

# 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

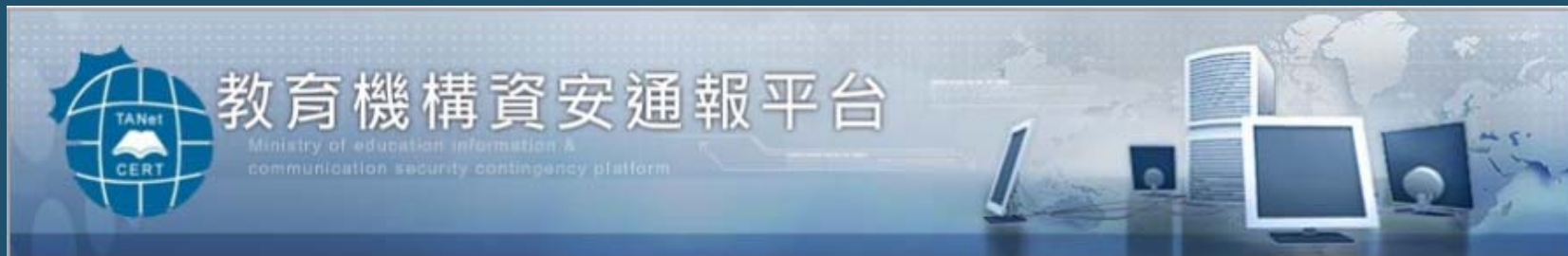
報告人：陳仕豪

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

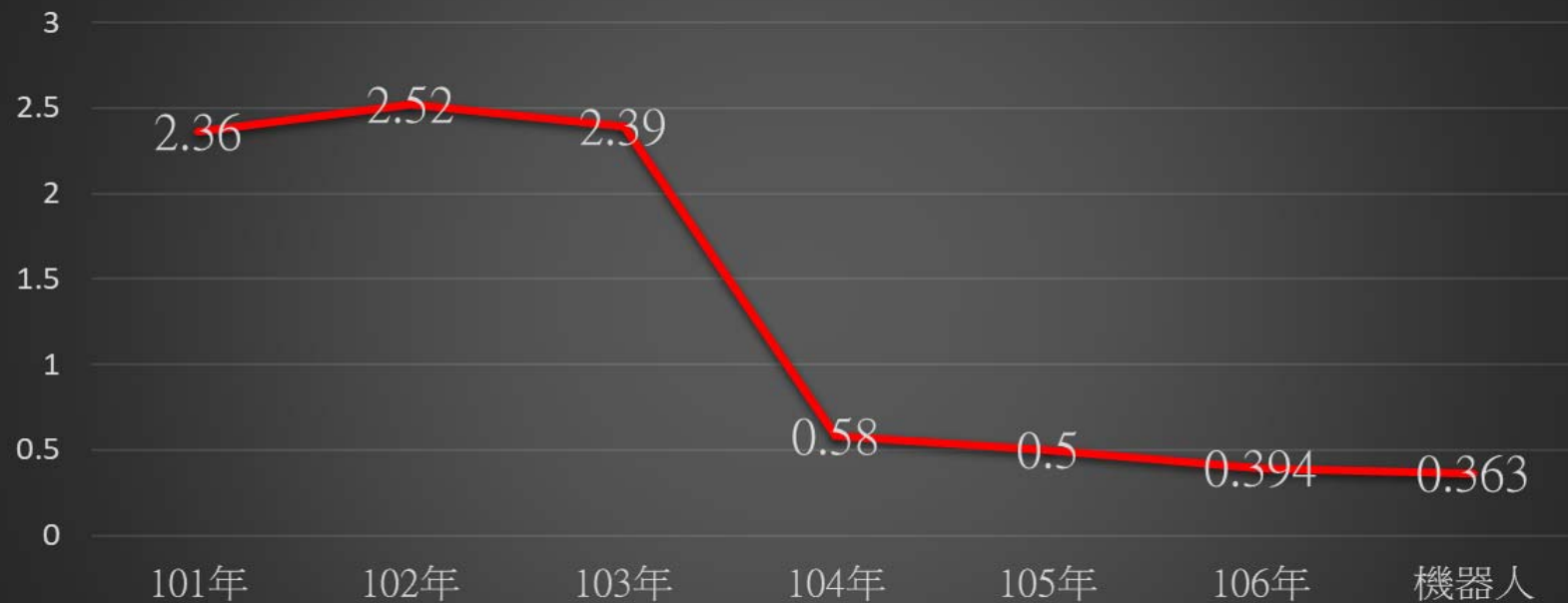
- 教育部資科司建立一**網路資安事件通報平台**，提供網路資安事件的應變處理通報。
- 事件通知係利用**電子郵件**以及**簡訊**，缺點是**不具即時性**，導致網路資安事件因為處理的延遲而造成損害擴大。



