**網路攻防技術與應用作業：自動登錄買票(訂票)–台灣虎航**

座號：I4B18 姓名：李信均 學號：410906218

**適用作業系統**：windows

**目標：**台灣虎航自動訂機票

**使用語言：**python

**準備：**需要有安裝python、pip，使用chrome註冊虎航帳號並選擇信任此裝置

**內容說明**：

主要功能為可以自動化模擬人訂票動作，目前期望的功能有：

1. 快速訂票
2. 指定日期/地點
3. 尋找最便宜機票
4. 設定預算/票數

**操作/使用方法**：

1. 將檔案解壓縮
2. 開啟終端進入檔案，輸入 ./build or build.bat or build (可能因每天電腦不同設定而須不同執行方式)
3. 前項指令將會自動下載chromedriver並下載程式所需的package，下載完成後可以看到一個可使用參數，目前有些可用有些無法，請先直接無視，這些參數功能未來版本將會慢慢補上
4. 輸入執行python3 main.py即開始自動化操作
5. 如果需要圖片驗證目前需要手動完成，其餘都不需要進行任何操作
6. 最後付款確認按鈕會由使用者自行決定是否按下去
7. 到付款介面後回到終端輸入quit即可結束程式

**基本功能**：

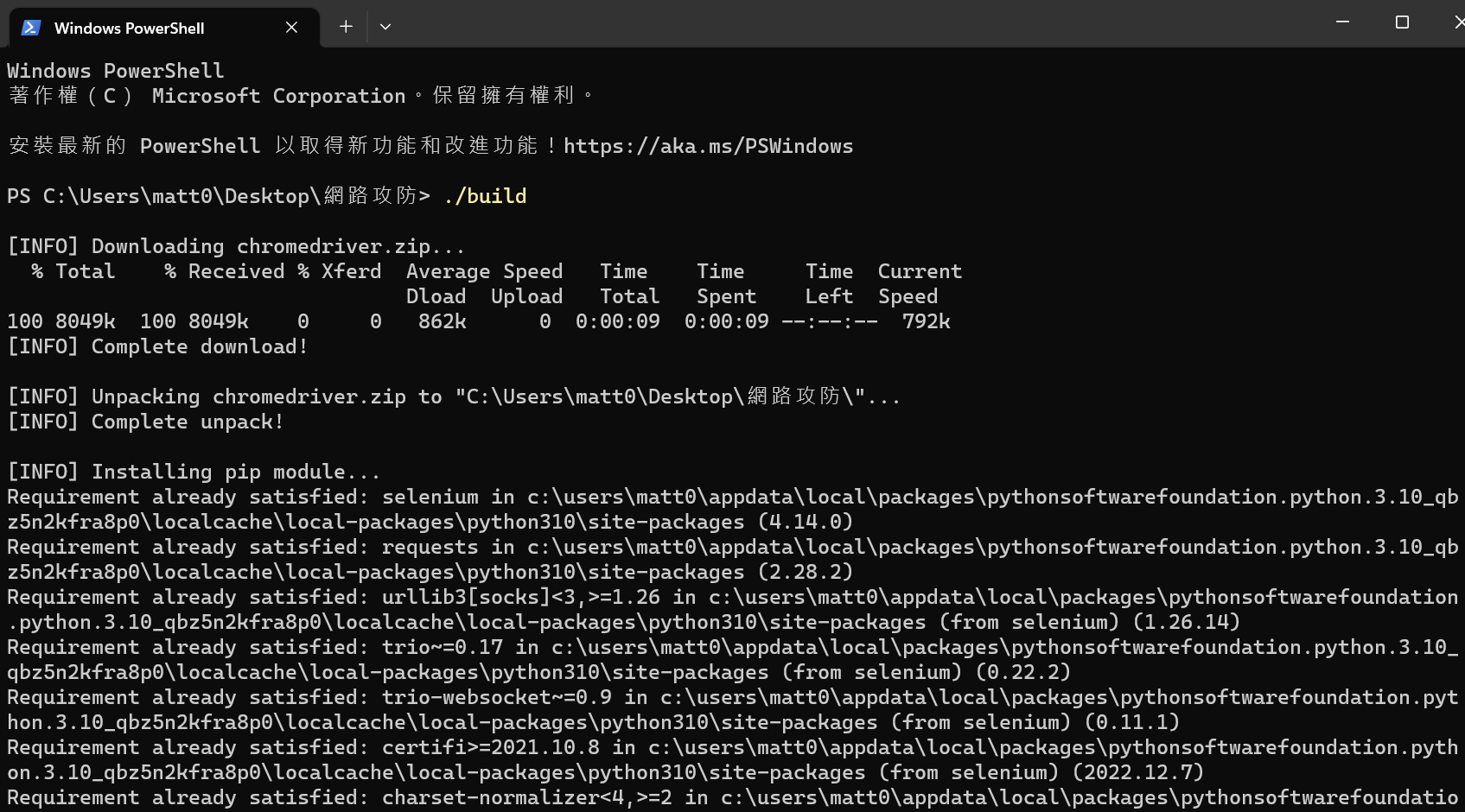
基本功能已實作完成，實作方式為使用python的selenium套件執行使用者的動作，出現錯誤的原因可能因為網頁還沒跑完而程式已進行下去，而造成不正常。

