使用LINE機器人(LINE BOT)之自動即時資安通報系統設計

國立中興大學計算機及資訊網路中心

報告人：陳仕豪

2018/1/29

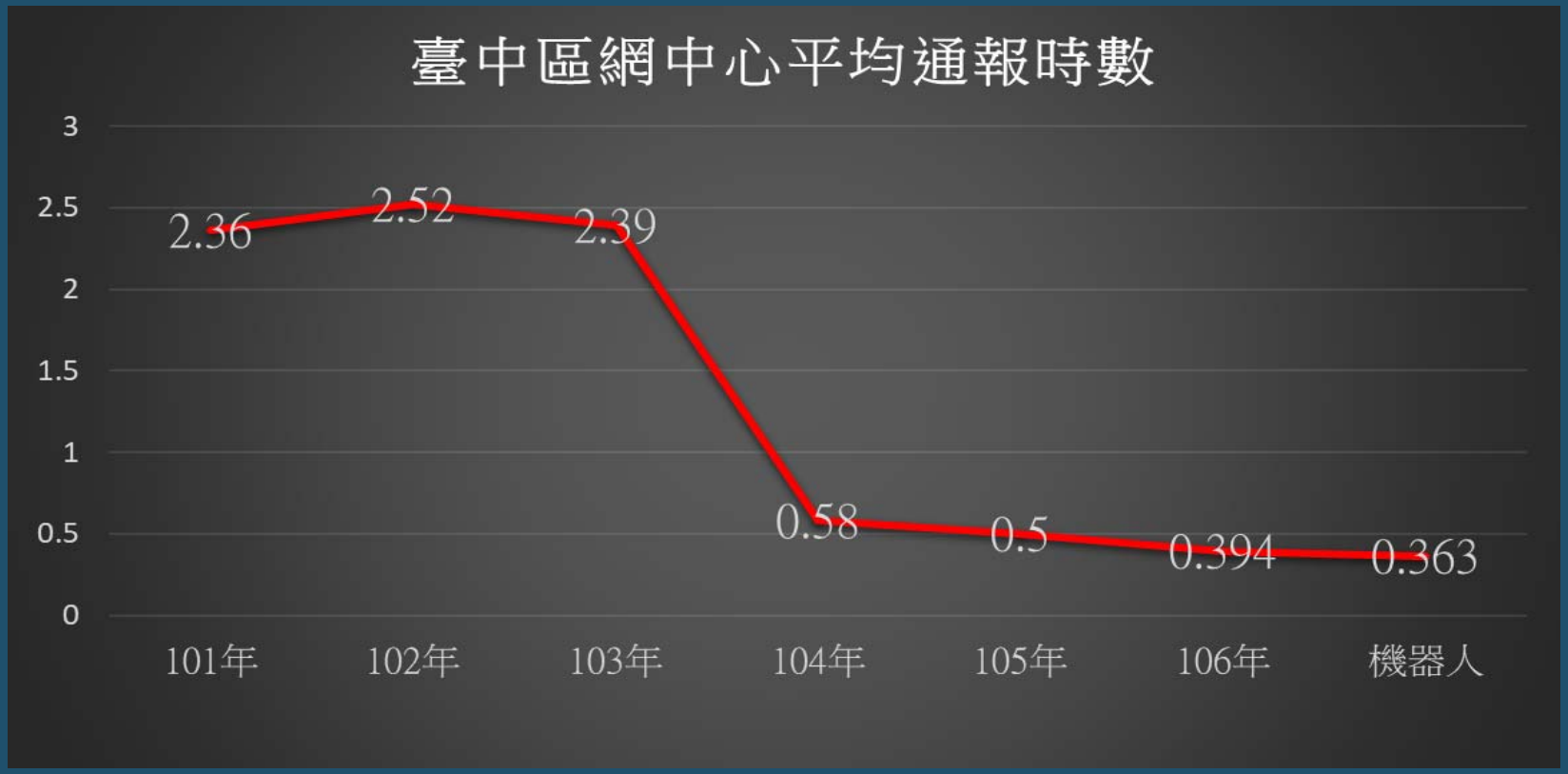
- 緣起與目的
- 關於 LINE BOT
- LINE BOT 設定與開發
- 使用 LINE BOT 於資安通報
- 系統運行成效

緣起與目的

- 教育部資科司建立一網路資安事件通報平台，提供網路資安事件的應變處理通報。
- 事件通知係利用電子郵件以及簡訊，缺點是不具即時性，導致網路資安事件因為處理的延遲而造成損害擴大。