## 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。

臺中區網中心平均通報時數



# 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**。

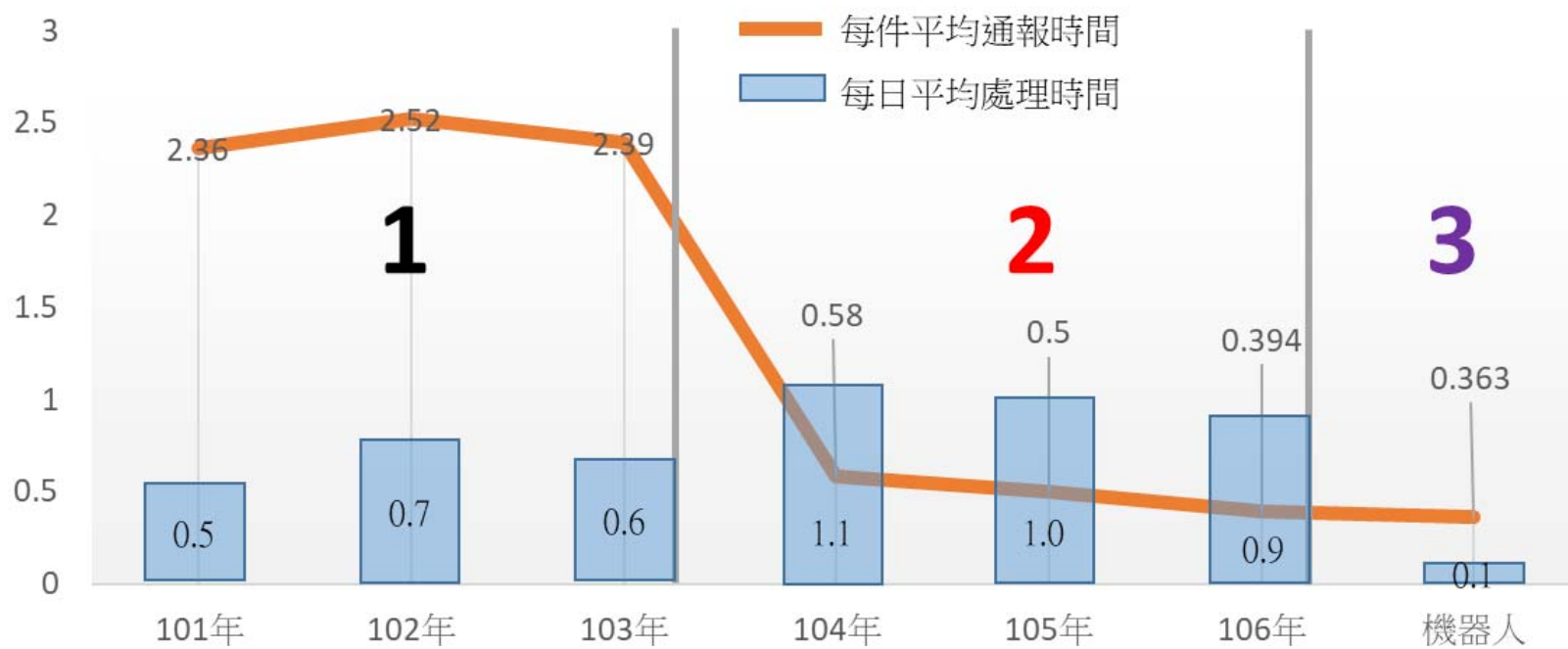