目前處理驗證碼的部分因為需要搭配機器人學習還沒時間完成，後續有時間會補上。

最後付款按鈕為避免系統選了使用者不想要的，因此確認付款動作將由使用者按下。

**執行結果：**

**輸入 ./build 下載所需套件等**

****

**一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述**

輸入python3 main.py開始程式

**一張含有 文字, 軟體, 字型, 多媒體軟體 的圖片

自動產生的描述**

進入虎航首頁

**一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述**

點擊幣別按鈕選TWD

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

選擇單程票、出發地跟目的地，目前預設為台北到大阪單程票，日期預設為當日，接下來進行搜尋

這裡遇到兩個困難點：

1. 等待系統取資料的時間
   1. 說明：

因為虎航的系統是選擇完出發地後，去後台request一個可選目的地的data，因此這裡需要wait系統request到的時間

* 1. 解決方法：

selenium有一個功能是當尋找到目標元素時才繼續執行動作，使用這個功能可以等待我的目標元素被系統get到並輸出於網站後點擊，即為偵測到KIX這段text時開始動作

1. 驗證碼
   1. 說明：關於是否需要驗證請見下方圖，系統會給此操作一個介於0~1的分數，機器人驗證需在框框內打勾，圖形驗證為選圖片，兩者不相同
   2. 推測：目前實測下來只要每個action間隔2秒以上跟短時間內不超過三次的搜尋分數皆會在0.5以上，問號部分沒有找到答案，但推測為短時間內進行多次的搜尋(5次以上)或action間幾乎沒有wait，即會觸發圖形驗證

一張含有 文字, 圖表, 螢幕擷取畫面, 行 的圖片

自動產生的描述

* 1. 解決方法：

目前機器人驗證可以自動完成、圖形驗證需要手動驗證，但在每個action間增加random time的wait發現較不易觸發驗證，即使觸發也只是機器人驗證，但多次搜尋還是會觸發圖形驗證

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

搜尋當天航班完畢後，會發現系統發送很多request，從許多封包中發現其中一個封包為

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 軟體 的圖片

自動產生的描述

這個封包向後台post一個要求，發現post包括20240209到20240211台北出發抵達大阪等關鍵參數，而後台也回傳一個data內容包括幾個吸引人的參數，date & amount，經過驗證後發現這些amount為對應日期的最低票價，我想我可以從這裡著手找出某個時間區段的最低票價

一張含有 文字, 螢幕擷取畫面, 功能表, 字型 的圖片

自動產生的描述

我把時間區段設為20231021到20240215，果真對這個網址post我製作的請求後會收到一個回傳的data，我把他們分解後得到這些數據，驗證後得知票價0的日期為無座位了，再從這些數據裡找出最便宜的機票

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

最便宜為20231212這天有2799的價錢，接著再回browser，自動開始訂這個最便宜的機票

這裡遇到一個困難點：

1. sessionId：
   1. 說明：這個ID為進入搜尋頁面後系統產生的，驗證後約為15分鐘會被刷新一次，類似的機制為買賣股票系統，因為股票是每分每秒都在浮動而機票也是，你進入這個購買畫面後可享有一定時間的這個價錢，但不可能你永遠停在這裡就一直是這個價錢，因此一段時間後將會要求你重新搜尋
   2. 解決方法：只好先進行搜尋一次才能取得這個ID，後續一定時間內都可以使用這個ID搜尋

系統會自動輸入這個日期並開始搜尋

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述

搜尋到後，系統將會選擇最便宜機票，不過虎航提供三種票價組合，因為本系統目標為最便宜機票，因此選擇最便宜的組合(行李、餐點需要皆要加錢)，因為票價為浮動，後來去截圖發現票價已經不同，因此組合的圖為範例，再下方訂票圖為原本的