緣起與目的

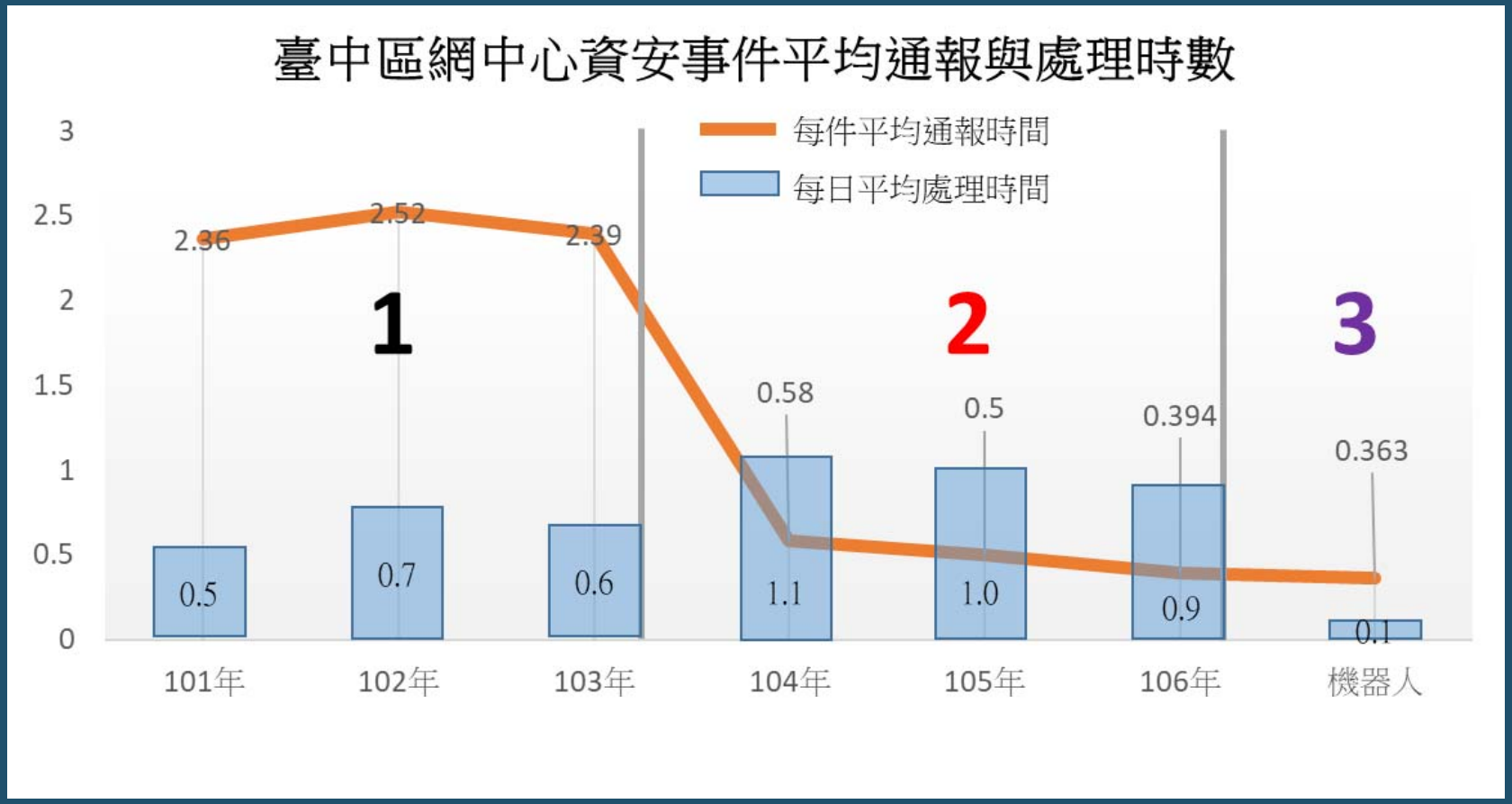
- 台中區網中心在 2015 年即利用手機通訊軟體 **LINE** 將下轄連線單位主要聯絡人組成一個聯絡群組，在網路資安事件發生時，區網中心資安負責人於台中區網 **LINE** 群組發布事件訊息，提醒相關單位負責人注意。



緣起與目的

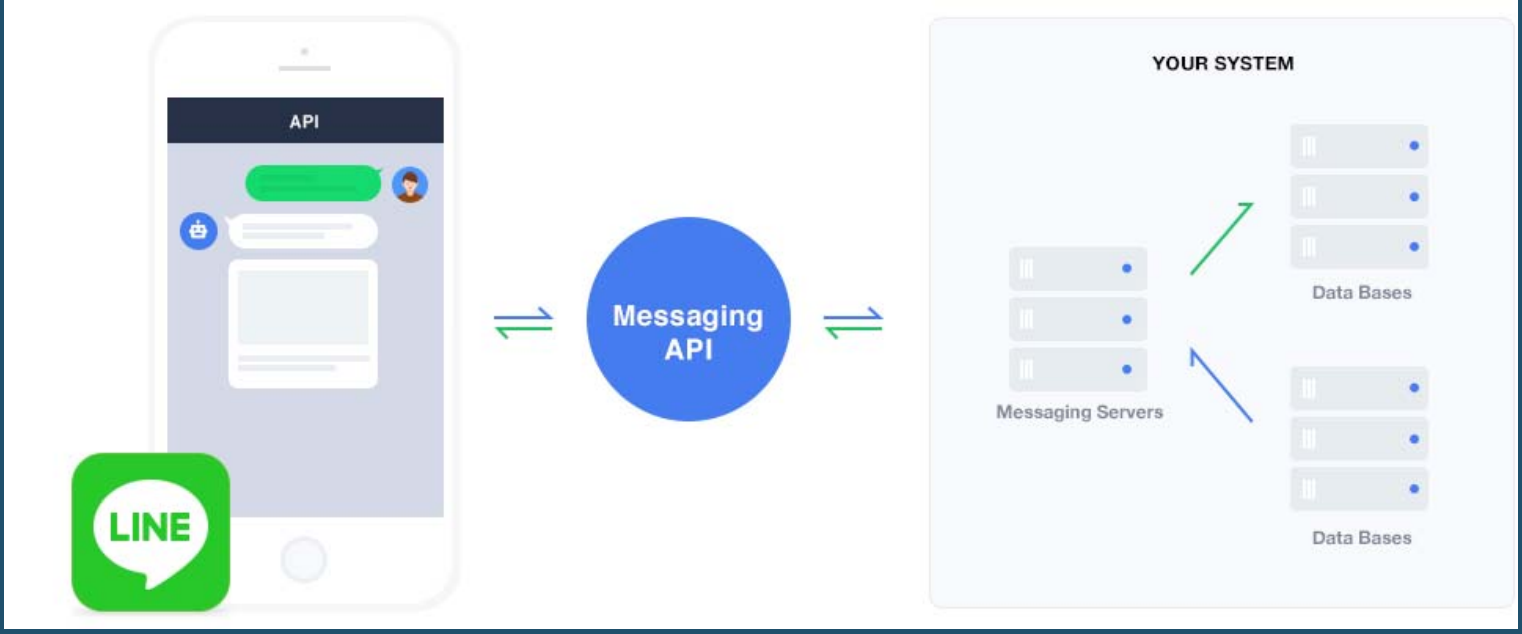
- 由區網中心資安負責人採用人工方式將相關訊息輸入、發布於 LINE 群組，仍無法達到完全自動化的目標。

系統運行成效


臺中區網中心資安事件平均通報與處理時數
每件平均通報時間
每日平均處理時間
3
2.5
2
1.5
1
0.5
0
2.36
2.52
2.39
0.58
0.5
0.394
0.363
0.5
0.7
0.6
1.1
1.0
0.9
0.1
1
2
3
101年
102年
103年
104年
105年
106年
機器人