## Messaging API

將能夠讓LINE用戶針對您所提供的服務進行  
雙向溝通。

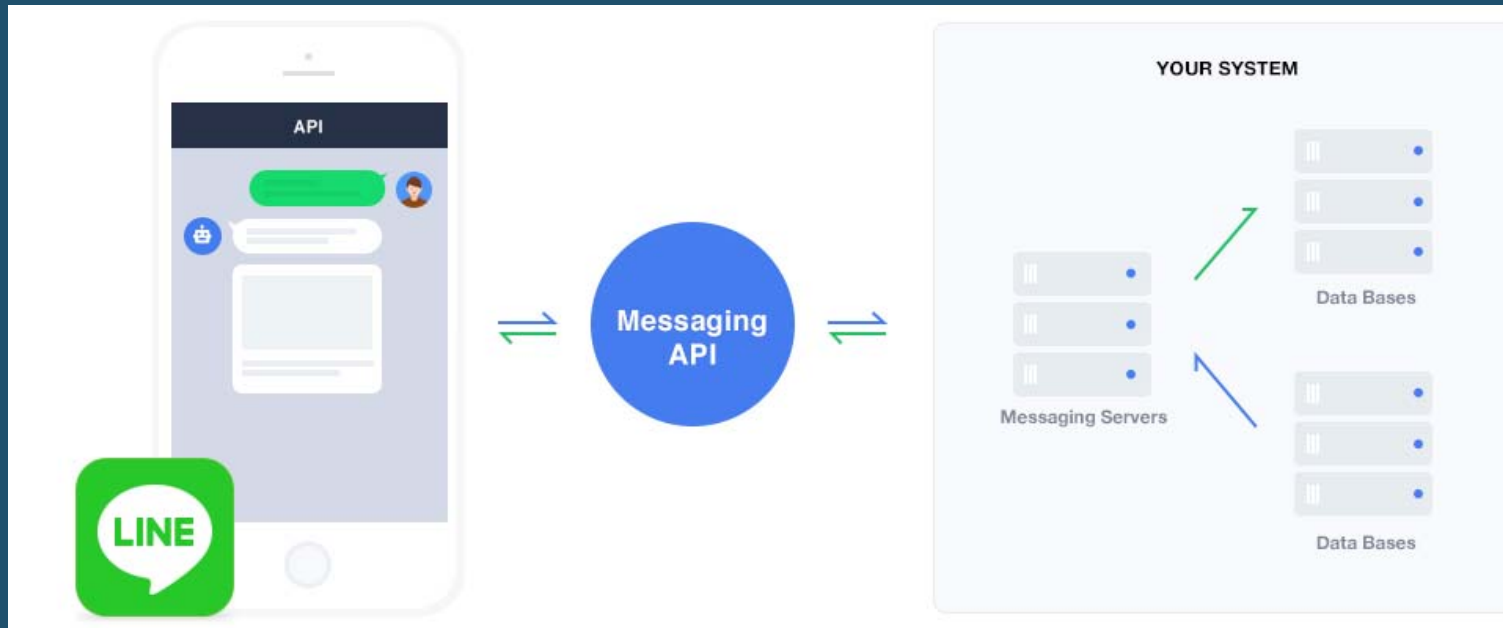
# 系統運行成效

臺中區網中心資安事件平均通報與處理時數



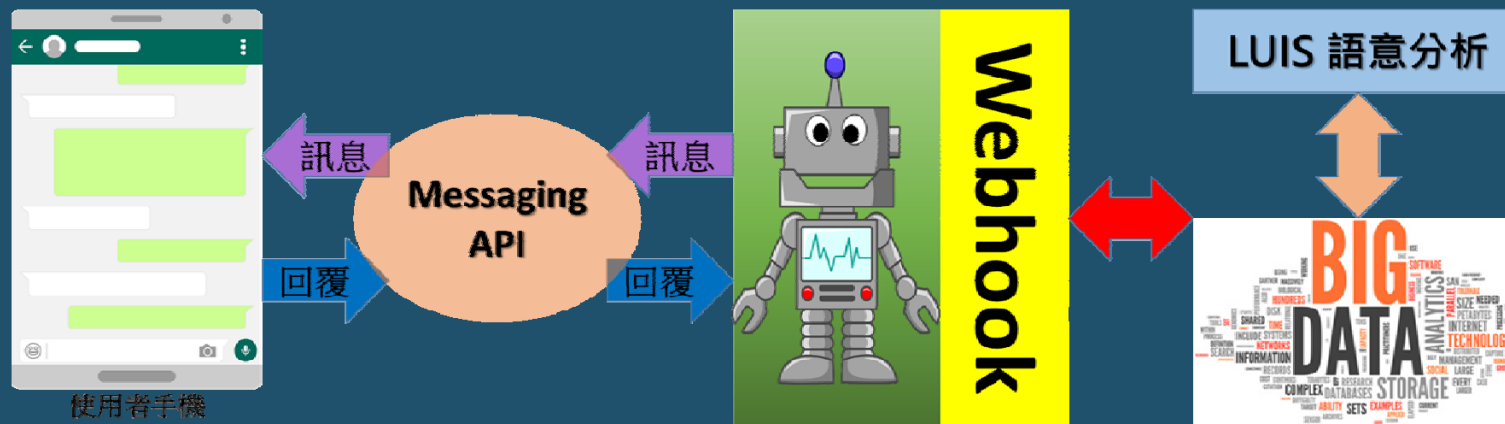
# 關於 LINE BOT

- 