這裡遇到兩個困難點：

1. 同天不同票價
   1. 說明：有時會有同天但兩班不同的票價，下圖這個就是一個範例，因此當需要找較便宜票價時會比較麻煩，一開始我選的點擊按鈕為箭頭，但出現兩班時系統會點擊第一個箭頭而不是便宜的，後來改為點擊價錢時又發現因為上方也出現過這個價錢，因此系統優先點擊上方的
   2. 解決方法：後來研究過後還是以價錢點擊，因為發現selenium找元素時如果有多個元素，會形成類似陣列的東西供選擇，因此選擇陣列值為-1也就是最後一個就可以解決了
2. 下一步按鈕被擋住
   1. 說明；因為虎航系統會要求使用cookie，如果不點擊接受本系統無法成功按下下一步
   2. 解決方法：其實有種方法，第一種為調整click的位置、第二種為接受cookie，後來選擇後者是因為怕調整click位置後造成其他錯誤可能要修更久

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

自動產生的描述

訂票需要會員或訪客購買，因為有會員之後新增功能應該會比較方便，因此我使用一個fake email來登入，因為我事先註冊過並勾選信任此裝置因此不需要驗證。

老師可能也需要用chrome先用一個email註冊後登入並勾選信任此裝置後到程式檔案的data裡面改email跟password才能繼續使用，這裡提供fake email網址：https://temp-mail.org/en/

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

這步驟為護照、聯絡人資料，系統都會自動輸入

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

以下都為系統自動按下一步直到最後付款頁面

這裡遇到的困難點就是系統會找不到該按鈕在哪

1. 說明：

因為下一步按鈕不是在固定最下方，而是隨著網頁往下拉一點才會看到，前面的滾動頁面皆是只要拉到最下方就好，這裡較特別

1. 解決方法：

研究過後，發現selenium有提供尋找這個元素位置的function，因此滾動到該元素的位置即可繼續下一步

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

進入付款頁面，信用卡資料都是系統自動匯入網路上找到的假資料，右方價錢是加上稅金服務費等等的價錢，最後的付款按鈕要有使用者自己按，我怕用假資料付款太多次會被警察抓走。

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

終端機都會輸出現在在執行甚麼動作，方便使用者閱覽。

一張含有 文字, 螢幕擷取畫面, 功能表, 字型 的圖片

自動產生的描述

心得：

1. 特色：
   1. Chromedriver、套件下載自動化
   2. 找到自訂時間區段內的最低票價
   3. 比人類更快速的操作
   4. 登入、信用卡、護照資料只需在程式中輸入一次，之後每次使用都會自動填入
2. 困難處：
   1. 等待系統取資料時間
   2. 驗證碼
   3. sessionId
   4. 同天不同票價
   5. 系統會找不到該按鈕在哪
   6. 首次使用前需要先註冊過並勾選信賴該裝置
   7. 要花時間找每個標籤特別的地方
   8. 有時一出錯直接關掉browser完全沒法除錯只好再跑一次
   9. 一直進入別人網站觸發機器人驗證
   10. 一開始其實非常沒想法
3. 突破點：
   1. 找每個標籤特別地方的部分沒什麼突破點，就只能下功夫慢慢找
   2. 後來索性在程式有可能結束的地方先wait一段時間看哪裡有錯在慢慢改
   3. 一開始其實是想去找國外的廉價航空，比較不需煩惱法律問題
4. 應加分點
   1. 我認為找到可以取得最低票價這個功能可以加20分，這個功能或許能在航空公司誤植價錢時使用，假如2799的票價key成279，也剛好被我讀取到，則馬上下定，航空公司也無法反悔
5. 寫作過程心得：

從一開始甚麼都沒有到現在有一點樣子其實還滿開心，像是看著一個小孩長大的感覺。

坦白說，從大學開始學習程式，真的實作的次數屈指可數，大部分都是考邏輯題目而非這樣的實作，對我來說很是新鮮，而且雖然這份實作雖然算是在攻擊其他網站，但也讓我學到，原來web的devTools都隱藏了這麼多的資訊，想必這次經驗會讓我未來有更多想法。像是前幾周有教授要求要看完線上影片才算有出席該堂課，我就靠著這個經驗去抓可以用的request封包，結果證明是可行的，2.5小時的影片只花了半小時寫code的時間