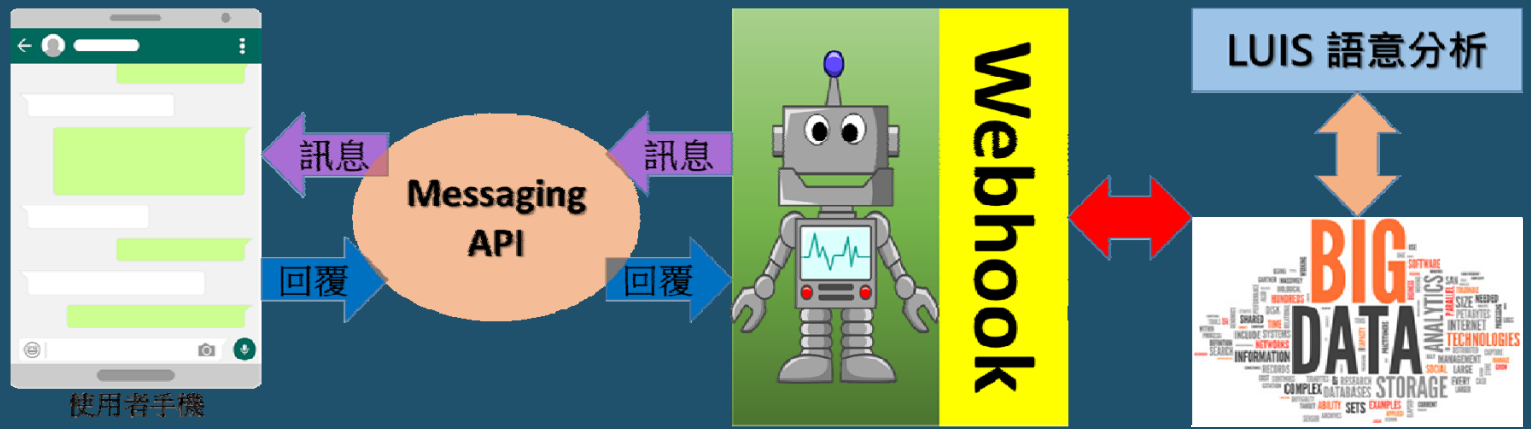
關於 LINE BOT

- 是一項能夠讓 **LINE** 用戶與您所提供的服務進行雙向溝通的功能。
- 新版叫做 **Messaging API**。
- **Messaging API** 將透過 **LINE** 伺服器，在您的服務伺服器與 **LINE** 應用程式間互相收發訊息。



關於 LINE BOT

- 所有從用戶傳送給 LINE BOT 的訊息，就會傳送到 **Webhook**，而你透過程式碼寫的這個 Webhook，在接收到這個訊息之後，就可以依照用戶的訊息內容，來回應(回覆)不同的訊息給用戶。這就是一個 LINE 對談器人最基本的架構：

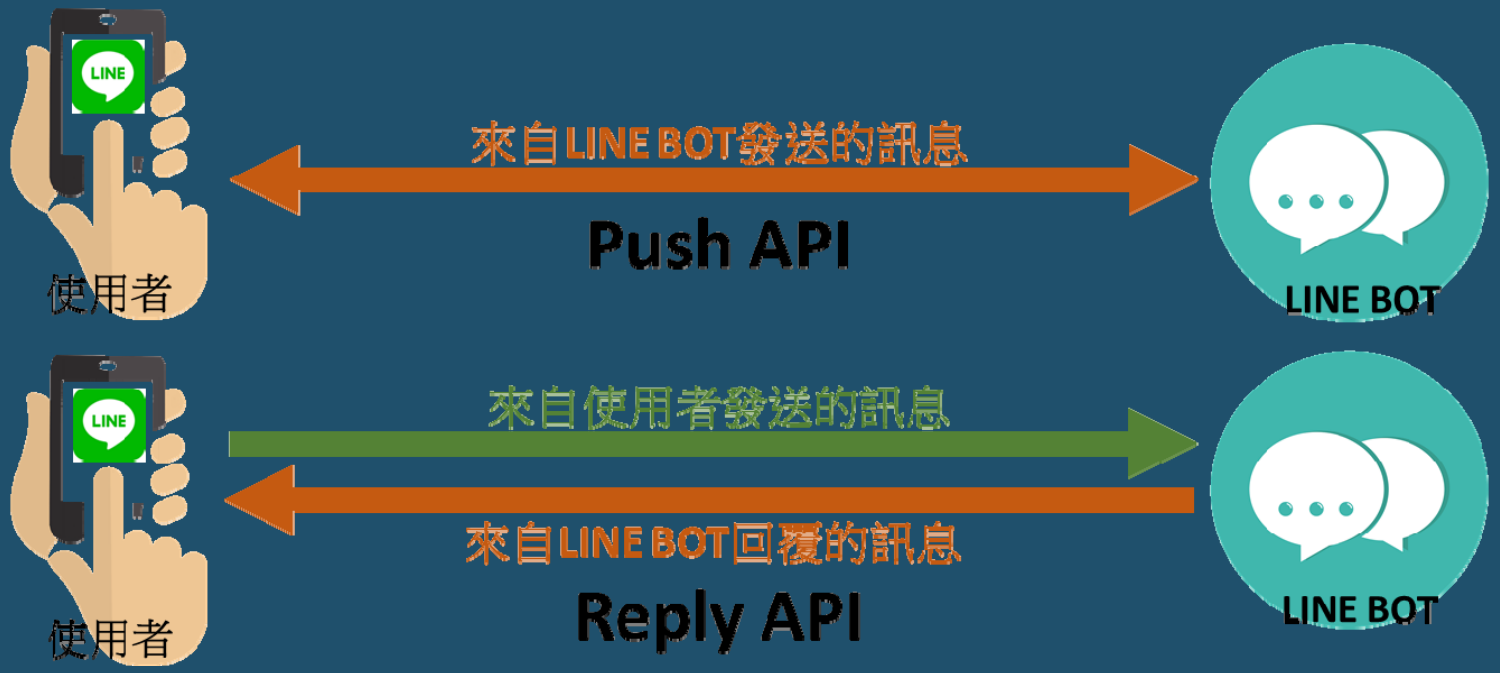


Webhook 是一種 Http 推送 API，為所建立的應用服務提供即時回覆訊息的一種方法。當 Webhook 被觸發後，會將回覆內容傳送到應用程式端去，這樣使用者可以立即獲得回覆訊息。

關於 LINE BOT

- Push API 與 Reply API
- Push API 指的是 BOT 能夠在任何時間點主動對用戶傳送訊息的 API。
- Reply API 指的是 BOT 可以針對用戶傳來的訊息進行回覆的 API。