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：



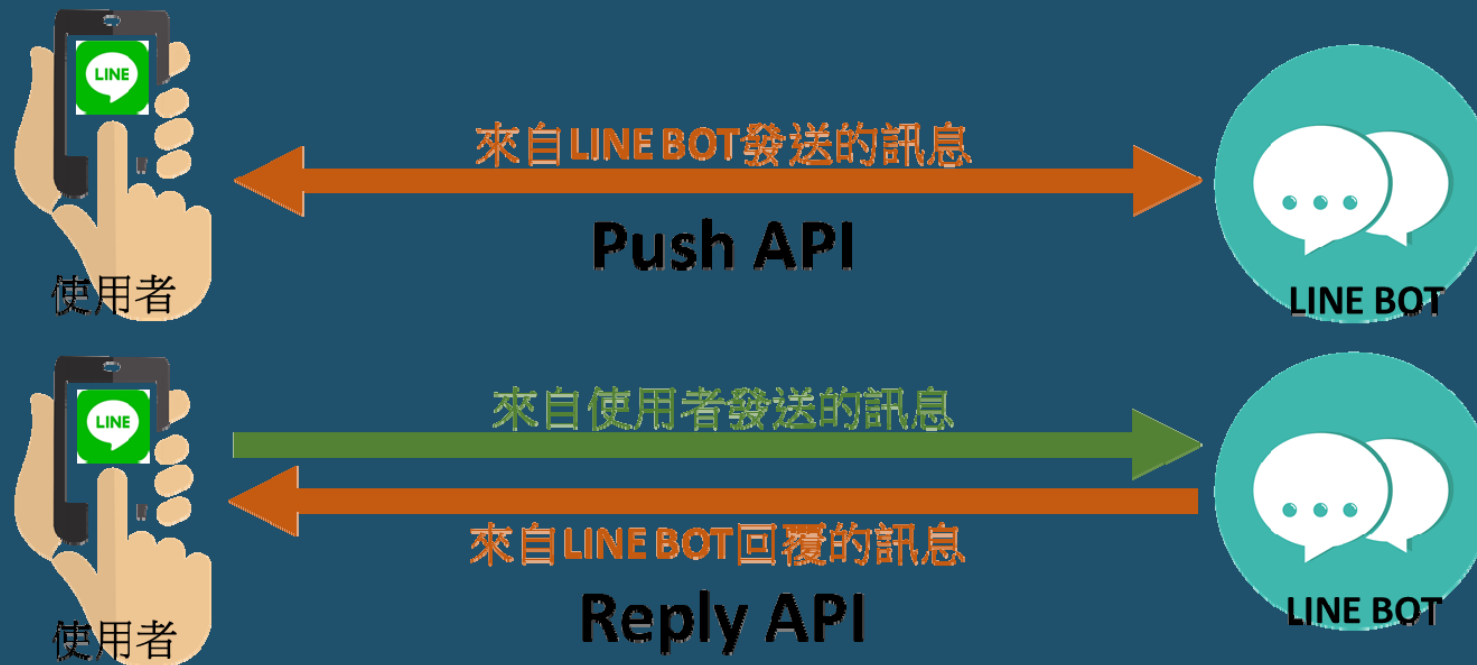
**Webhook** 是一種 **Http** 推送 **API**，為所建立的應用服務提供即時回覆訊息的一種方法。當 **Webhook** 被觸發後，會將回覆內容傳送到應用程式端去，這樣使用者可以立即獲得回覆訊息。