1. 抱怨點：

常常修bug修到有點懷疑人生，而且有時動作太快，我甚至不知道發生了甚麼網頁就被關掉，關於selenium我也是第一次接觸，雖然是個很方便的套件，但網路上資料沒那麼豐富，大部分都是java實作的，但我java實在寫得非常差，一開始原本要用c去實作，但發現c的套件不知為何怎麼裝都裝不好，最後果斷放棄，改用python。

--------------------------v1 end here----------------------------

-----------------------------v2 start here--------------------------

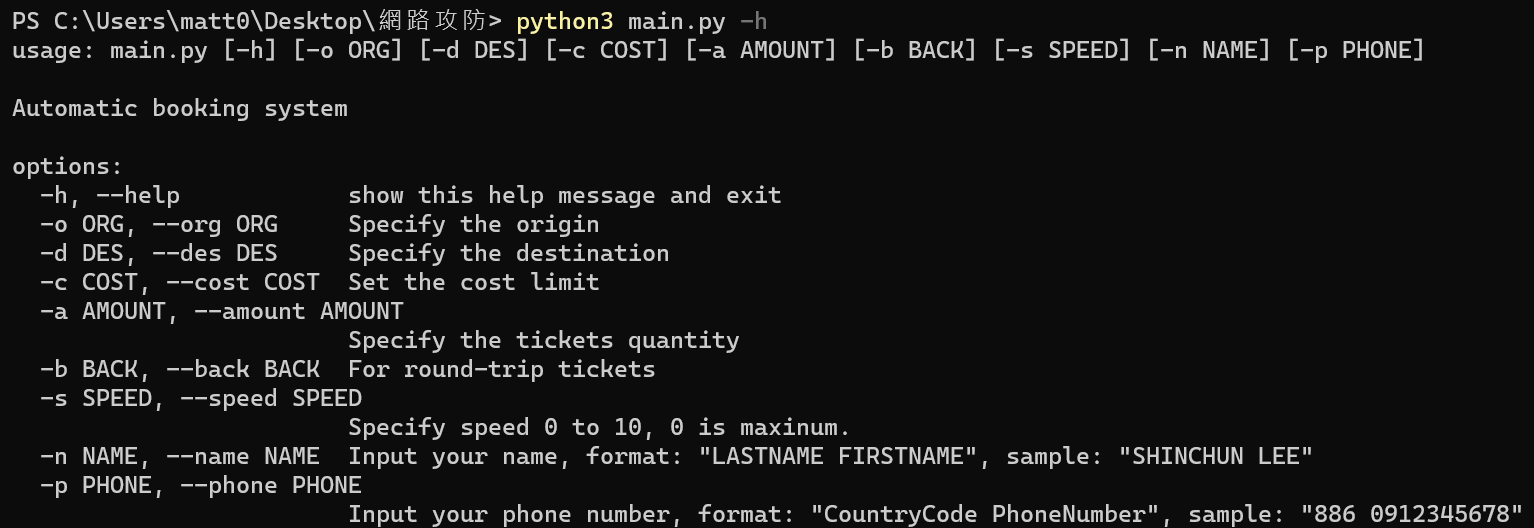
目前可用新功能：

-h 可以查看有什麼argument可用

-o 可以指定出發地，但僅能輸入機場英文縮寫，ex. KIX / TPE

-d 可以指定目的地，但僅能輸入機場英文縮寫，ex. KIX / TPE

-s 可以指定每個action中間暫停多久，base是1sec，如果要追求最快速則設定為0



困難點：

* 這兩周報告考試等等的事情很多，只能百忙中擠出一點時間來做這份作業
* -s 非常容易因為速度過快導致系統錯誤，目前還在尋找解決方法

--------------------------v2 end here----------------------------

--------------------------v3 start here--------------------------

**內容說明：**

* 如果./build無法執行，請手動到<https://googlechromelabs.github.io/chrome-for-testing/> or <https://chromedriver.chromium.org/downloads> 下載chromedriver，需要先手動查詢chrome版本，更新到最新版再進行下載
* 如果出現下圖的提示就需要先到註冊信箱驗證，這步驟需要老師自行驗證