我們用這個！



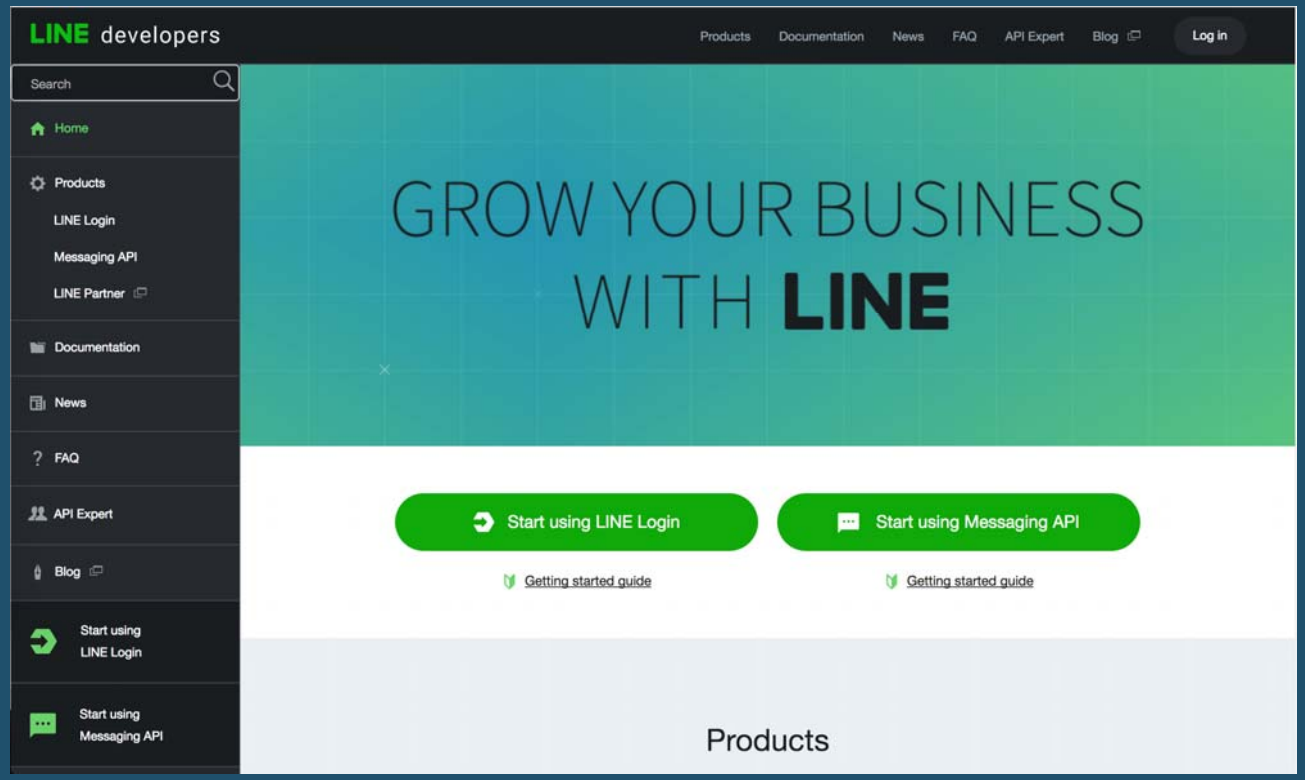
關於 LINE BOT

- LINE BOT 付費方案 (https://at.line.me/tw/plan)

		Developer Trial	免費版	入門版	進階版	進階版 (API)	專業版	專業版 (API)
費用	設定費	免費	免費	免費	免費	免費	免費	免費
	月費	免費	免費	798元	1,888元	3,888元	5,888元	8,888元
目標好友數	目標好友數	50	無上限	20,000	50,000	50,000	80,000	80,000
每月群發訊息則數	群發訊息傳送數量	無上限	每月1,000則以內	無上限	無上限	無上限	無上限	無上限
每月主頁投稿數	動態主頁投稿數量	無上限	每月10則以內	無上限	無上限	無上限	無上限	無上限
Reply API		○	○	○	×	○	×	○
Push API		○	×	×	×	○	×	○

關於 LINE BOT

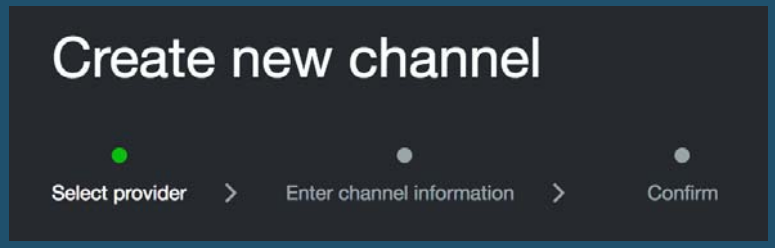
- 申請 LINE BOT (https://developers.line.me/en/)

