以將 Lambda 函數的個別歷史版本新增至群組，這些版本各有其本身的一組群組專屬組態，而您的群組會將這些版本視為個別且不同的 Lambda 函數。

新增另一個版本

取消

更新

13.

更新後，選擇下圖紅框處返回群組畫面：

服務 ▾ 資源群組 ▾ ★

←

GREENGRASS LAMBDA

rogerTestMqttPublish

OPENWRT-DEMO-012_GROUP

版本

版本

編輯

資源	版本	記憶體限制	逾時	Lambda 生命週期	容器化
	版本 publish...	16 MB	3 秒	隨需	預設

14.

回群組畫面後，選擇下圖左側紅框處，進入訂閱畫面後，選擇下圖右側紅框處：



15.

進入新增畫面後，來源選項按照下圖紅框處設定：



16.

目標選項，按照下圖紅框處設定，請選擇剛剛已新增的欲部屬的 **Lambda** 程式，之後選擇下一步：

選擇目標

未選擇物件

關閉

服務	裝置	Lambda	連接器
搜尋			
<div></div>			
rogerTestMqttPublish			

取消

返回

下一步

17.

在下圖紅框處，依需求設定 **MQTT 的 Topic 篩選條件**，之後選擇下一步

來源會發佈資料至目標。主題篩選條件可用來限制或控制目標接收的資料。如果未定義主題篩選，則會將所有來源訊息傳送至目標。

來源

IoT Cloud

服務

主題篩選條件

如何輸入主題篩選條件？

roger/remote/command

目標

rogerTestMqttPublish

LAMBDA



返回

下一步

18.

最後確認下圖紅框處的資訊是否正確後，選擇完成：

您的訂閱完成，而且您的物件已在此群組中連線。您現在可以儲存然後部署新的群組定義，以使此變更生效。

 IoT Cloud	服務
roger/remote/command	
 rogerTestMqttPublish	LAMBDA

[返回](#) [完成](#)

19.

完成後，回到群組畫面，選擇下圖右上紅框處：

(需先點擊“動作”，才會出現選單)

GRENGRASS 群組

openwrt-demo-012_Group

● 已成功完成

動作

部署

訂閱

核心

裝置




Lambda


資源

連接器

標籤

設定

來源	目標	主題
 IoT Cloud	 rogerTestMqttPublish:publi:	roger/remote/command
		

 連接器可以與 Lambda 函數、裝置和其他實體交換 MQTT 訊息

20.

待系統顯示部署完成後，就完成設定了：

GREENGRASS 群組

openwrt-demo-012_Group

● 已成功完成

動作 ▾

部署

訂閱

核心

裝置

Lambda

資源

連接器

標籤

設定

群組歷史記錄概觀

依據部署 ▾

已部署	版本	狀態	
2020年5月19日 上午11:05:27	8f7a1dec-e00f-4d67-92eb-6c8d8e689fb5	● 已成功完成	● ●
2020年5月18日 下午6:41:03	67522e2d-0c71-471a-baa5-2c724660c10b	● 已成功完成	● ● ●
2020年5月18日 下午6:17:37	31e11a0d-6bdd-47fd-b9fc-811ae3a4a4d3	● 已成功完成	● ● ●
2020年5月18日 下午5:30:54	e7d47de8-b12a-4788-be66-9d2cd1bef767	● 已成功完成	● ● ●
2020年5月18日 下午5:19:02	74bef87d-4279-44c3-8f62-ef1470b16ddc	● 已成功完成	● ● ●
2020年5月18日 下午5:09:20	013b97d7-e1ac-46dc-adc2-c4dbf0b8ba4c	● 已成功完成	● ● ●