一張含有 文字, 電子產品, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述

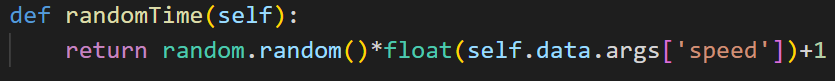
**新增功能：**

可以選擇人數。指令為：-a後面增加整數代表幾張，預設為1張

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

修改了-s的程式，將其改為一秒加上使用者輸入的值乘上一個隨機值，發現穩定性增加了，不過同時會比本來要求的極致快速差一點，但還是比人為操作更快速



**困難點：**

* 本周花比較多時間處理信箱驗證的問題，原先想法是可以透過網頁封包去下手，繞過信箱驗證，有找到一段token疑似是經過hash的token，我推測這和驗證碼會有關，不過目前還沒破解出來，現在想法有三種，使用者手動處理、用程式進入email讀取郵件、改用訪客訂票，三者都有不同的優缺點
* 選擇票數部分，因為涉及後續訂票時需要輸入所有乘客的資料(護照號碼、電話等等)，目前想法是直接使用預設的資料訂，後續在加錢更改資料，畢竟更改資料只需1000元，搶到便宜的票價，或許多這1000還是比其他票價便宜許多

**心得：**

這次的更動多了許多嘗試，因為這些功能比較偏需要實際嘗試才能找到最好的方法，因此當初把這些功能放到較後面才實作，email驗證這點我相信如果找出繞過驗證的方法一定可以加分，可能會使用hashid或john the ripper這兩個試試看能不能解決。

--------------------------v3 end here----------------------------

-------------------------v4 start here---------------------------

新增功能：

可以指定電話。指令為：-p後面為電話號碼，需要包含dial code，中間以連接符間格，例如：886-0912345678

可以指定姓名。指令為：-n後面為姓名，先輸入姓再輸入名，中間以連接符間格，例如：SHINCHUN-LEE

自動填寫乘客資料：原先情況只設定為只有一位乘客，後來新增選擇票數功能，第二位乘客後的資料就需要手動輸入，現在可以直接由系統帶入

**結果：**

指定電話



一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述

指定姓名&自動填入所有乘客資料

一張含有 文字, 字型, 螢幕擷取畫面, 圖形 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

**困難點：**

* 自動填入乘客資料這點，我想之後會用其他方式讓使用者決定是否要自己填入，因為目前的方法為每個名字後面多加一個單字，這樣可以避免系統檢查名字重複的情況發生，但這樣就必定得花錢改資料，我想還會有更好的方法

**心得：**

這周很尷尬，因為卡了一個專題的驗收必須忙，所以能實作的時間其實很少，但目前功能一切正常，希望下周可以完成度更好。

--------------------------v4 end here----------------------------

-------------------------v5 start here---------------------------

**新增功能：**

從原本使用selenium操作網頁的方式改成使用偽造封包的方式輸入，優點為不須等待每個操作所浪費的時間且程式行數小非常多，意味著效能肯定更好了。

**說明：**

我使用的偵查工具為burpsuite，一個可以對封包進行各種操作的工具，我側錄了所有操作tigerair時的封包，解析過後我再加以修改並重新送出，會發現基本上資料也都跟著改變了，由此得知所有資料變更都藏在後台且不去檢查前端資料，以下解釋所有封包。