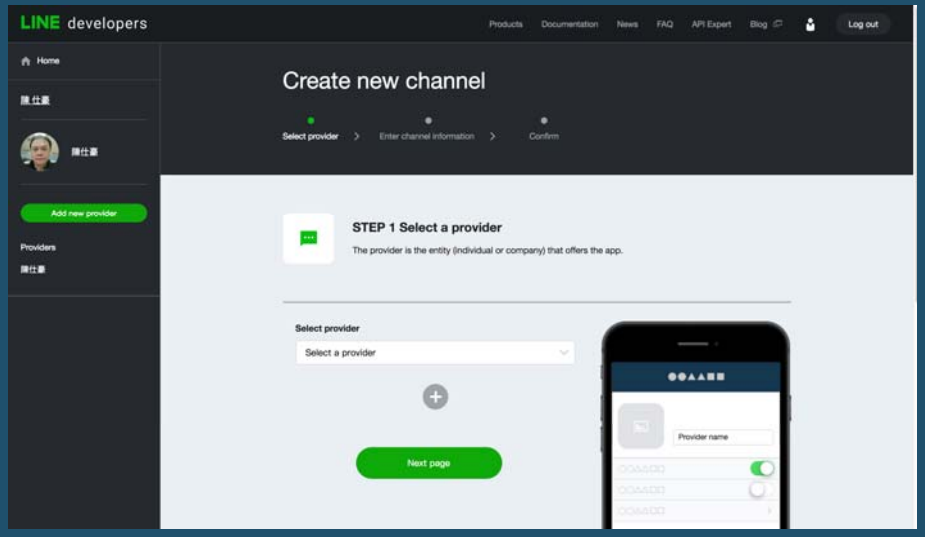


點選右邊的『 Start using Message API』，接下來系統將會要求使用者完成三個設定步驟來完成新增一個通道 (LINE BOT)。

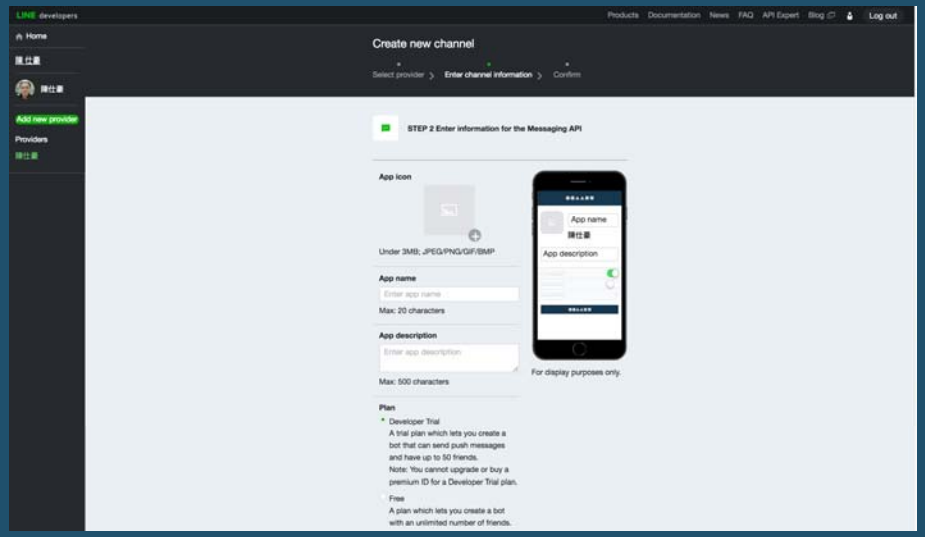
關於 LINE BOT

- 申請 LINE BOT (https://developers.line.me/en/)





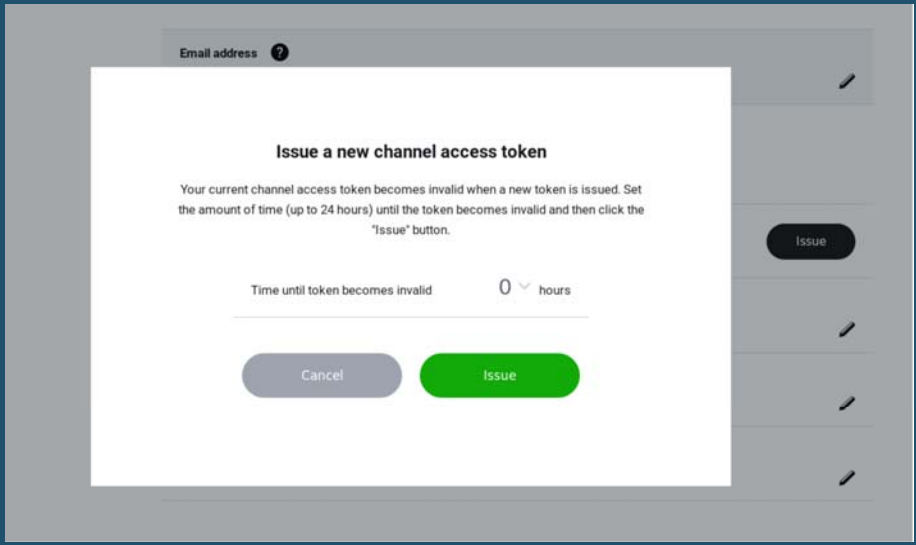
選取一個『提供者』，也就是告訴系統是誰提供這個 LINE BOT。



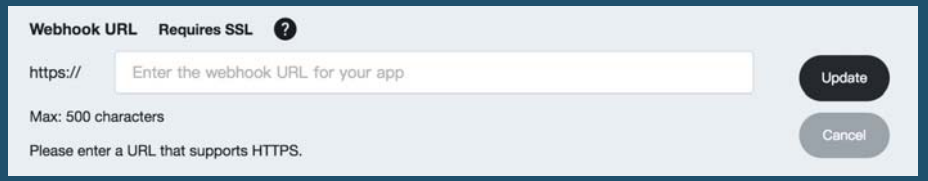
接下來，填上這個 Messaging API (LINE BOT) 的基本資料。

關於 LINE BOT

- 申請 LINE BOT (https://developers.line.me/en/)



