**物聯網核心技術**

**實驗作業三**

**oneM2M及Node-red**

**中正大學資工系黃仁竑教授**

**Fall, 2017**

**實驗目的**

本實驗主要在教導學生如何設定oneM2M的環境，及Node-red的設定。

**實驗所需軟體**

* PC
  + Windows7/8/10
  + Vmware
  + Ubuntu 14..04 LTS 以上穩定版本
  + Java SDK1.7以上
  + Network Enable Environment
  + Maven 3以上版本
  + Node.js
  + Node-red
  + Postman

目錄

[Part1 oneM2M環境設置 3](#_Toc479254330)

[1. oneM2M 介紹 3](#_Toc479254331)

[2. oneM2M 環境建置 4](#_Toc479254332)

3 Postman建置、新增oneM2M Application.…………………………………….…….21

[Part 2 Node-Red介紹與環境建置 26](#_Toc479254333)

[1. Node-Red介紹 26](#_Toc479254334)

[2. 環境建置 27](#_Toc479254335)

[3. Node-red example 29](#_Toc479254336)

Part1 oneM2M環境設置

1. oneM2M介紹

oneM2M是ETSI（European Telecommunications Standards Institute，歐洲電信標準協會）旗下的專門對M2M（機器到機器通信）以及IoT（物聯網）的一個國際標準組織。其目的在於研發一種物聯網架構及相關技術標準，以實現跨部門或者跨行業不同機器之間的訊息傳遞。目前的應用大部分都用於: 車載通訊系統、智慧交通系統、智慧醫療、智慧電網、工業物聯網、智慧家居等。

 oneM2M採用簡單的橫向平台架構，分為三層，包括了應用層、通用服務層、網路服務層，如右圖。

應用層(AE)會使用在端點設備例如智能手機及Sensor應用中。通用服務層(CSE)可視為在oneM2M的服務功能實例，透過不同的參考點與不同的應用服務連結做資訊交換並存取網路服務層。網路服務層則是提供通用服務層實體所需的服務，例如:

裝置管理、定位、裝置啟動等等。oneM2m透過以上標準使得可以被便捷的遷入各種硬體或者軟體中的通用的機器到機器的通訊服務，以實現將無數終端設備連結到全球佈署的應用伺服器中。

目前on2M2M open source implementations有以下四個:

1. Eclipse OM2M

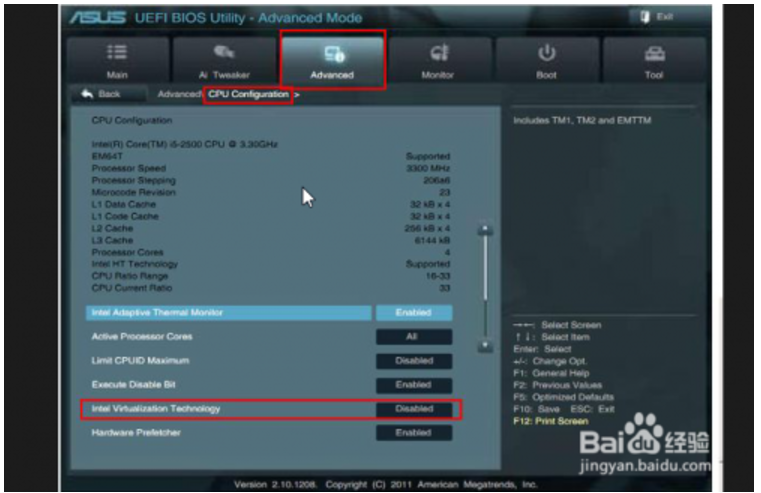
2. OCEAN Mobius

3. IoTDM

4. OASIS SI

本次Lab將會選用1 Eclipse OM2M作為使用。

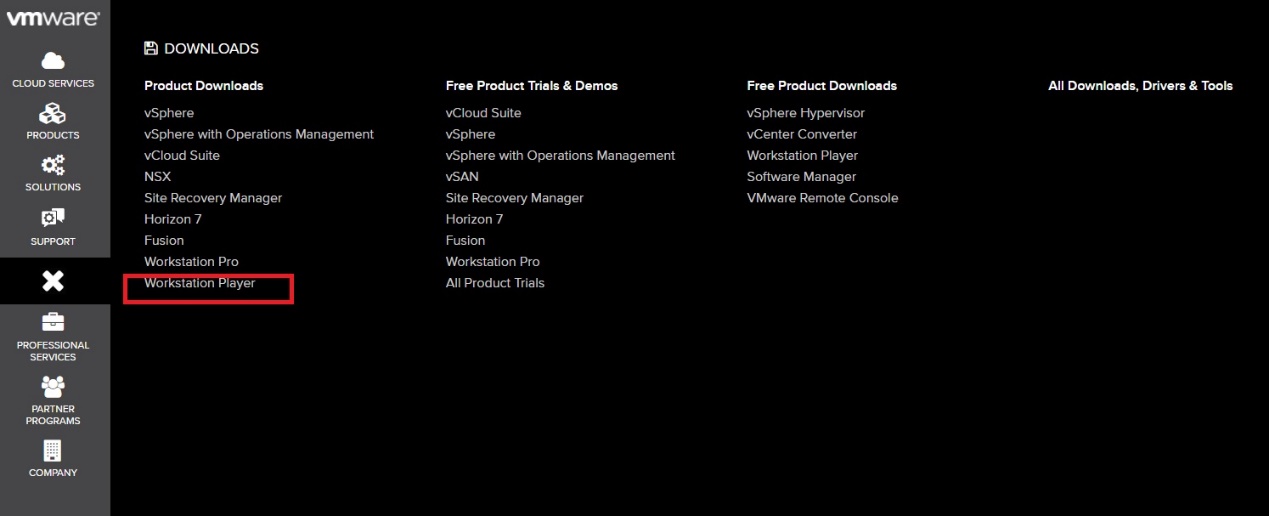
1. oneM2M 環境建置
2. oneM2M 環境建置
3. 請先重新啟動電腦按F2或者Del進入到Bios系統後，到CPU Options -> Advanced Setup，看到Intel Virtualization Technology將選項改成Enable儲存並重新開機即可。

以下為網路擷取範例示意圖，依每個人環境不同而有差異。

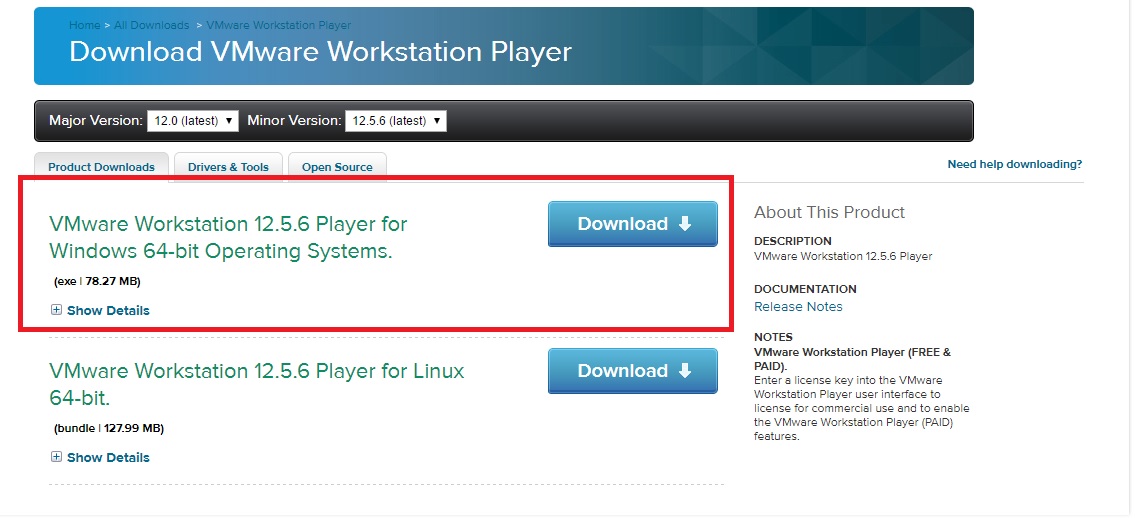
1. 請至<https://www.ubuntu-tw.org/modules/tinyd0/>下載ubuntu14.01LTS以上的版本並選擇好位元版本後開始下載。
2. 前往[VMware官方網站](https://www.vmware.com/tw.html)點選左方下載🡪Workstation Player🡪選取Windows版本Download