# 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！



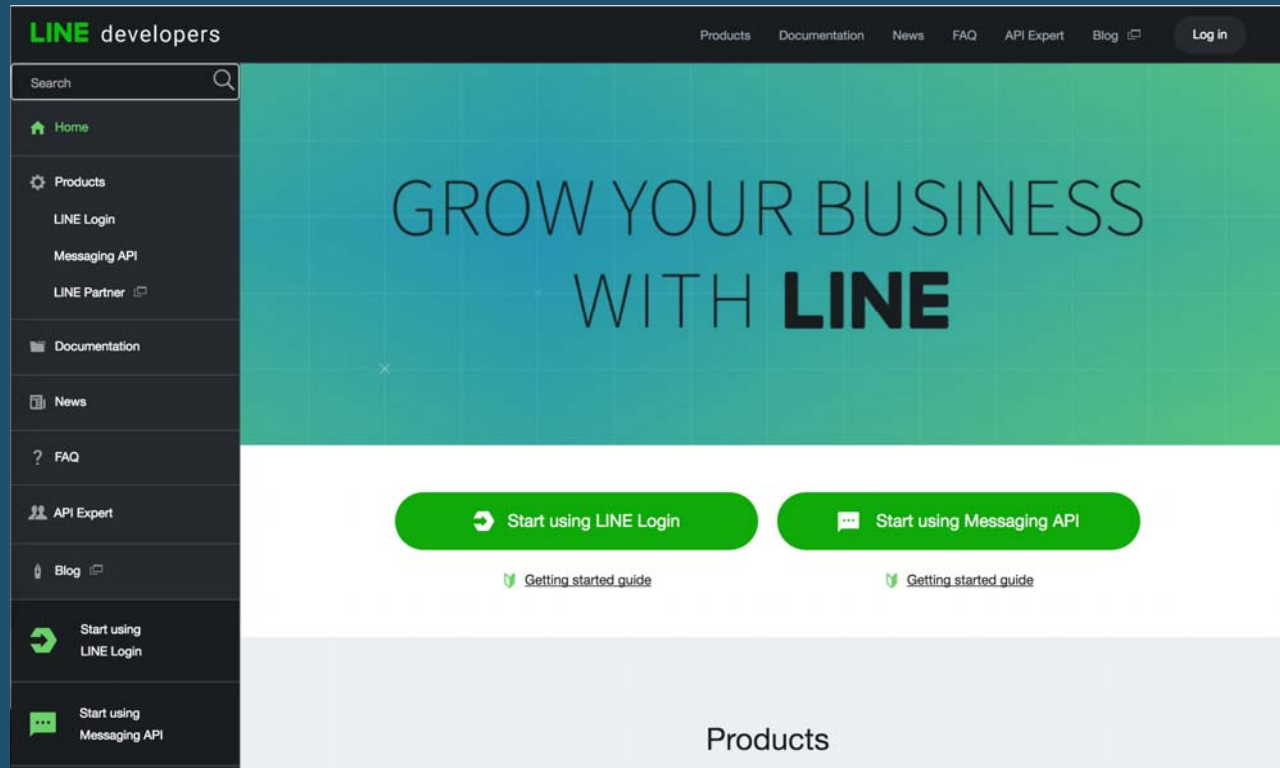
# 關於 LINE BOT

- 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

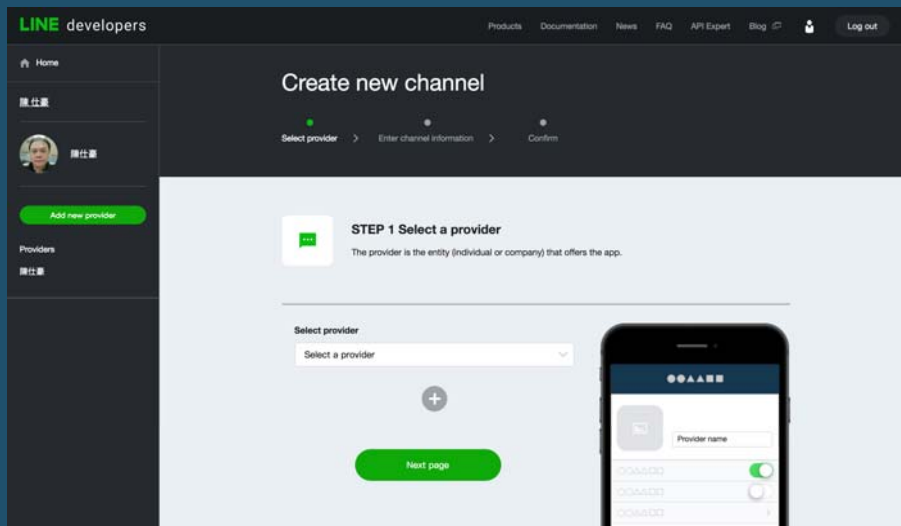
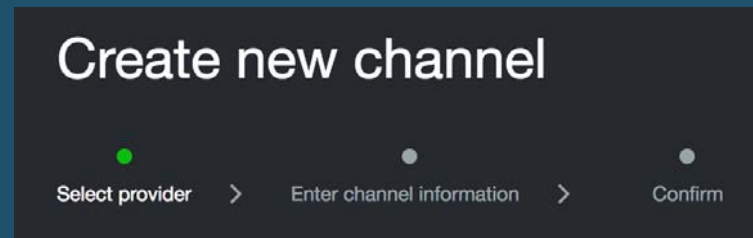
- 申請 LINE BOT (<https://developers.line.me/en/>)



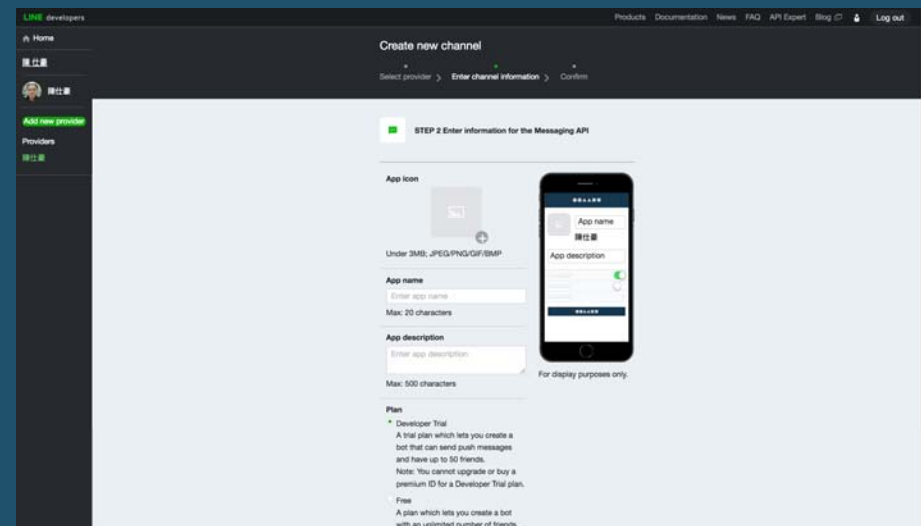
點選右邊的『Start using Message API』，接下來系統將會要求使用者完成三個設定步驟來完成新增一個通道 (LINE BOT)。