點選『Channel access token (long-lived)』的 Issue，產生一個 Token。
這就是發送訊息所需的『令牌』。



設定我們開發的 Webhook 服務的網址。

Webhook 需要支援 SSL 的網站

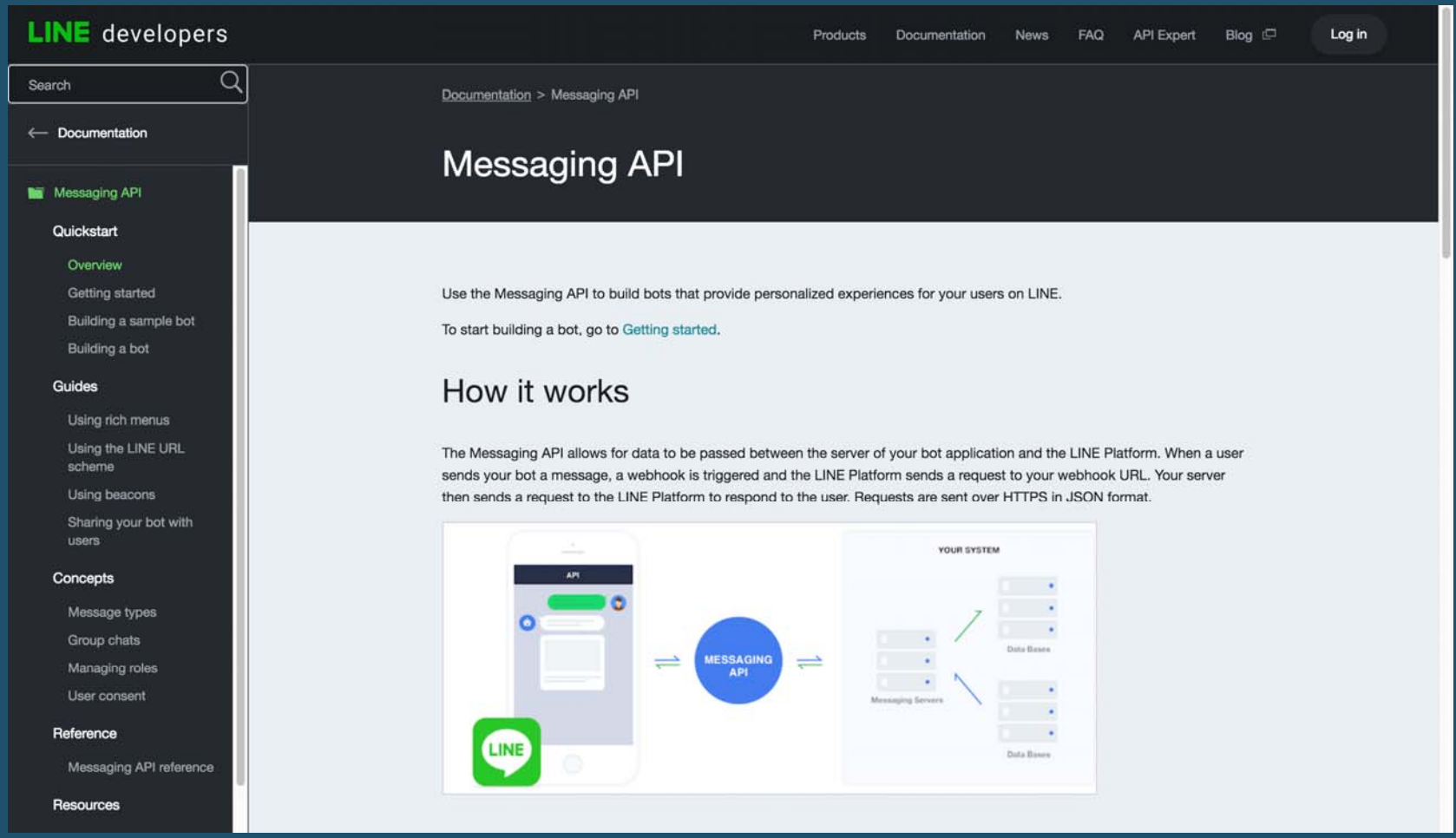
關於 LINE BOT

- 申請 LINE BOT (https://developers.line.me/en/)

記得打開以下兩項設定！

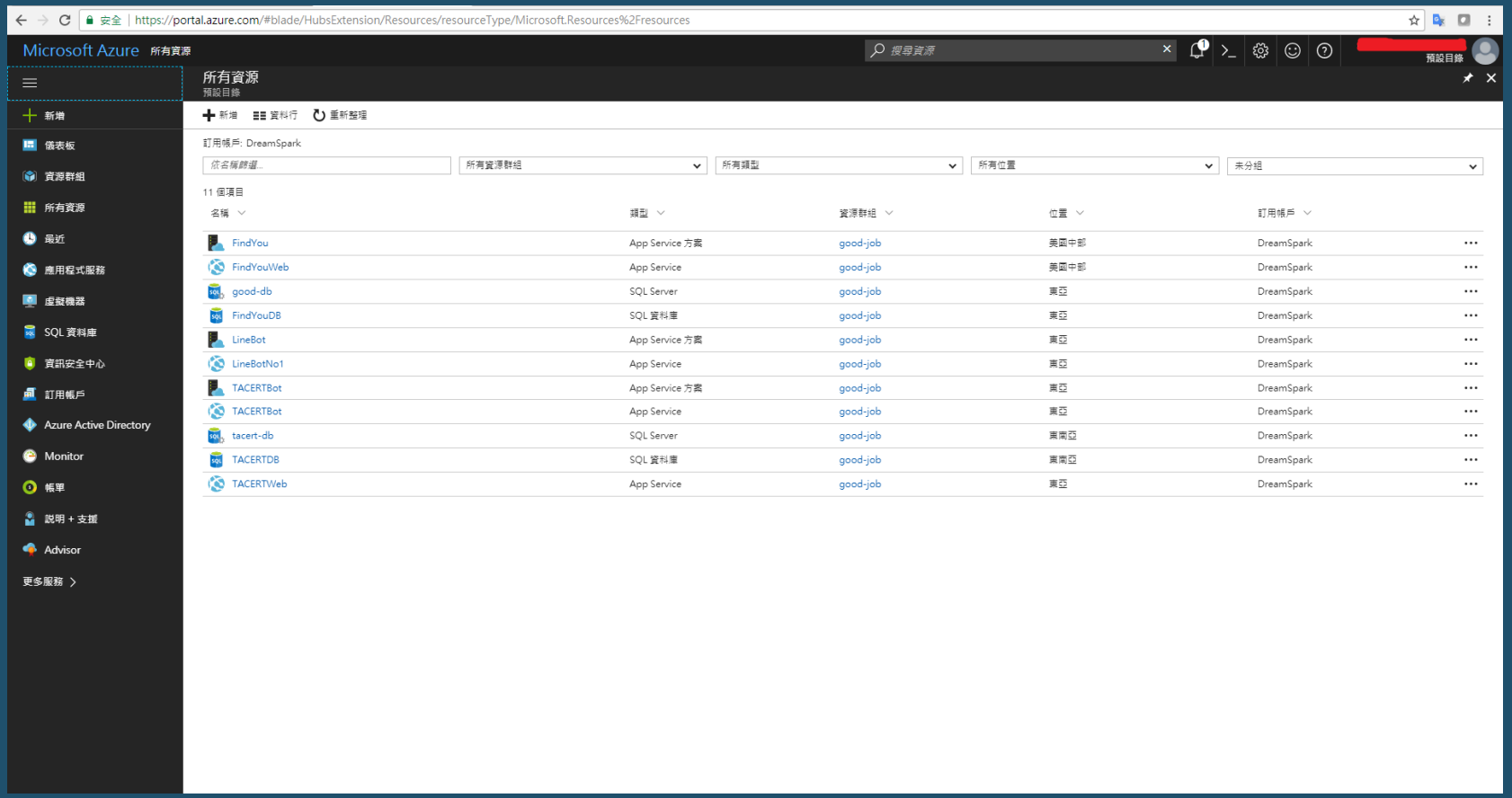
LINE BOT 開發

- LINE Messaging API Reference (https://developers.line.me/en/docs/messaging-api/overview/)