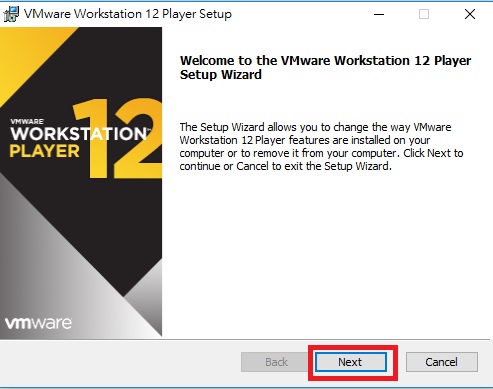
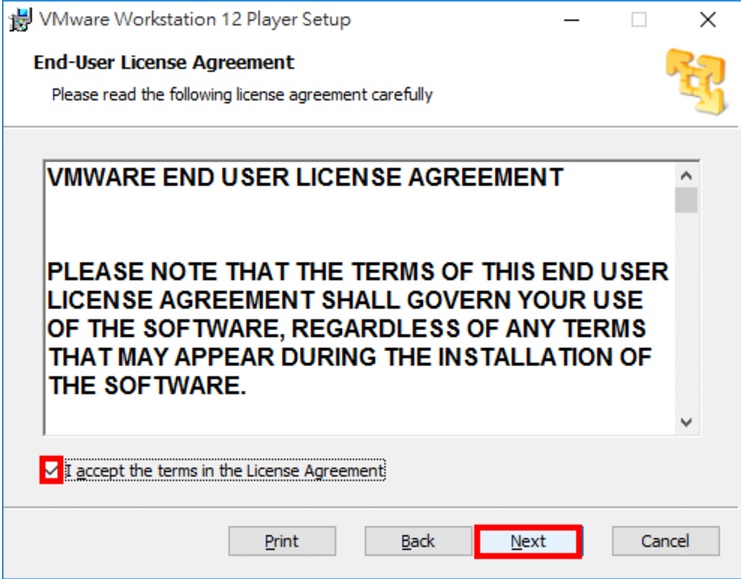
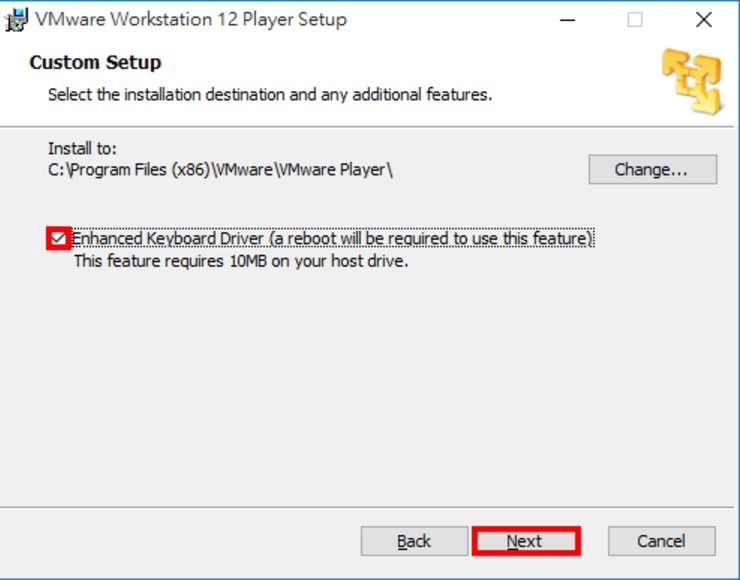
左方下載

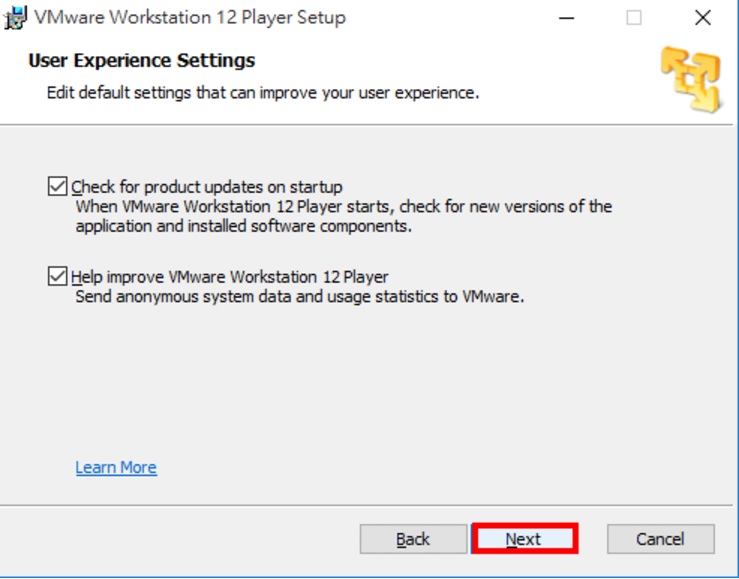
Workstation Player

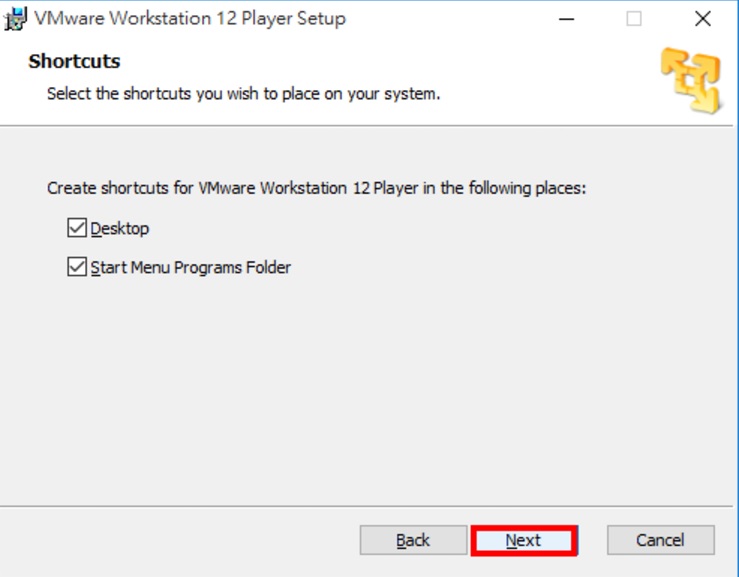


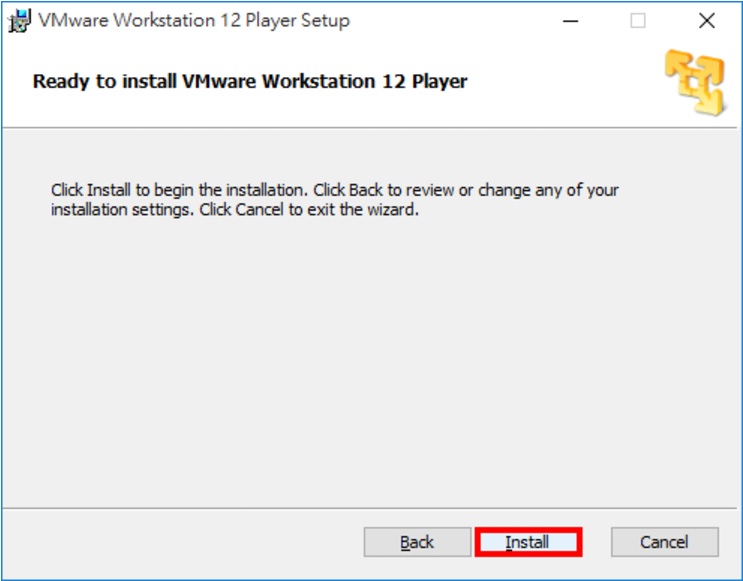
Windows版本Download

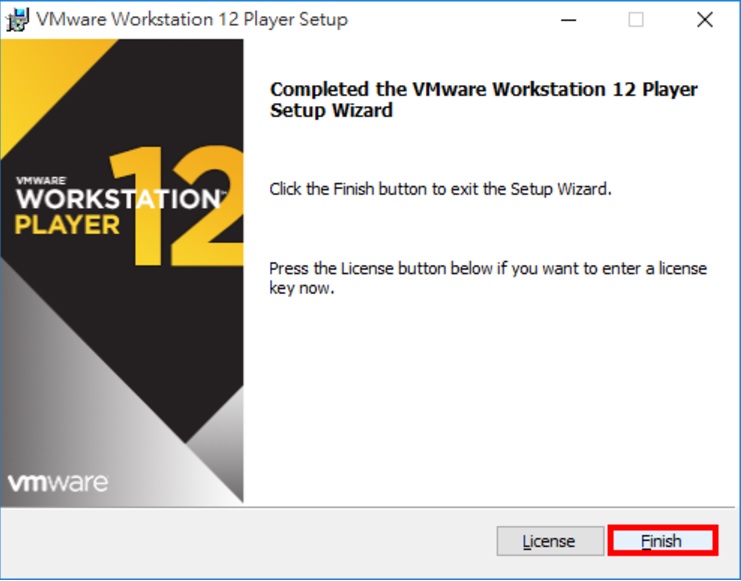
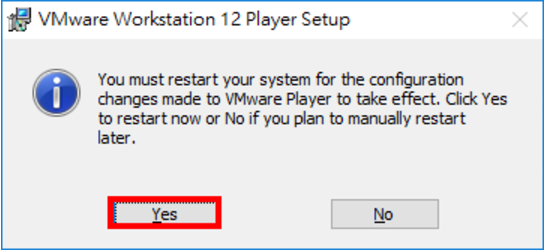
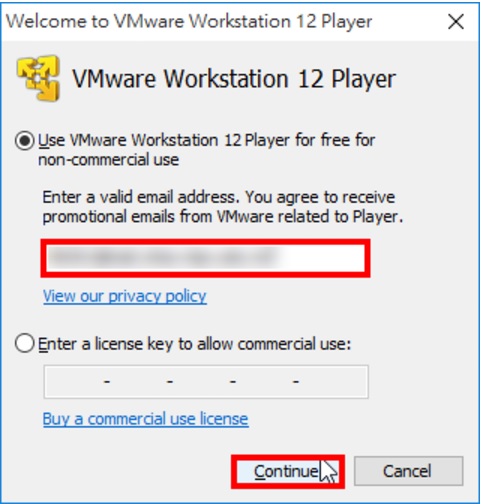