# 關於 LINE BOT

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



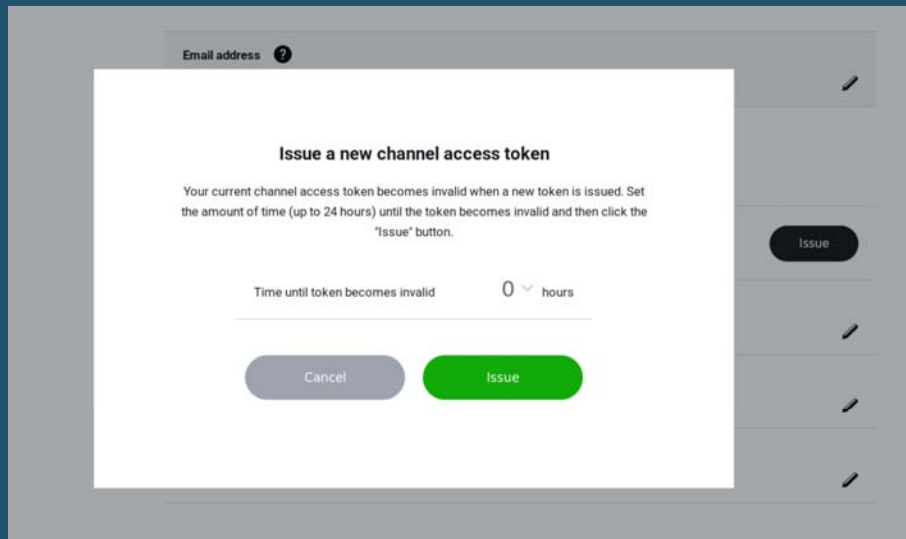
選取一個『提供者』，也就是告訴系統是誰提供這個 LINE BOT。



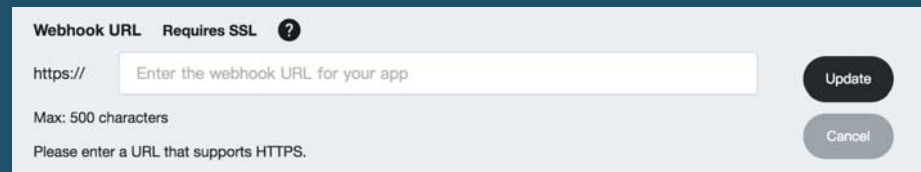
接下來，填上這個 Messaging API (LINE BOT) 的基本資料。

# 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！

Use webhooks ?

Enabled



Allow bot to join group chats ?