先在首頁填完所有資料並送出

一張含有 文字, 軟體, 電腦圖示, 網頁 的圖片

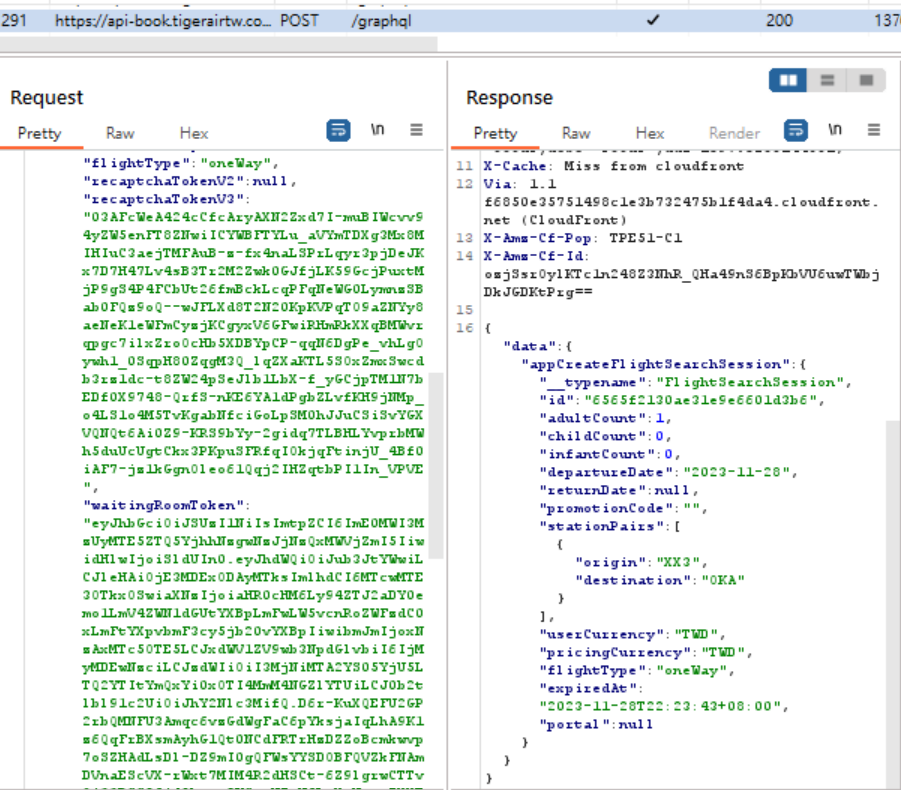
自動產生的描述

可以注意到這裏有個封包，post data名為create session，應該是由這裏開始作一連串的動作，

一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

自動產生的描述

接下來response回傳了許多資料，研究一下後得知幾乎都是前面首頁完成的資料，並返回一個id，這個id後續每個req都要用到，也是識別這個訂單的識別號碼，我一開始想直接generate一個session，不過左方req需要recaptchaTokenV2這個資料，由於要撈這個出來有點麻煩所以就先略過了，後續應該會搭配selenium直接撈這個封包的id，而不是用create一個，這樣的好處是可以使用重新整理頁面來確認是否有成功

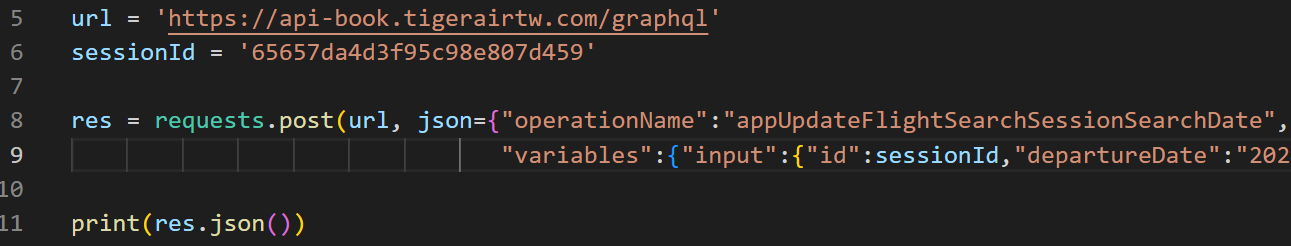


接下來的封包是search session，這個和之前所提到的查詢某個起始日到截止日的查詢不同，這個主要是查詢一個航班的詳細資料，主要需要的是兩個特別的key，分別是legSellKey、fareSellKey，前者是該航班的基本價，後者是方案，皆可透過這個封包查到，不過有時他還來不及查完，可以用另外一個封包查詢結果

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述

這裡sessionId暫時用手動輸入，且另開專案測試



這個封包名為search result，推測為檢測結果用的，總之的確legSellKey、fareSellKey(未截圖到)都有出現，不過有個問題是當座位沒了就會為null

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述

先用第一個search session找找看res有沒有回傳key，沒有的話再找一個search result，除非沒位置不然都不會有錯誤，當然後續會增加判斷是否有座位才繼續下一步，這也解決操作網站所產生的問題，就是當有意外情況出現時，比較難以去判斷，而這裡從分析封包就很清楚是否還有空座位

一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述

從網頁按下選擇方案、日期、航班後，看到這個封包，推測這是更新後台資料用的，內容有票價key、方案key、出發地、目的地等資料，且除了sessionId外沒有任何驗證，也方便我去偽造，可疑到我都覺得是網站故意讓駭客使用的，畢竟曾經聽說旅行社的便宜票價都是駭客去搶的

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述

我也嘗試去偽造這個封包，也成功獲得回應

一張含有 螢幕擷取畫面, 文字 的圖片

自動產生的描述

接下來我填寫了乘客資料和聯絡人資料等等並且在送出後，獲得了這樣一個封包，名稱為update passengers，顧名思義就是乘客資料嘛，內容也和我填的一模一樣，那本來使用網頁操作輸入的時間總共可能10秒，但封包一送出去就完成所有資料，可能一秒都不到呢！