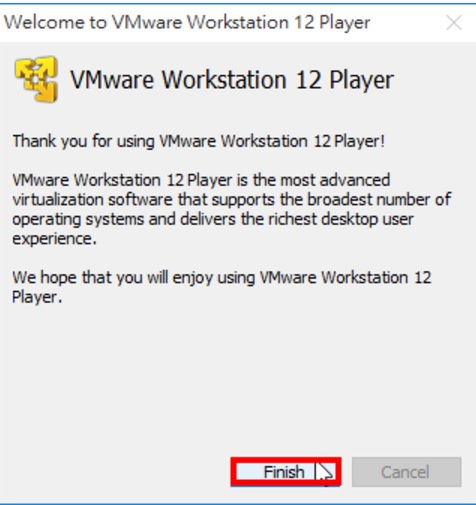
1. 開啟剛剛下載的VMware安裝檔案開始安裝點選next
2. 勾選”I accept the terms in the License Agreement”後按下Next。
3. 勾選”Enhanced Keyboard Driver”後按下Next。
4. 使用預設選項後按下Next。

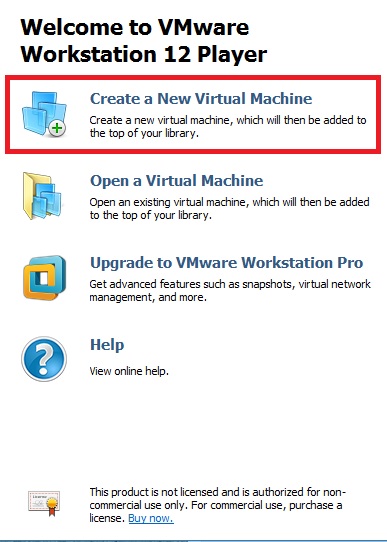
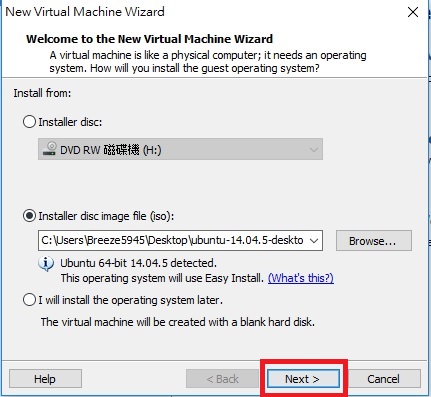
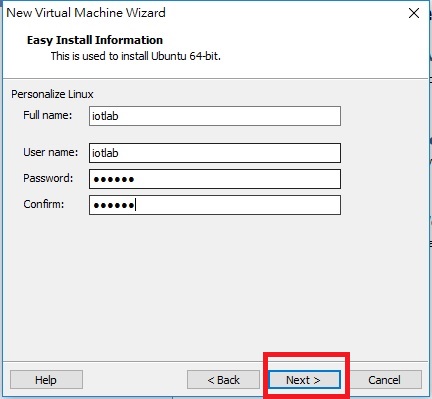
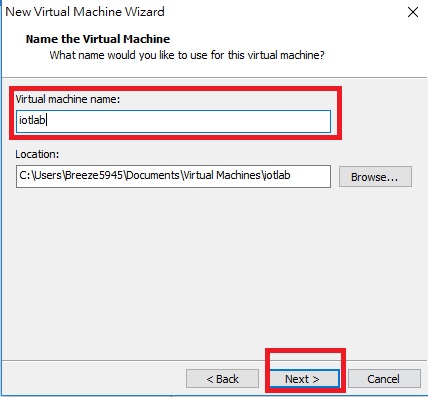
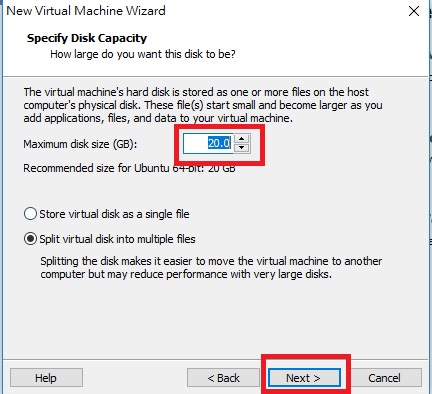