Enabled



打開才能讓這個 **LINE BOT**  
加入群組

# LINE BOT 開發

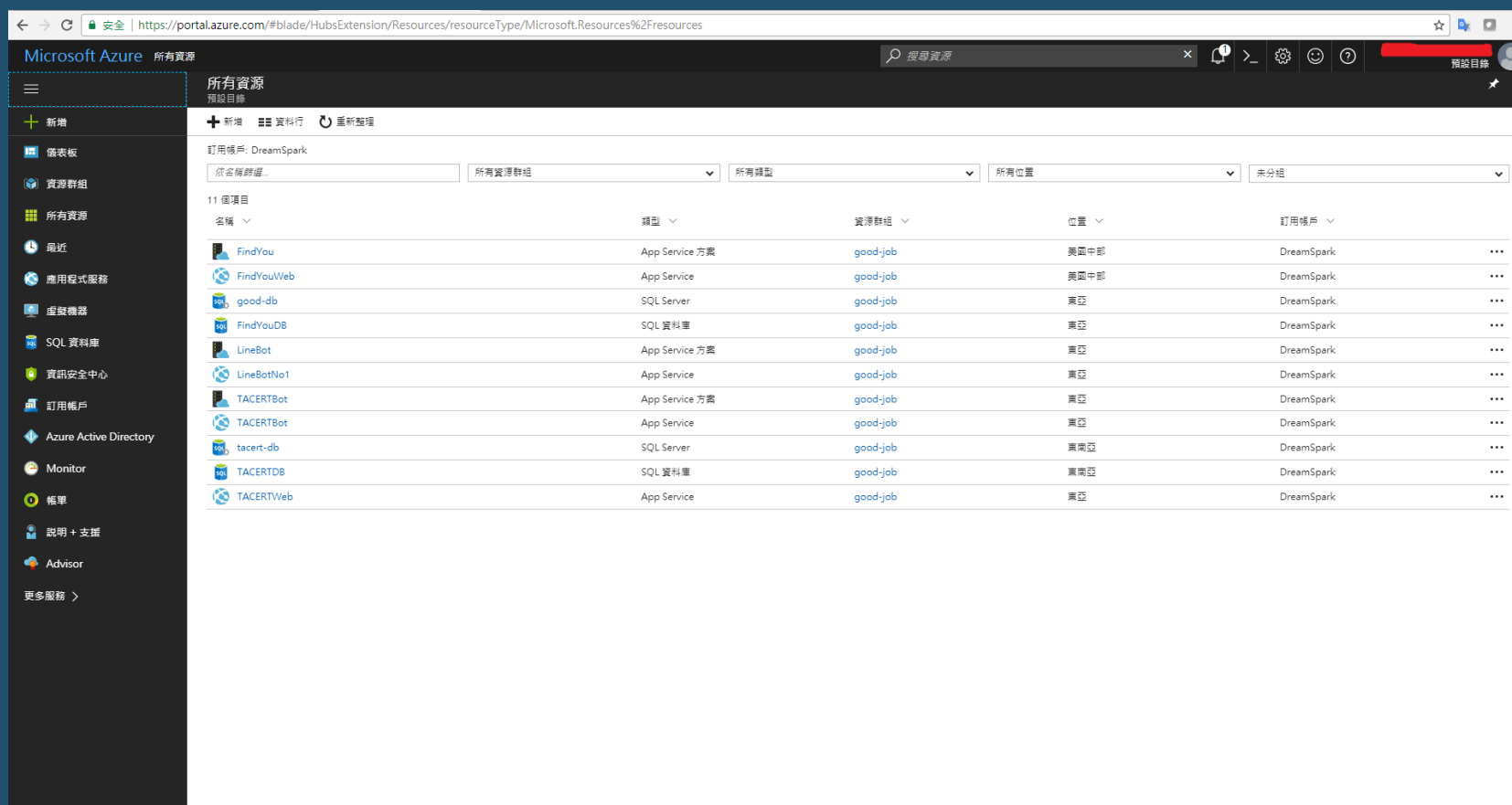
- LINE Messaging API Reference

(<https://developers.line.me/en/docs/messaging-api/overview/>)

The screenshot shows the LINE developers website's Messaging API documentation. The header includes the 'LINE developers' logo, navigation links for Products, Documentation, News, FAQ, API Expert, Blog, and a Log in button. A search bar is located on the left. The left sidebar contains a 'Documentation' section with a list of links: Messaging API, Quickstart (Overview, Getting started, Building a sample bot, Building a bot), Guides (Using rich menus, Using the LINE URL scheme, Using beacons, Sharing your bot with users), Concepts (Message types, Group chats, Managing roles, User consent), Reference (Messaging API reference), and Resources. The main content area is titled 'Messaging API' and includes an introduction: 'Use the Messaging API to build bots that provide personalized experiences for your users on LINE. To start building a bot, go to [Getting started](#).' Below this is a section titled 'How it works' with a paragraph explaining the data flow between the bot application, the LINE Platform, and the bot's server. A diagram illustrates this process: a smartphone labeled 'API' with a LINE logo icon is connected via a double-headed arrow to a central blue circle labeled 'MESSAGING API'. This circle is then connected via another double-headed arrow to a box labeled 'YOUR SYSTEM' which contains 'Messaging Servers' and 'Data Bases'.

# LINE BOT 開發

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



The screenshot displays the Microsoft Azure portal interface. The left sidebar contains navigation links for various services like Dashboard, Resource Groups, All Resources, Recent, App Service, Virtual Machines, SQL Databases, Security Center, Users, Azure Active Directory, Monitor, Billing, Help + Support, and Advisor. The main area shows a table of resources under the 'DreamSpark' subscription. The table columns are Name, Type, Resource Group, Location, and Subscription. The resources listed include FindYou, FindYouWeb, good-db, FindYouDB, LineBot, LineBotNo1, TACERTBot, TACERTBot, tacert-db, TACERTDB, and TACERTWeb.