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述

這裡我也一樣重新偽造一個封包出去，也是成功獲得回應

一張含有 螢幕擷取畫面 的圖片

描述是以中可信度自動產生

中間會有三個加購頁面，分析所有封包後發現如果無須加購則不會有任何封包需要傳送，我這裡是默認全都無須加購，因此我這裡直接跳到結帳，

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

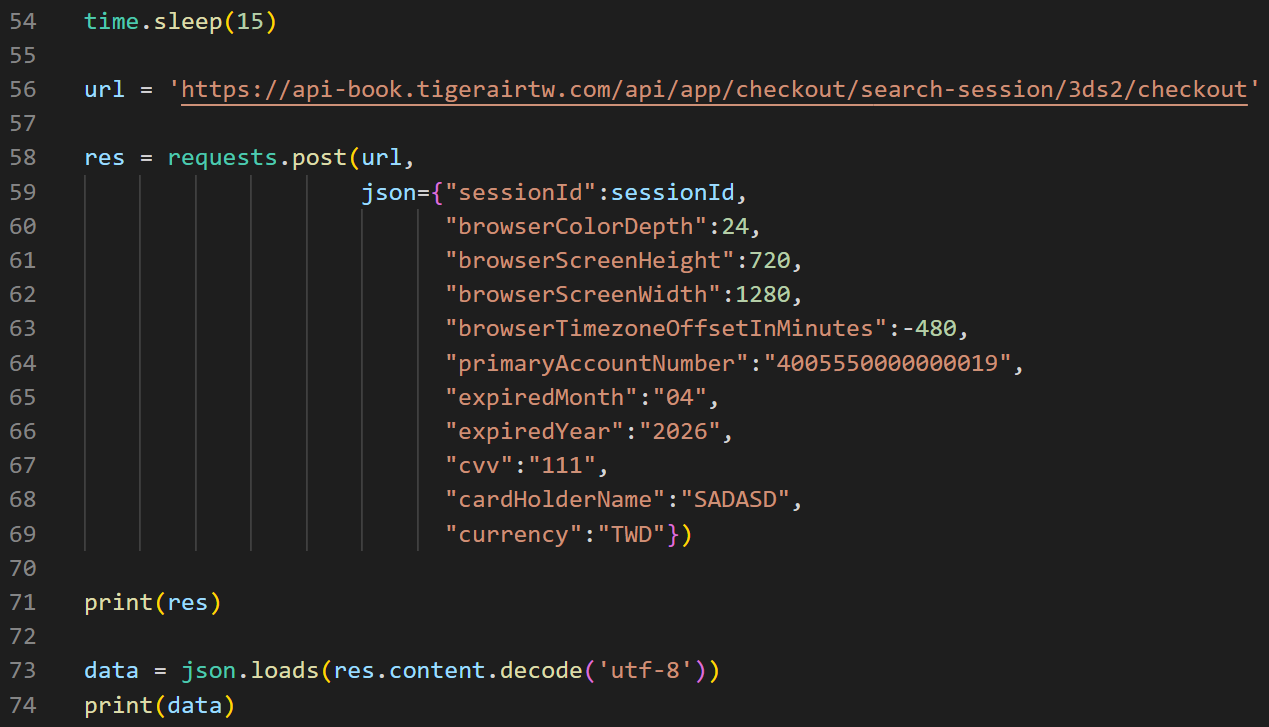
自動產生的描述

這裡當時burp忘記截圖封包，但我也不敢試太多次，因此直接看code，總之結帳頁面最簡單的方式就是信用卡，post data就是信用卡的資料外加一個sessionId，所以也是相當簡易