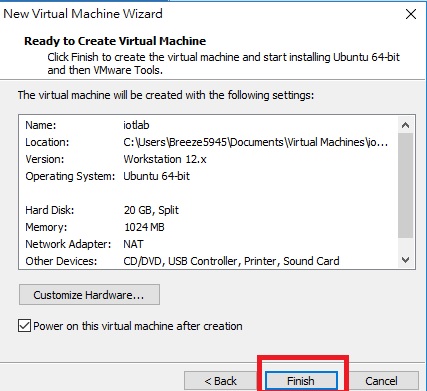
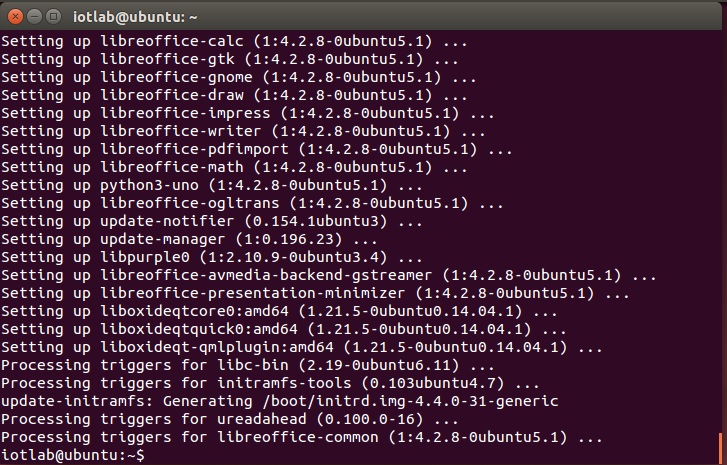
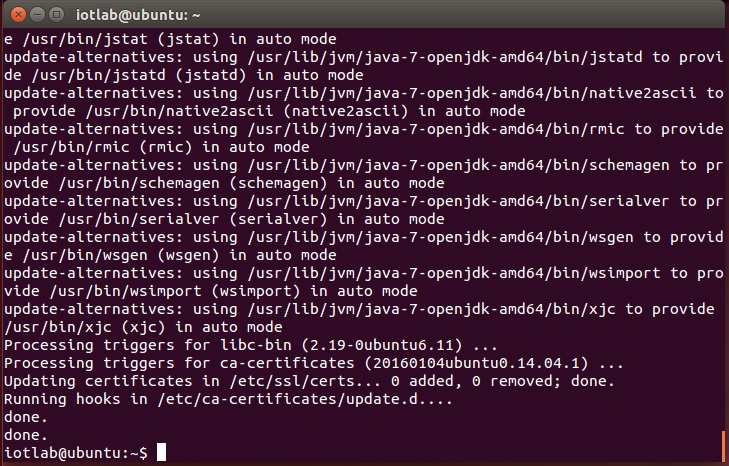
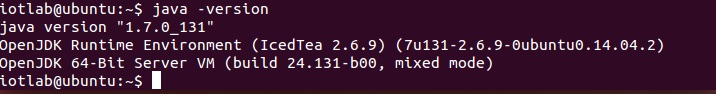
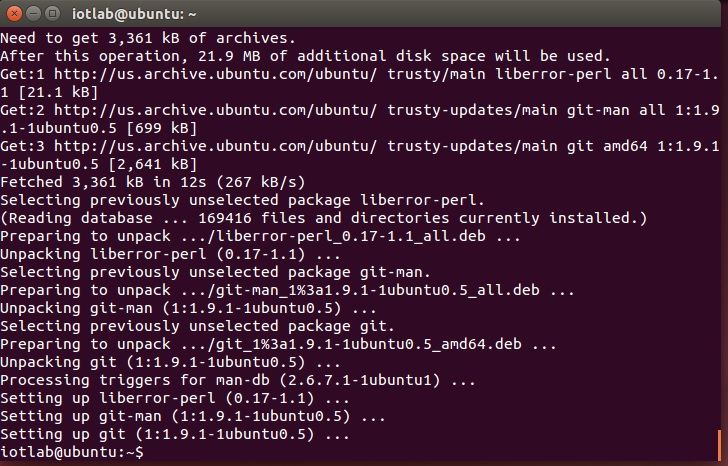
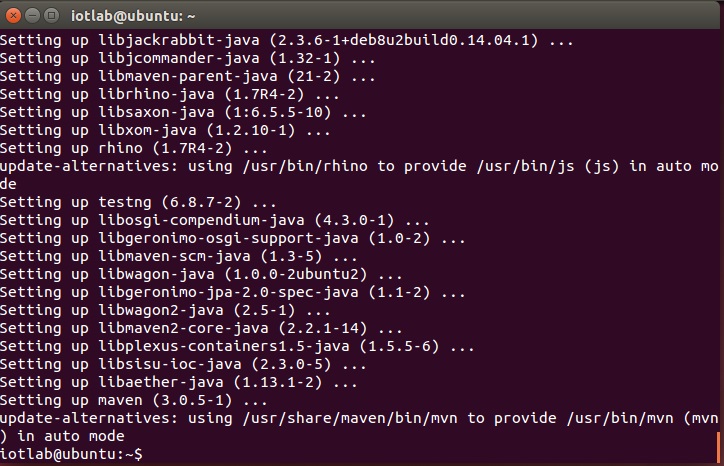
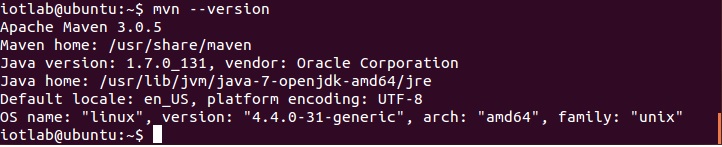
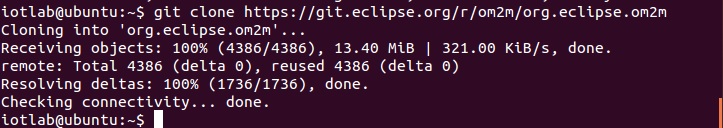
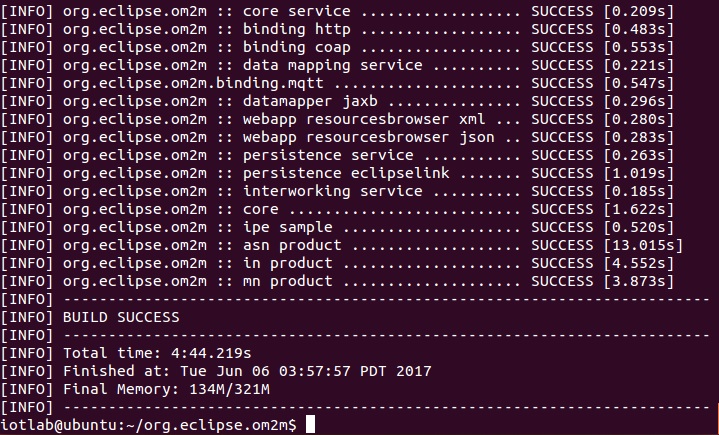
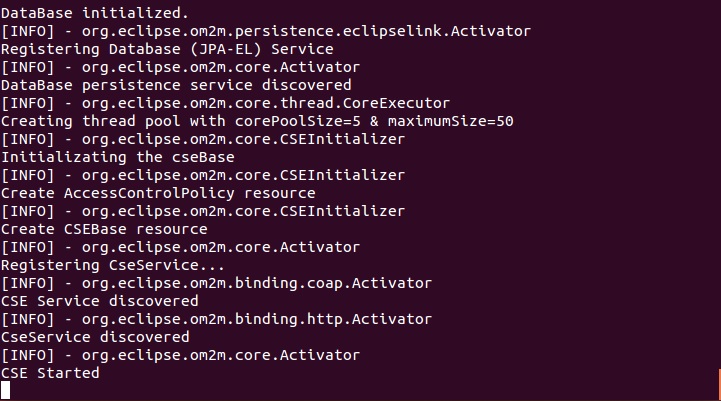