名稱	類型	資源群組	位置	訂閱帳戶
FindYou	App Service 方案	good-job	美國中部	DreamSpark
FindYouWeb	App Service	good-job	美國中部	DreamSpark
good-db	SQL Server	good-job	東亞	DreamSpark
FindYouDB	SQL 資料庫	good-job	東亞	DreamSpark
LineBot	App Service 方案	good-job	東亞	DreamSpark
LineBotNo1	App Service	good-job	東亞	DreamSpark
TACERTBot	App Service 方案	good-job	東亞	DreamSpark
TACERTBot	App Service	good-job	東亞	DreamSpark
tacert-db	SQL Server	good-job	東南亞	DreamSpark
TACERTDB	SQL 資料庫	good-job	東南亞	DreamSpark
TACERTWeb	App Service	good-job	東亞	DreamSpark



# 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': '6316095243215',
        'text': '?你好'
      }
    }
  ]
}
```



JSON 格式

# 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': '6316095243215',  
      '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：

電子郵件內容解析程式

- 常駐監聽程式，自動確認是否有收到資安事件通知信。



電子郵件內容解析程式

- 解析信件內容，重新組合通知內容。



電子郵件內容解析程式

- 將通知訊息以 LINE 訊息 JSON 封包格式包裝，發送至 Webhook 網址。



使用 Webhook 程式發送通知訊息至群組

- 從接收的訊息 JSON 封包中提取要接收此訊息的 GROUP ID。



使用 Webhook 程式發送通知訊息至群組

- 透過 LINE Messaging API 將通知訊息發送至接收此訊息的 GROUP。

# 使用 **LINE BOT** 於資安通報

- 運作流程 1：取得 **LINE** 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

台中區網路中心 事件自動通報程式 1.0

主機： [REDACTED] 埠號： [REDACTED] ☒ SSL

帳號： [REDACTED] 密碼： [REDACTED]

對象： 台中區網中心

內容： [REDACTED]

5 分鐘

service <service@cert.tanet.edu.tw>  
收件匣 service@cert.tanet.edu.tw 9月4日於 2:20 PM

教育機構資安通報平台  
事件類型：入侵事件資訊

事件單編號: AISAC- [REDACTED]

原發布編號	TW/CERTCC-INT [REDACTED]	原發布時間	2017 [REDACTED]
事件類型	對外攻擊	原發現時間	2017 [REDACTED]
事件主旨	教育部設備用戶IP: [REDACTED] 疑似透過跳板嘗試探測或攻擊		
事件描述	TW/CERT/CC於近日經APWG系統獲得資訊，發現貴單位資訊設備IP: [REDACTED] 於2017 [REDACTED] 期間，疑似透過跳板嘗試探測或攻擊。為避免不必要之資安風險，請針對該系統進行詳細檢查並實施相關防護措施。		
手法研判	跳板嘗試		
建議措施	1.檢查防火牆記錄，查看系統是否曾與C&C Servers或Port進行異常連線，如發現異常連線，建議移除產生異常連線之程式；如暫時未發現異常行為，建議持續觀察一個星期左右。2.請將系統更新至最新版本，若僅移除惡意程式而不修補，再次受相同攻擊的機率極高。3.請依貴單位系統入侵回應方式對上述主機進行檢查及處理。		
參考資料	date discovered: 2017 [REDACTED] ip: [REDACTED] description: PP OSINT indicates this may be used as a proxy IP confidence level: 100 meta submission: 1 meta collaborator: 0 ip: dealer: TANETADMIN isp: TANET-8		

此事件需要進行通報，請貴單位資安聯絡人登入資安通報平台進行通報應變作業  
如果您對此通告的內容有疑問或有關於此事件的建議，歡迎與我們連絡。

教育機構資安通報應變小組  
網址: <https://info.cert.tanet.edu.tw/>  
專線電話: 07-5250211  
傳真電話: 98400000  
E-Mail: [service@cert.tanet.edu.tw](mailto:service@cert.tanet.edu.tw)

# 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

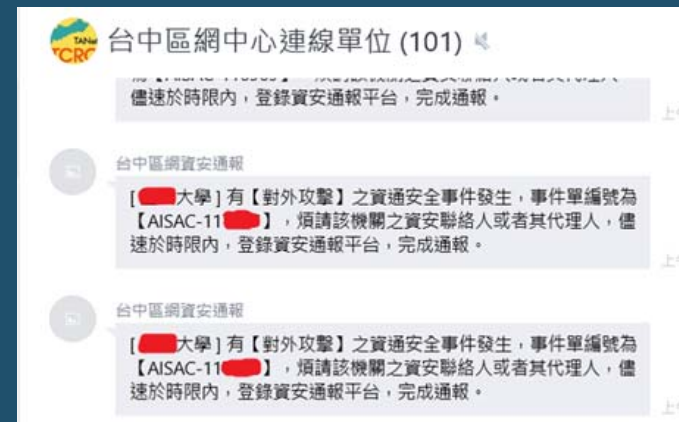


將訊息內容與群組  
**GROUP ID** 包裝成  
**JASON** 訊息封包



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+KOW2g11zC7x41JBYa82xHAOu74yikA4K0TD1iNQh4DI6XHsbyblv
```

- 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 TcrCertSqlService.TACERTWebService().QueryUserIDFromUserID(ReceivedMessage.events[0].source.groupId);
        if (user_id == "")
        {
            bool up = new TcrCertSqlService.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 TcrCertSqlService.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