LINE BOT 開發

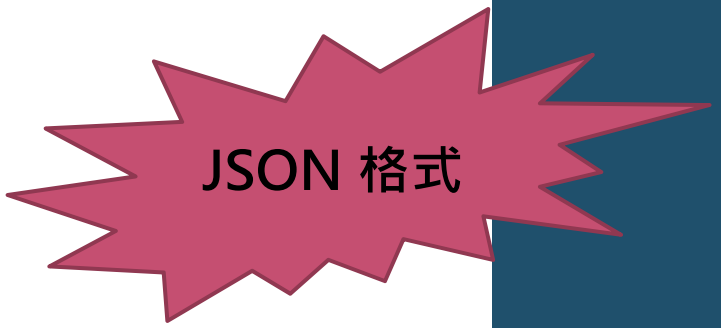
- 免費SSL網頁空間：Microsoft Azure (https://azure.microsoft.com/zh-tw/)



LINE BOT 開發

- LINE BOT 傳送與接收的訊息封包


```
{
 'events':
  [
   {
    'type':'message',
    'replyToken':'c049d461facc41268cf74dd0c79b5843',
    'source':
     {
      'groupId':'C8c86064b3991[redacted]',
      'userId':'Ufe21d353ced1b[redacted]',
      'type':'group'
     },
    'timestamp':1498809522091,
    'message':
     {
      'type':'text',
      'id':'631609524321 5',
      'text':'?你好'
     }
   }
  ]
}
```

LINE BOT 開發

- LINE BOT 可以發送的訊息種類
  (https://developers.line.me/en/docs/messaging-api/message-types/)

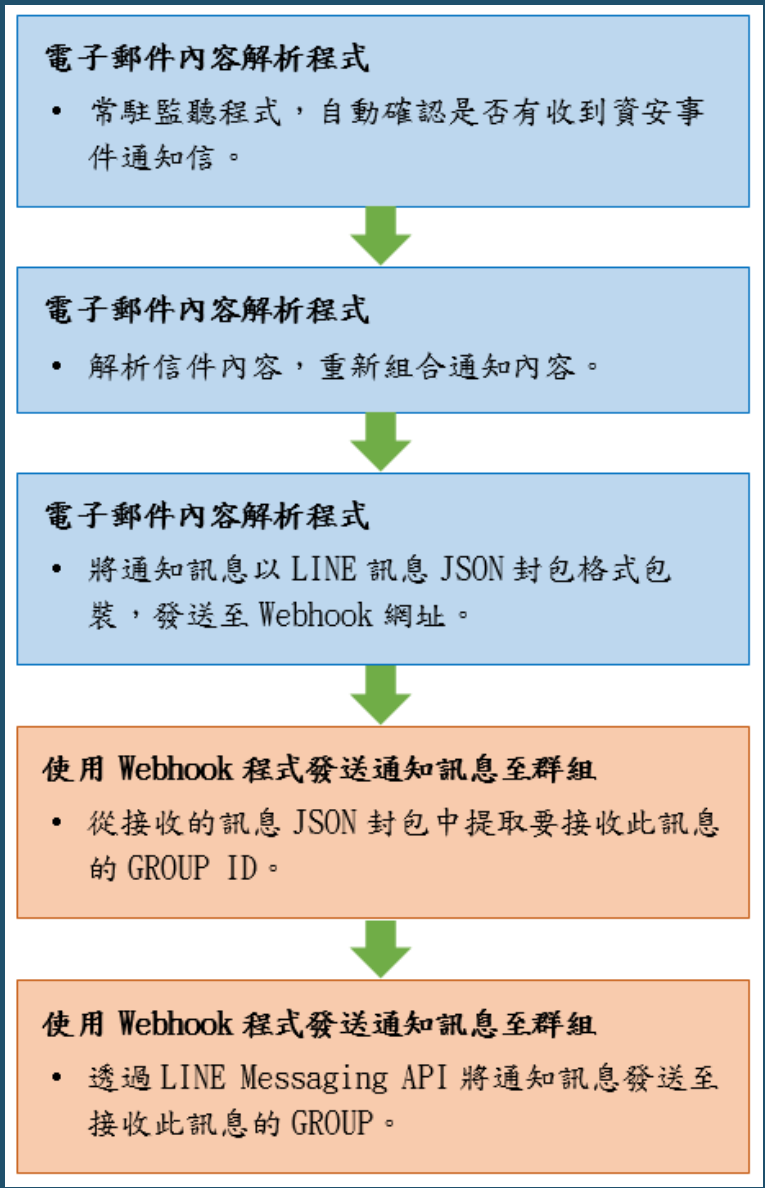
  「message」裡面的「type」屬性

```
    'message':
    {
      'type':'text',
      'id':'631609524321 5',
      'text':'?你好'
    }
  }
 ]
}
```

- *text*：純文字。
- *image*：圖片。
- *video*：影片。
- *audio*：聲音。
- *location*：地點。
- *sticker*：表情符號、貼圖。

使用 LINE BOT 於資安通報

- 運作流程：



使用 **LINE BOT** 於資安通報

- 運作流程 1：取得 **LINE** 群組 **GROUP ID**

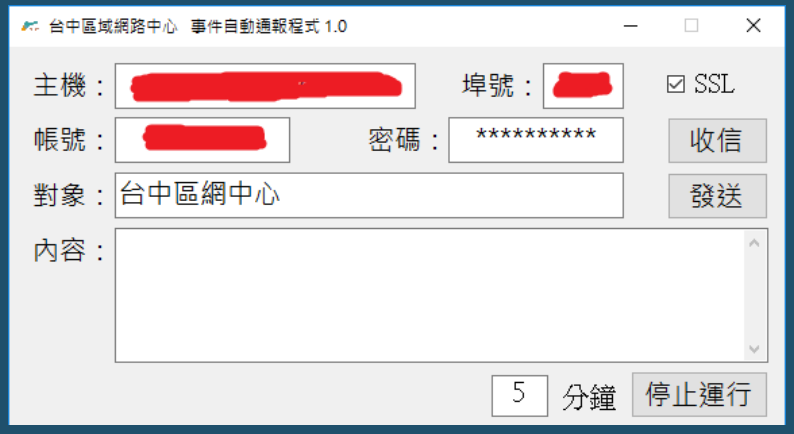
群組將 **LINE BOT** 加為好友 → 向 **LINE BOT** 發送對話訊息 → **Webhook** 程式 解析對話 **JSON** 封包 取得群組 **GROUP ID**