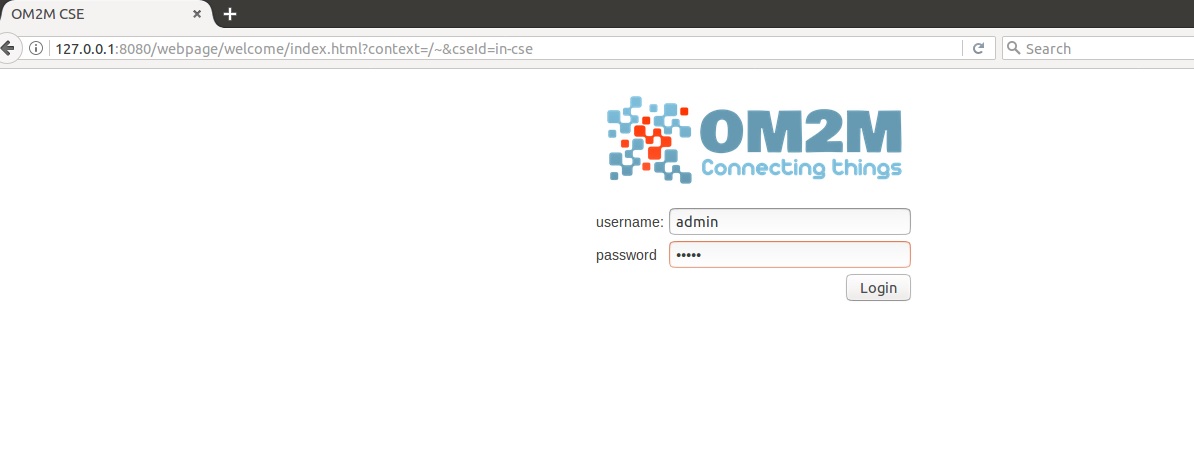
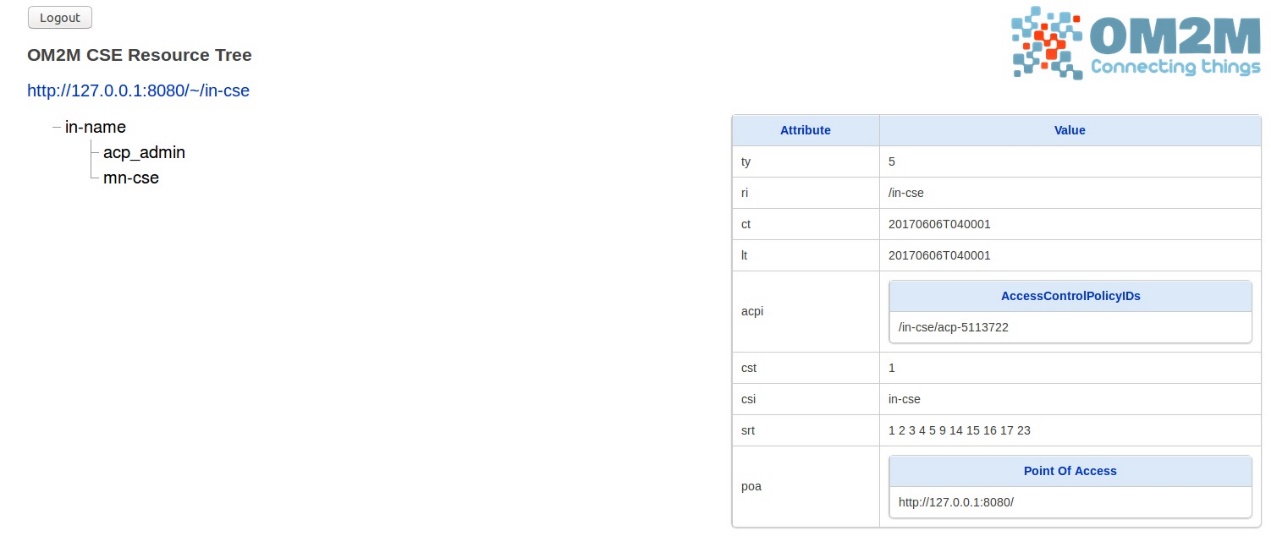
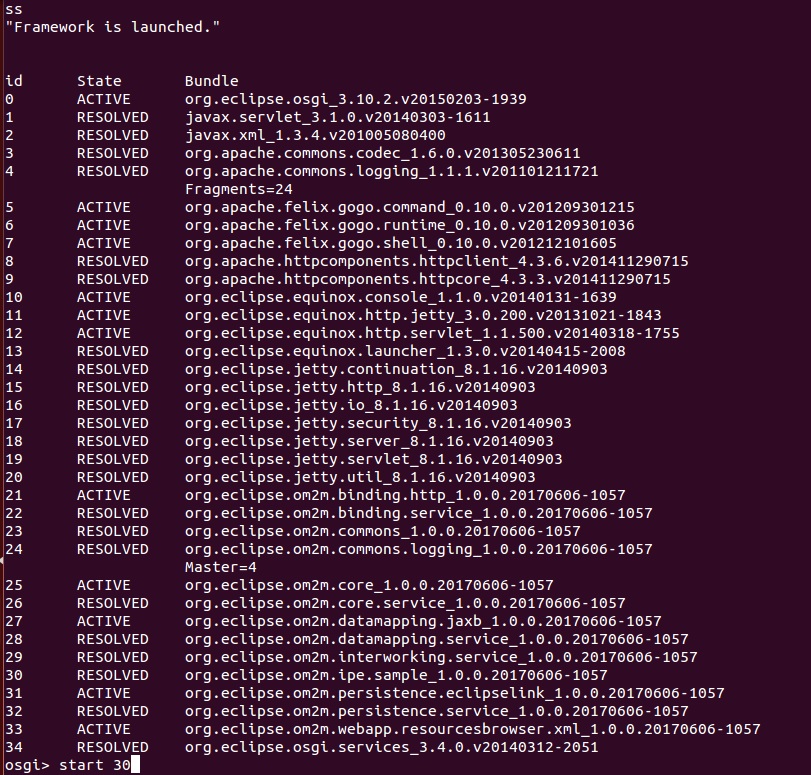
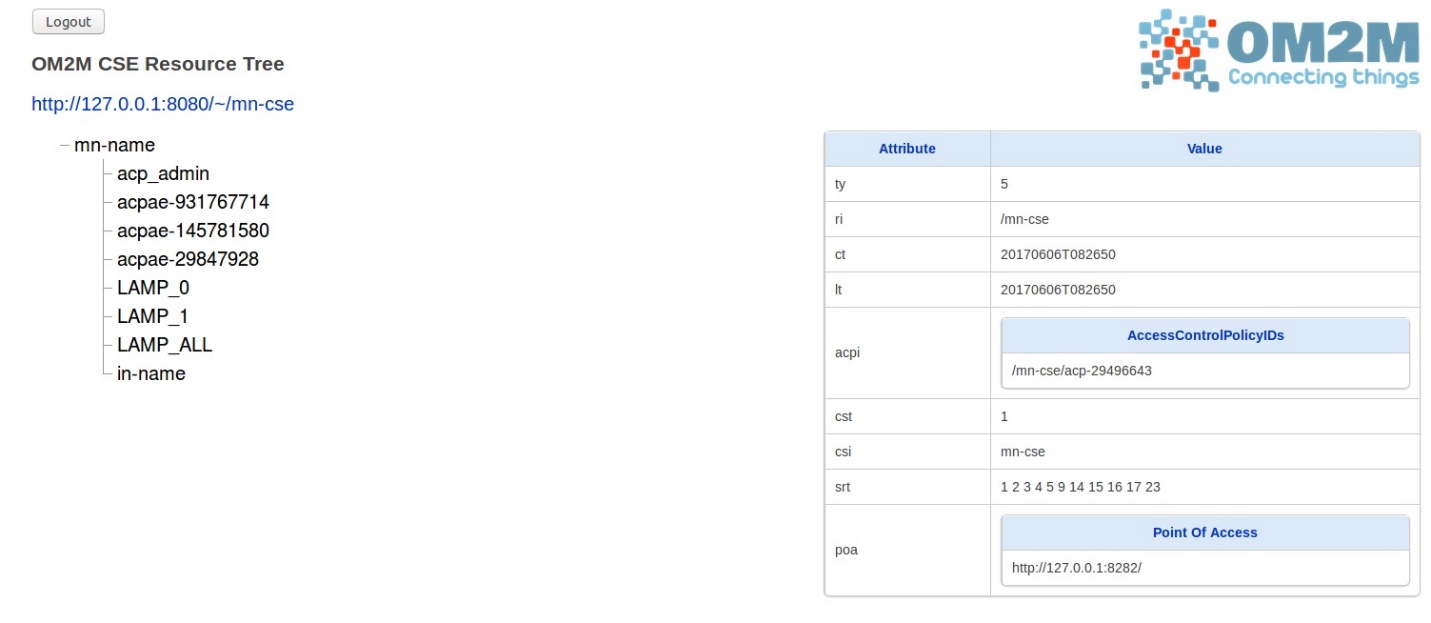
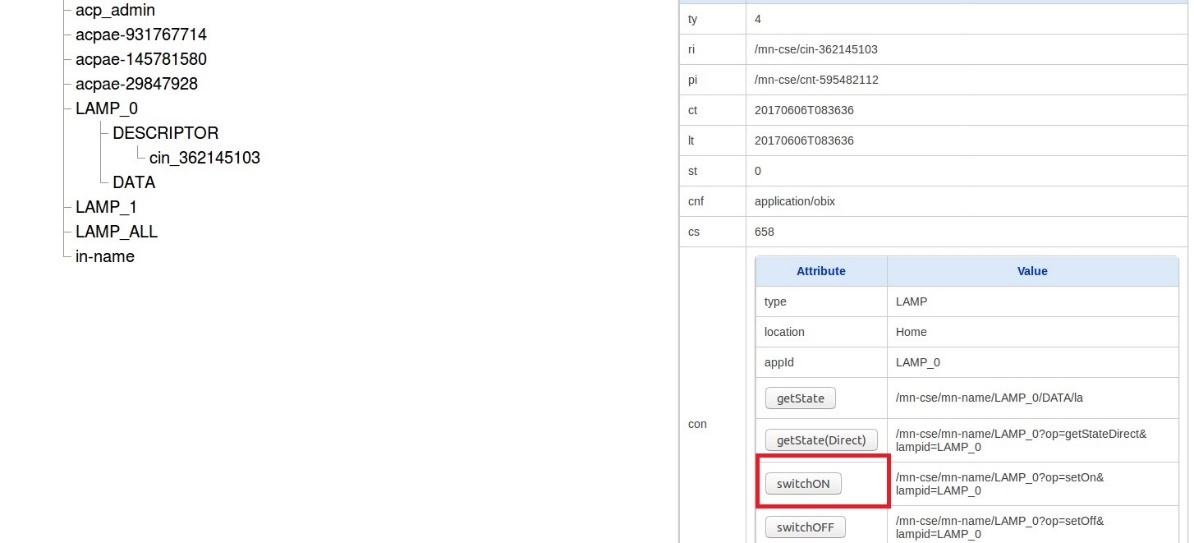
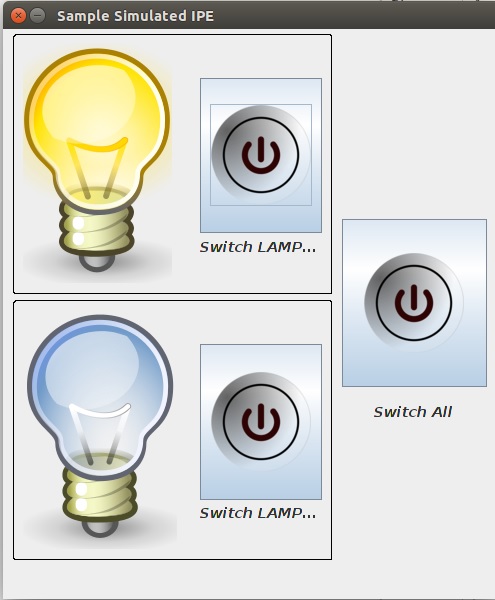
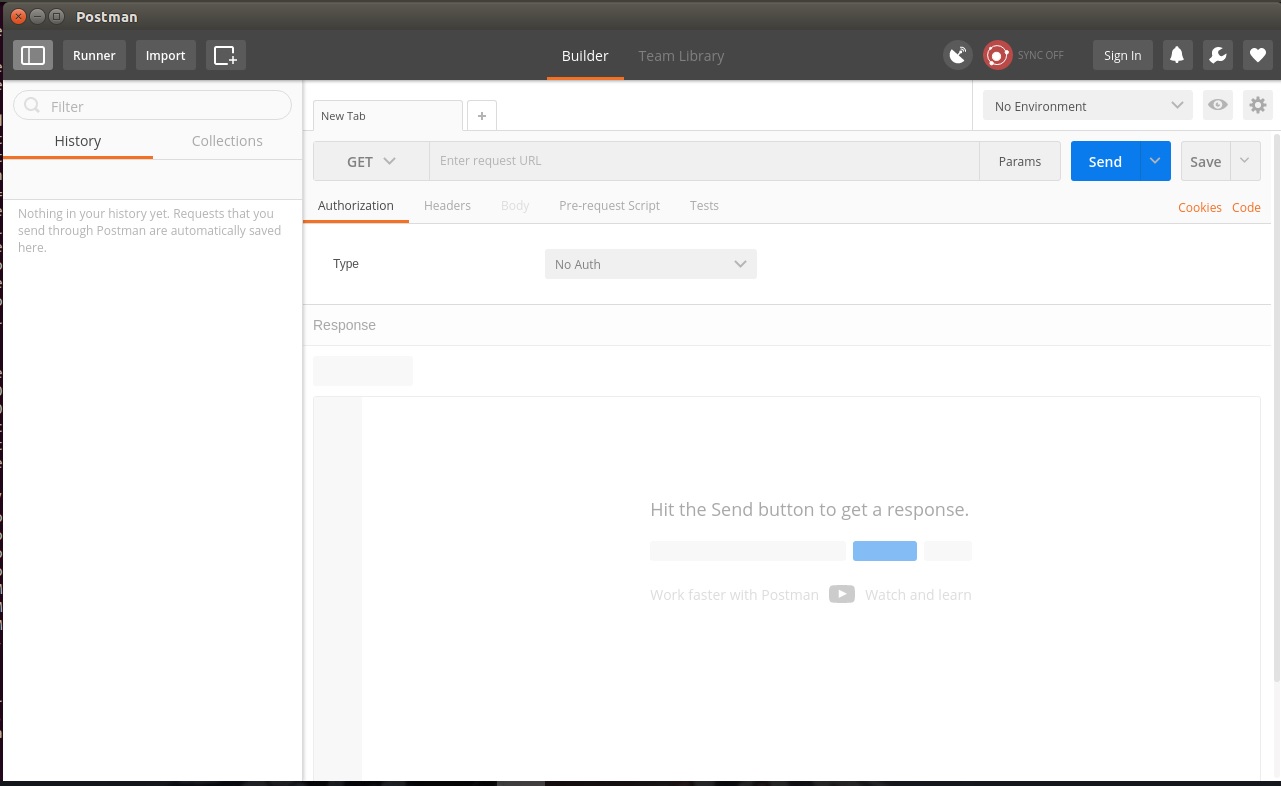
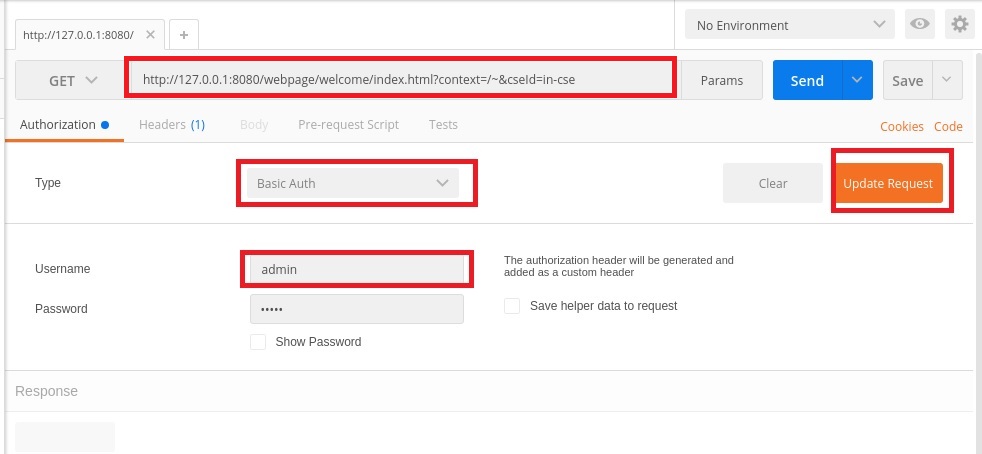
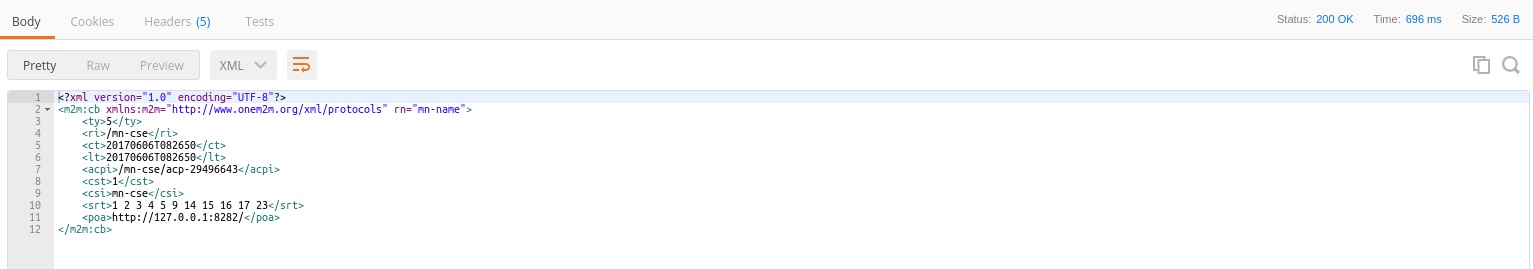
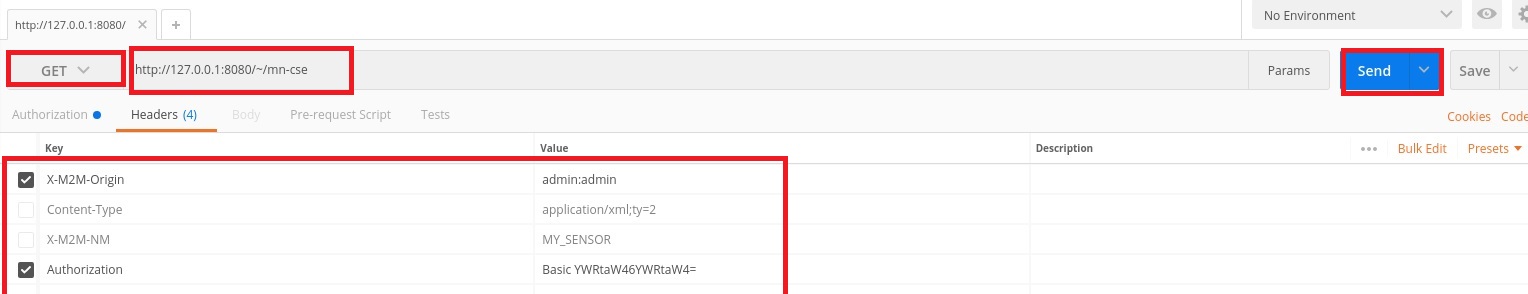