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

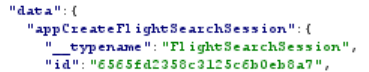
自動產生的描述

並在15秒後(當時觀察網頁送出信用卡資料和檢查所間格的秒數)確認結果，信用卡資料需要重新傳送一次，



**結果：**

我先用burp取得一個sessionId



原本網頁為11/28的航班

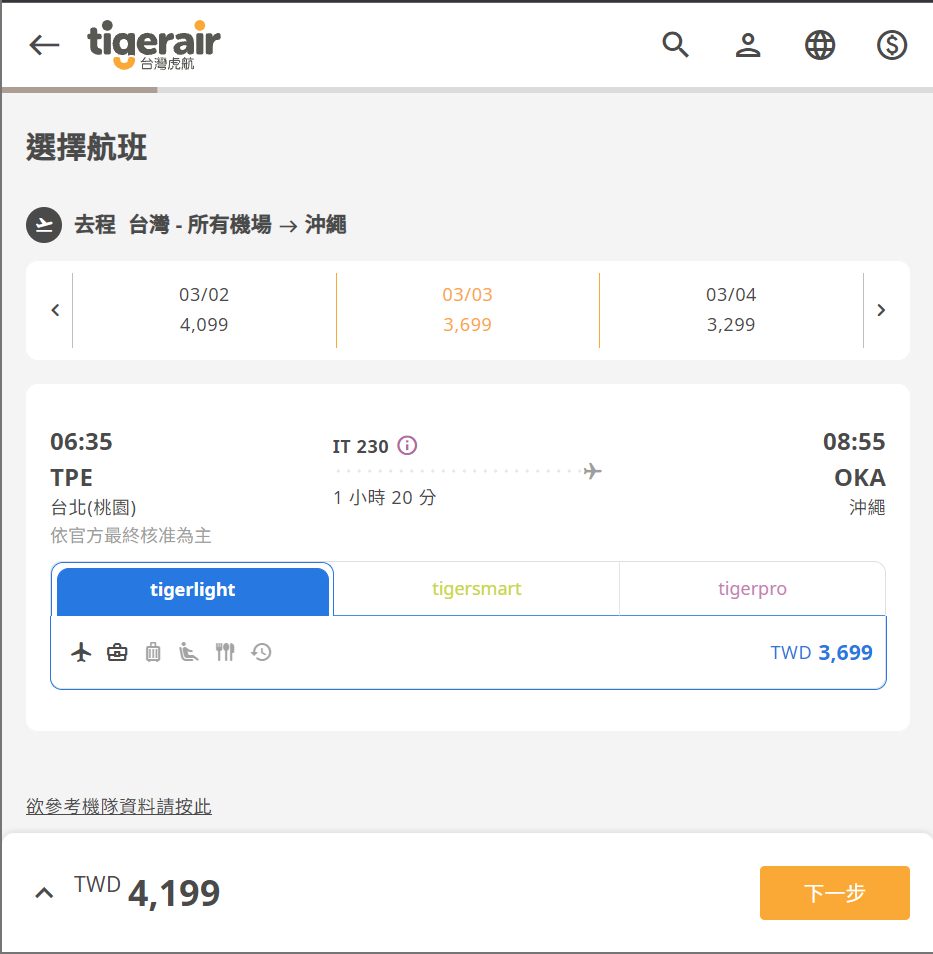
一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

自動產生的描述

經過我偽造的封包(如下資料)後，網頁經過重新整理後發現航班已經選擇完畢了，且和我設計的參數日期吻合，多虧前面所提到的更新後台資料的封包，省去了點擊的時間就自動完成

一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述



原先網站應該是這樣甚麼都沒有輸入，但經過我偽造的封包過後，

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

資料已經自動處理完畢，內容也和我設計的參數都是相同的，

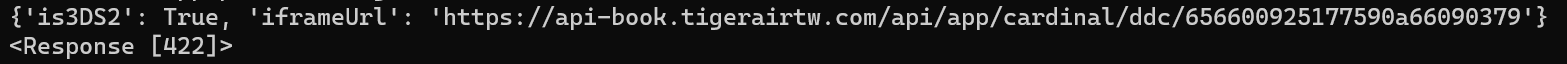
一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 字型, 圖表 的圖片

自動產生的描述

最後進行信用卡結帳，但因為我使用假的信用卡資料結帳，因此會返還422的status code，內容由unicode編碼，內文為「無法結帳」



**心得：**

這周嘗試的以測試為主，後續會和之前的作搭配，像是使用selenium內建的refresh頁面，還有從第一次搜尋後直接去取得一個sessionId而無須手動去抓，相信就可以完成一個非常漂亮的購票程式。

本周的困難點大概就是去分析那些封包，老實說眼睛都快脫窗了，但也獲得好多收穫，大大增強了這個程式的速度。

我想這些功能應該可以看在我辛苦分析封包的勞力上，加個至少20分，再加上這購票功能我想從開始到結帳，可能只需要五秒，希望也可以加個10分。

如果想測試，可以先打開DevTools(F12)的network後進入虎航網站並先搜尋一次任意航班，之後應該可以看到封包名為graphql或heartbeat都會有包含sessionId，複製到test.py這個檔案裏面的sessionId變數後終端執行python3 test.py，就會開始購票了，可以回到網站按下一步或重新整理可以看到偽造的參數已經輸入了，如果需要更改參數都可以到檔案內修改，下周就會交出最終版本了，謝謝老師。