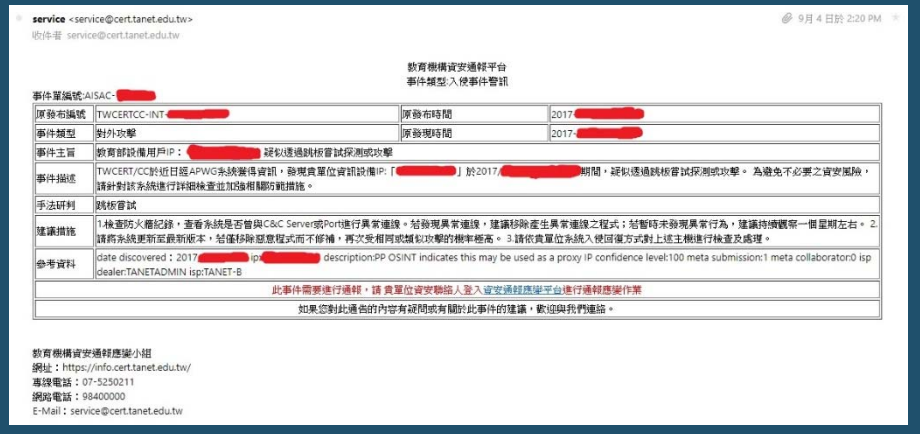

```
{
 'events':
  [
   {
    'type':'message',
    'replyToken':'c049d461facc41268cf74dd0c79b5843',
    'source':
     {
      'groupId':'C8c86064b3991[redacted]',
      'userId':'Ufe21d353ced1b[redacted]',
      'type':'group'
     },
    'timestamp':1498809522091,
    'message':
     {
      'type':'text',
      'id':'6316095243215',
      'text':'?你好'
     }
   }
  ]
}
```

使用 LINE BOT 於資安通報

- 運作流程 2：接收資安通報信件，解析信件資料
  - ➢找出通報單位，與通報事件

確認是否有資安通報郵件 → 接收資安通報郵件 → 解析郵件資料 找出通報單位 與事件類型



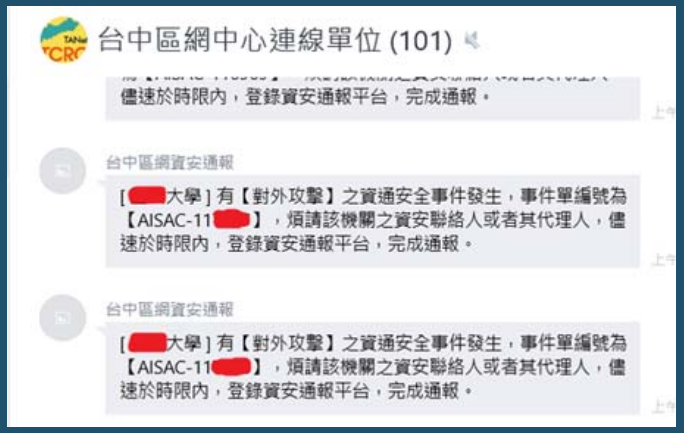


使用 LINE BOT 於資安通報

- 運作流程 3：將訊息內容與群組 GROUP ID 包裝成 JASON 訊息封包
  - ➢ 發送至 LINE 群組

完成通報內容重新組合 → 將訊息內容與群組 **GROUP ID** 包裝成 **JASON** 訊息封包 → 將包裝好的 **JASON** 封包透過 **API** 發送至 **LINE** 群組


```
{
 'events':
  [
   {
    'type':'message',
    'replyToken':'c049d461facc41268cf74dd0c79b5843',
    'source':
     {
      'groupId':'C8c86064b3991[redacted]',
      'userId':'Ufe21d353ced1b[redacted]',
      'type':'group'
     },
    'timestamp':1498809522091,
    'message':
     {
      'type':'text',
      'id':'6316095243215',
      'text':'?你好'
     }
   }
  ]
}
```

LINE BOT 開發參考

http://studyhost.blogspot.tw/2016/05/linebot-1-clinebot.html

使用 LINE BOT 於資安通報

- 先引用 / 設定 Channel access token

```
string ChannelAccessToken = "O+K0W2g1lzC7x41JBYa82xHAOu74yikA4K0TD1iNQh4DI6XHsbybl
```

- 設定解析訊息的變數

```
//取得 http Post RawData(should be JSON)
string postData = Request.Content.ReadAsStringAsync().Result;
//剖析JSON
var ReceivedMessage = isRock.LineBot.Utility.Parsing(postData);
```

使用 LINE BOT 於資安通報

- 取得使用者 / 群組 的 ID，並加入資料庫，之後發送訊息使用！

```
if (ReceivedMessage.events[0].message.text.Substring(0, 1) == "@")
{
    string[] UserData = ReceivedMessage.events[0].message.text.Split(',');
    if (UserData.Length == 2)
    {
        string user_id = new TcrcCertSqlService.TACERTWebService().QueryUserIDFromUserID(ReceivedMessage.events[0].source.groupId);
        if (user_id == "")
        {
            bool up = new TcrcCertSqlService.TACERTWebService().UploadUserData(ReceivedMessage.events[0].source.groupId, UserData[0].Substring(1, UserData[0].Length - 1), UserData[1]);
            if (up == true) isRock.LineBot.Utility.ReplyMessage(ReceivedMessage.events[0].replyToken, "會員加入成功！", ChannelAccessToken); //回覆用戶
            else isRock.LineBot.Utility.ReplyMessage(ReceivedMessage.events[0].replyToken, "會員加入失敗！", ChannelAccessToken);
        }
        else isRock.LineBot.Utility.ReplyMessage(ReceivedMessage.events[0].replyToken, "該組織已有會員資料！", ChannelAccessToken);
    }
}
```

使用 LINE BOT 於資安通報

- 將傳過來的資安通報內容，傳送給區網群組

```
else if (ReceivedMessage.events[0].message.text.Substring(0, 1) == "/")
{
    string[] UserData = ReceivedMessage.events[0].message.text.Split(',');
    if (UserData.Length == 2)
    {
        string user_id = new TcrcCertSqlService.TACERTWebService().QueryUserIDFromSchool(UserData[0].Substring(1, UserData[0].Length - 1));
        if (user_id != "")
        {
            string[] user_id_array = user_id.Split(',');
            for (int i = 0; i < user_id_array.Length; i++)
                isRock.LineBot.Utility.PushMessage(user_id_array[i], UserData[1], ChannelAccessToken);
        }
        else isRock.LineBot.Utility.PushMessage(user_id, "找不到會員資料！", ChannelAccessToken);
    }
}
```

使用 **LINE BOT** 於資安通報

- 簡報下載

http://www.tcrc.edu.tw/phocadownload/data/linebot.pdf

THANK YOU