1. 使用預設選項後按下Next。
2. 點選Install



1. 完成後按下Finish
2. 點選Yes後重新開機。
3. 開啟VMware後選擇”Use VMware Workstation 12 Player for free for non-commercial use”，輸入Email點選Continue。
4. 按下Finish後結束並開始使用。



1. 開啟剛剛安裝好的VMware，點選” Create a New Virtual Machine”
2. 選擇中間的”Installer disc image file”並點選Browse選擇下載完成的Ubuntu系統的Image檔，按下Next。
3. 將Full name，Username、password、Confirm 填寫完成後，按下Next，以下欄位可自由填寫。
4. 填寫”virtual machine name”後按下Next。virtual machine name”可自由填寫。
5. 選擇Disk size，預設為20Gb，按下Next。
6. 按下Finish後開始安裝。
7. 安裝完成後輸入密碼進入系統桌面並開啟終端機以root身分執行Update和Upgrade。  
   執行: sudo apt-get update && sudo apt-get upgrade –y  
   此為完成之後之狀態圖。  
   
8. 安裝Java SDK 1.7  
   指令: sudo apt-get install openjdk-7-jdk  
   完成如下  
   
9. 輸入 java –version 檢查版本是否無誤。  
   
10. 安裝git。  
    指令: sudo apt-get install git –y  
    完成如下  
    
11. 安裝Maven。指令: sudo apt-get install maven –y  
    完成如下  
      
    檢查Maven版本 : mvn –version  
    
12. Git Clone Om2m one。  
    指令: git clone <https://git.eclipse.org/r/om2m/org.eclipse.om2m>  
    
13. 到剛剛clone完的org.eclipse.om2資料夾中執行 mvn clean install。完成如下  
    
14. 啟動Om2m one，  
    到org.eclipse.om2m/org.eclipse.om2m.site.in-cse/target/products/in-cse/linux/gtk/x86\_64裏頭執行start.sh，成功執行後會看到”CSE starteed”。  
    
15. In-cse啟動後到org.eclipse.om2m/org.eclipse.om2m.site.mn-cse/target/products/mn-cse/linux/gtk/x86\_64裏頭執行start.sh，成功執行後會看到mn-cse成功向in-cse註冊成功。  
    
16. Mn-cse成功向in-cse註冊成功後，可以開啟網頁瀏覽器在網址列打上http:127.0.0.1:8080/webpage開啟Om2m的頁面。帳號密碼皆為admin。
17. 登入之後可以看到在in-cse下有mn-cse存在。
18. 在啟動mn-cse終端機下達ss可以看到目前提供哪些服務 輸入start 30的指令可以開啟om2m內建的燈泡sample。
19. 啟動之後可以看到mn-cse出現了LAMP\_0、LAMP\_1及LAMP\_ALL。
20. 點選其中一個LAMP再點選DESCRIPTOR出現cin\_random\_number點選之後可以透過出現的按鈕控制燈泡發亮。  
      
      
    
21. Postman建置及新增onem2m Application
22. 接著安裝等等需要的Postman套件。打開瀏覽器輸入Postman後點選的一個連結，並點選Linux x64開始下載。
23. 下載完成後切換到Download目錄進行解壓縮  
    tar zxvf Postman-linux-x64-4.11.1.tar.gz，解壓縮完成後進入目錄，執行./Postman即可開啟Postman。
24. 啟動Postman後，在URL輸入in-cse的URL後在選擇Type選擇Basic Auth輸入帳號密碼後按下Update Request。
25. Retrieve a resource，在URL輸入<http://127.0.0.1:8282/~/mn-cse>  
    Method選用GET，在Header中輸入X-M2M-Origin : admin:admin並按下Send。  
      
    得到回應會是
26. 新增Appliccation在Mn-cse中:

在URL中輸入<http://127.0.0.1:8080/~/mn-cse>，並在

Header輸入:  
X-M2M-Origin: admin:admin  
Content-Type: application/xml;ty=2  
X-M2M-NM: MY\_SENSOR(可任意輸入)  
在Body中選擇raw型態選擇XML(application/xml)並輸入以下

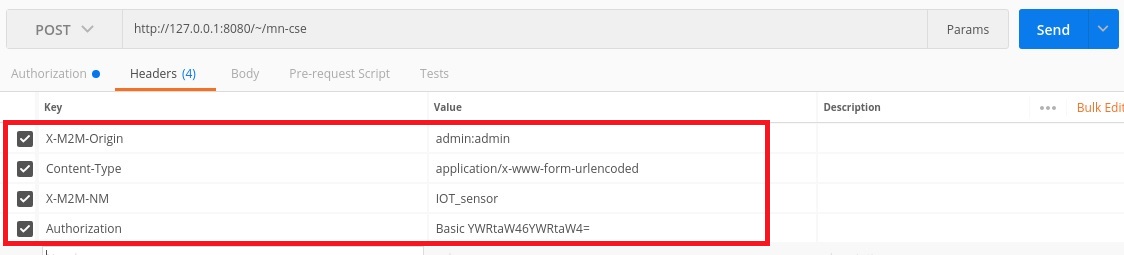
<om2m:ae xmlns:om2m="http://www.onem2m.org/xml/protocols">

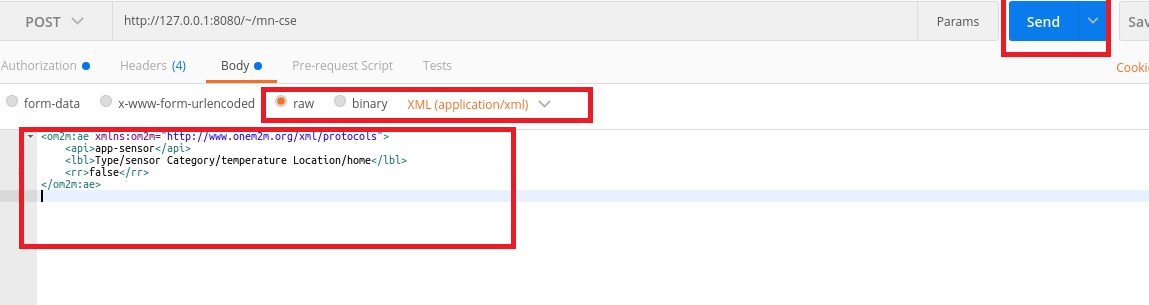
<api>app-sensor</api>

<lbl>Type/sensor Category/temperature Location/home</lbl>

<rr>false</rr>

</om2m:ae>按下Send送出即可。





1. 完成Send之後打開Om2m的頁面並到mn-cse的頁面中發現新增IOT\_sensor成功。



Lab3-1: 請使用Postman新增一個Sensor名字隨意並在Sensor下創造

* + - 1. DESCRIPTOR container
      2. Description contentInstance
      3. DATA container
      4. Data contentInstance

完成應如下圖所示，在MY\_SENSOR下會出現DESCRIPTOR及DATA又各自包含各自的contentInstance。



Part 2 Node-Red介紹與環境建置

1. Node-Red介紹

Node-RED 是由IBM Emerging Technology所製作，是一款基於Node.js並應用在物連網的視覺化套件，裡面整合了許多的服務，甚至能擴充IBM Bluemix的雲端服務API，原本是專門為了物聯網而設計，但現在不論是http建網站、websocket、爬蟲等等…都有包辦! Node-RED安裝完成後，我們就可以在任何電腦的瀏覽器上編輯我們的”node flows”。

Node-RED是利用瀏覽器的介面，來開發IOT的工具，其開發流程已flow為基礎。Node-RED流行的理由，在於它擁有以下諸多的優點：

1. 相較於撰寫程式碼，使用圖形化界面可更快速開發。
2. 適合作為概念性應用驗證的開發工具。
3. 淺顯易懂，每個API都有文字說明可供查詢。

具有高度擴充性，已有大量的Library、Flow、Node供人下載。

1. 環境建置

為了要使用Node-red，需要完成以下兩個步驟:

1. 安裝Node.js。
2. 安裝Node-red。

* 安裝Node.js

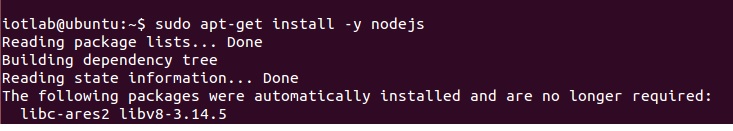
Step1:

到終端機輸入:

curl -sL https://deb.nodesource.com/setup\_4.x | sudo -E bash -



Step2:輸入 sudo apt-get install -y nodejs



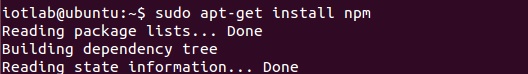
Step3:

完成之後輸入node –v 檢查版本。



* **安裝node-red。**

Step 1.:安裝npm，打開終端機輸入 sudo apt-get install npm



Step 2.:完成後輸入node –v 檢查版本。

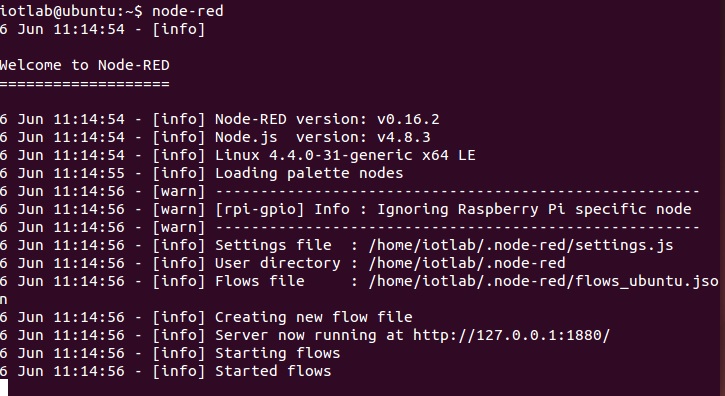


Step3.:

安裝node-red，在終端機輸入

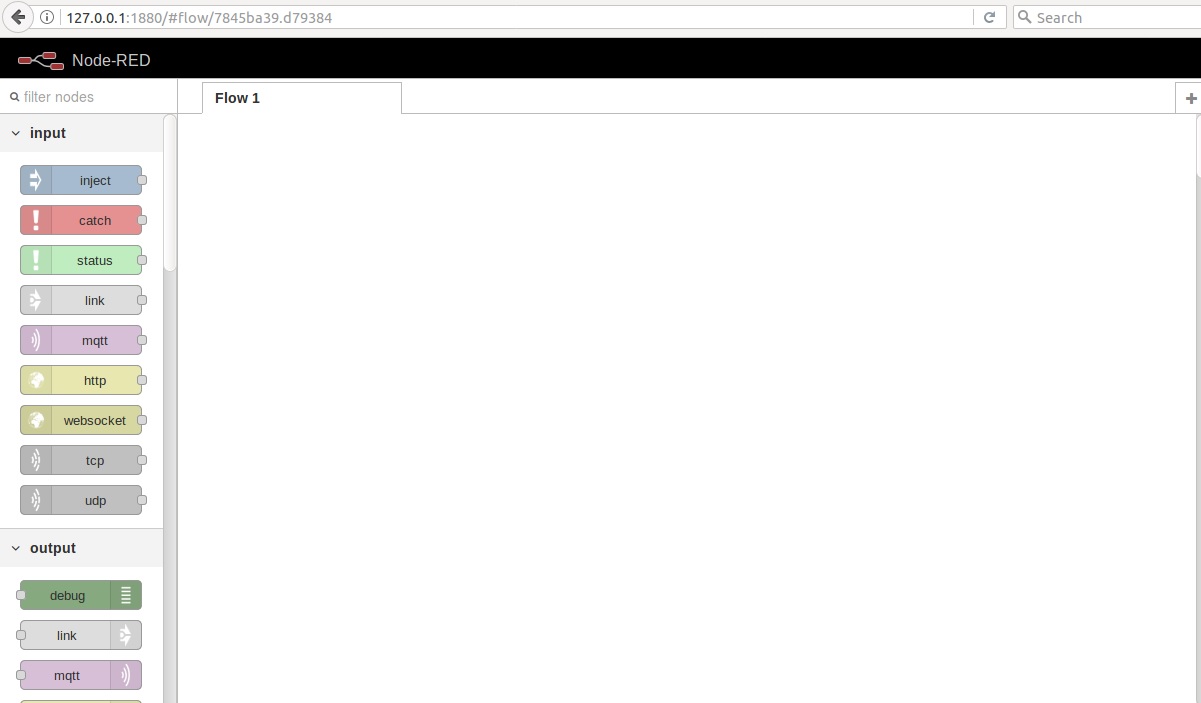
sudo npm install -g --unsafe-perm node-red node-red-admin

完成安裝後輸入: node-red啟動node-red



Step5:

在瀏覽器輸入<http://127.0.0.1:1880>開啟Node-red網頁。

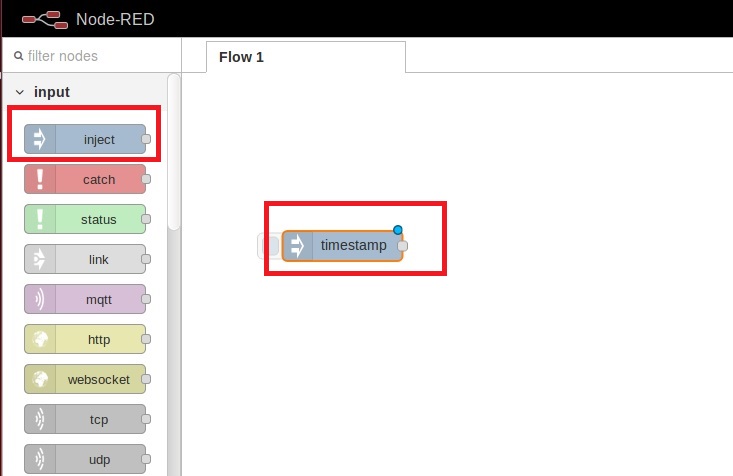


1. Node-red Example

使用Node-red在Mn-cse中新增一個Application

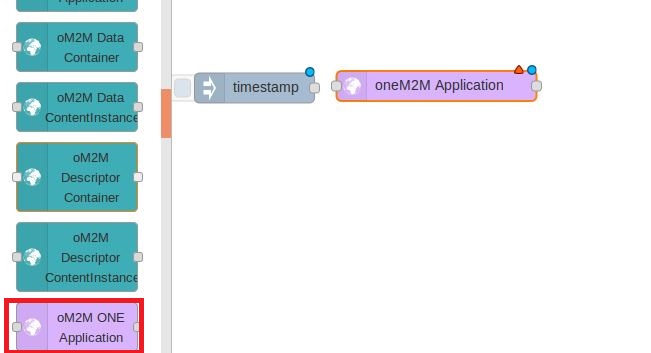
Step1:

從左邊的 ”input” library拉出”Inject”object到Flow1



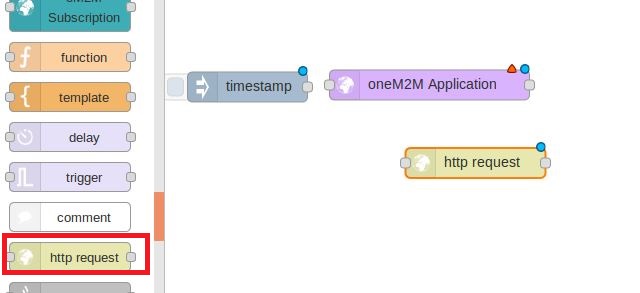
Step2:

選取function中的oM2M one Application



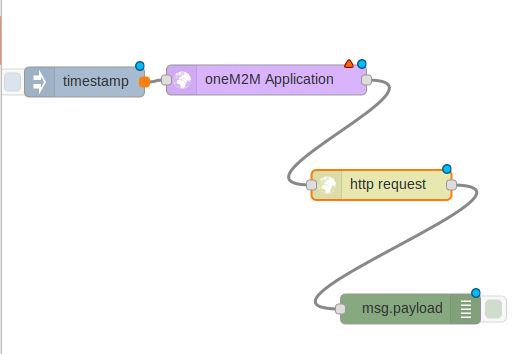
Step 3:

選取function中http request



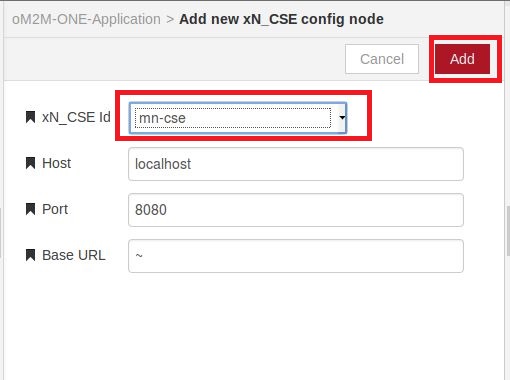
Step 4:

選取output中的debug，並將剛剛選取的元件連結起來。

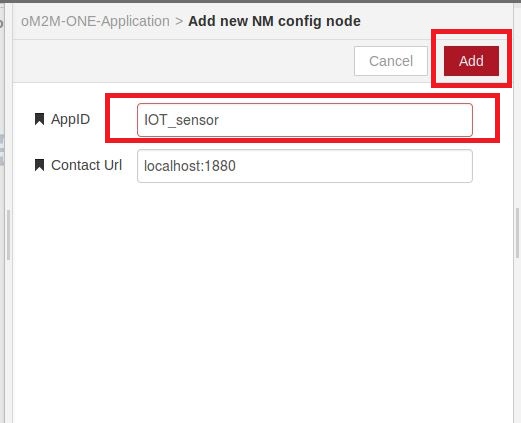


Step 5:

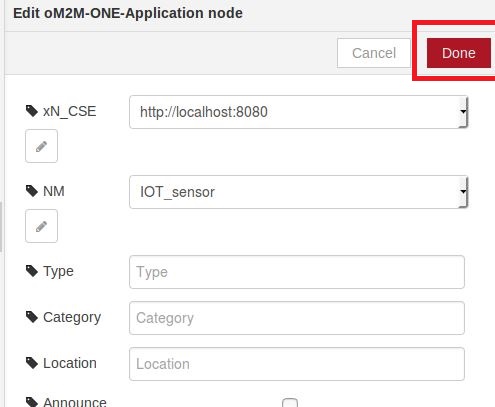
雙擊oneM2M Application打開編輯頁面進行編輯。並且對xN\_CSE及NM點選編輯動作。



此處的AppID可以任意填寫。

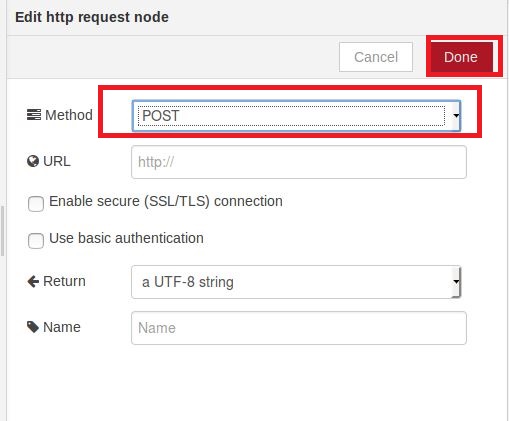


完成之後按下Done。



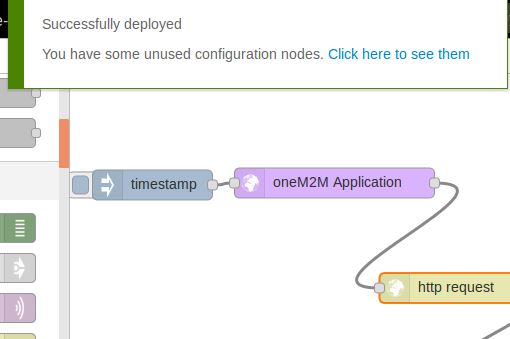
Step 6:

雙擊http request進行編輯，選取POST並按下Done。



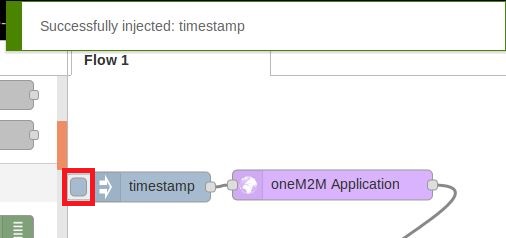
Step 7:

完成後按下上方Depolyed鍵可看到成功訊息。



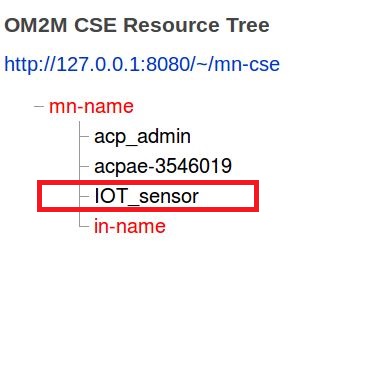
Step 8:

按下紅色框框所框起之地方已啟動。



Step 9:

成功新增了Application在mn-cse中。



LAB3-2 使用Node-red新增一個sensor包含

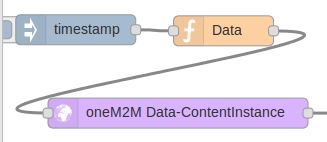
1. DESCRIPTOR containe、Description contentInstance、DATA container

Data contentInstance。

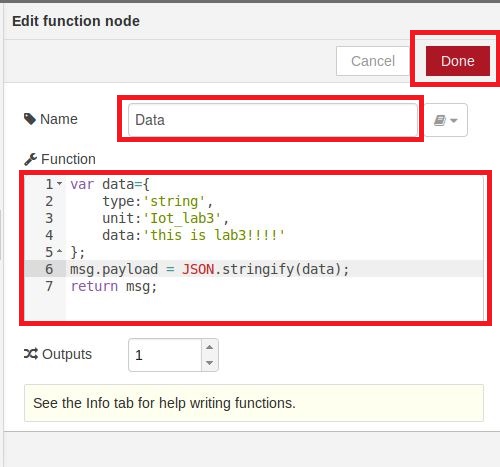
1. 透過Http接收Postman所傳之值顯示在Node-red所新增之sensor上。

關於Data使用方式如下:

選取function中的funtion元件，並且進行編輯，名字可隨意取名。

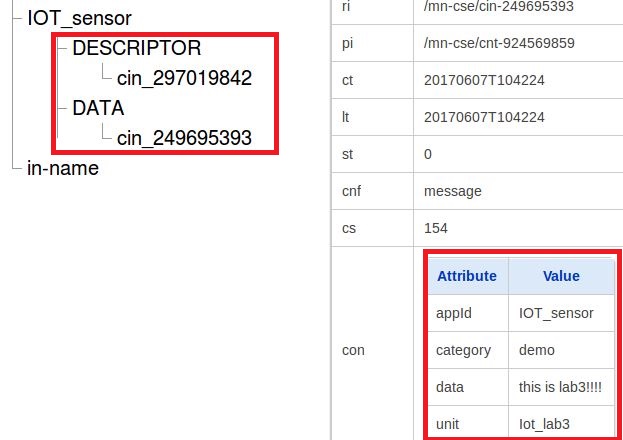


Unit 及 data 內容也可進行更動。



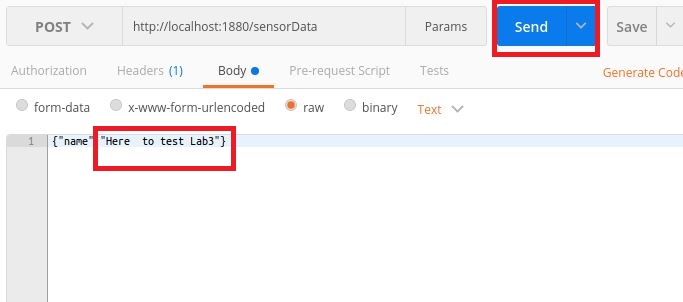
完成之後如下

1.



2.

透過Postman傳值。



顯示成功於Node-red所新增之Application

