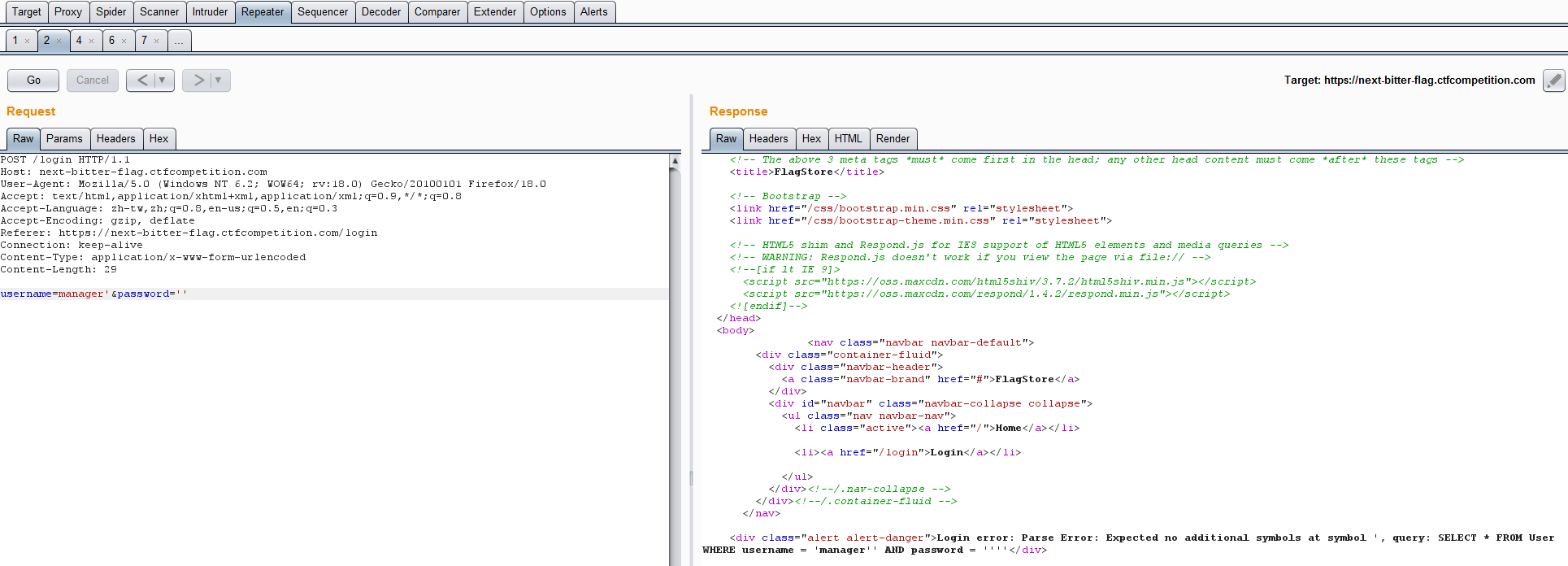
Flag Storage Service

1. 一開始在 /login 的地方輸入 username=' & password='' ，便出現SQL 語法錯誤，從錯誤訊息猜測語法是 SELECT \* FROM User WHERE username = 'ARG\_username' AND password = 'ARG\_passwd'，且從錯誤訊息大概可推出是Google研發的GQL。 ( 參考連結: <https://cloud.google.com/datastore/docs/apis/gql/gql_reference>)

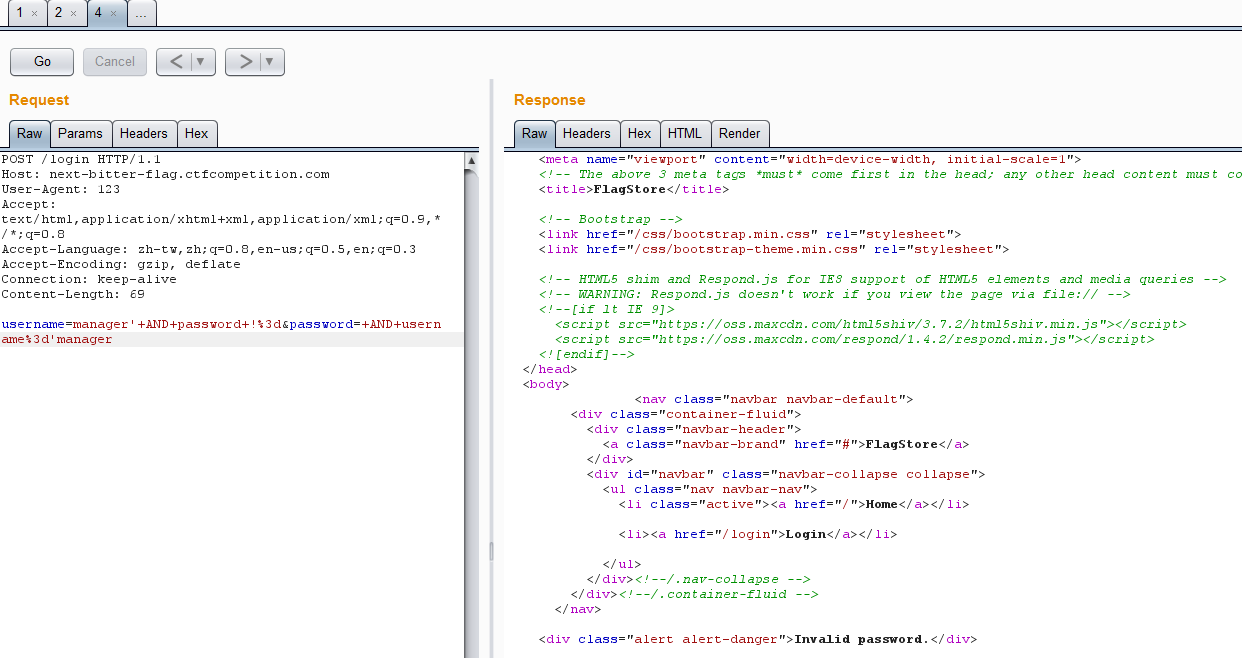


1. 輸入單引號出錯，確認存在特殊字元未處理造成的錯誤，極可能有SQLi。由於是GQL，所以用一般SQL的注入方法無法成功，以下簡要說明GQL的特性造成的問題：
   1. GQL 沒有 OR 可以用，所以無法靠 or '1'='1 來繞過身分驗證
   2. 搜尋一下網路文章，沒甚麼comment符號來斷掉不想要的原生地QUERY語法
   3. GQL 沒有太多函數或是特殊字元可用，但condition形式滿多變有趣的(可能是關鍵的切入點，愈複雜的東西就越容易出錯)
2. 接下來處理QUERY語法的問題。 我希望把語法改成 username = 'manager' AND password !='xxx' 來達到我登入的條件，但是要在原本的語句中插入 ! 不太可能，也不可能利用OR來達到恆TRUE，幾次嘗試後，發現下面方法可以達到目的。

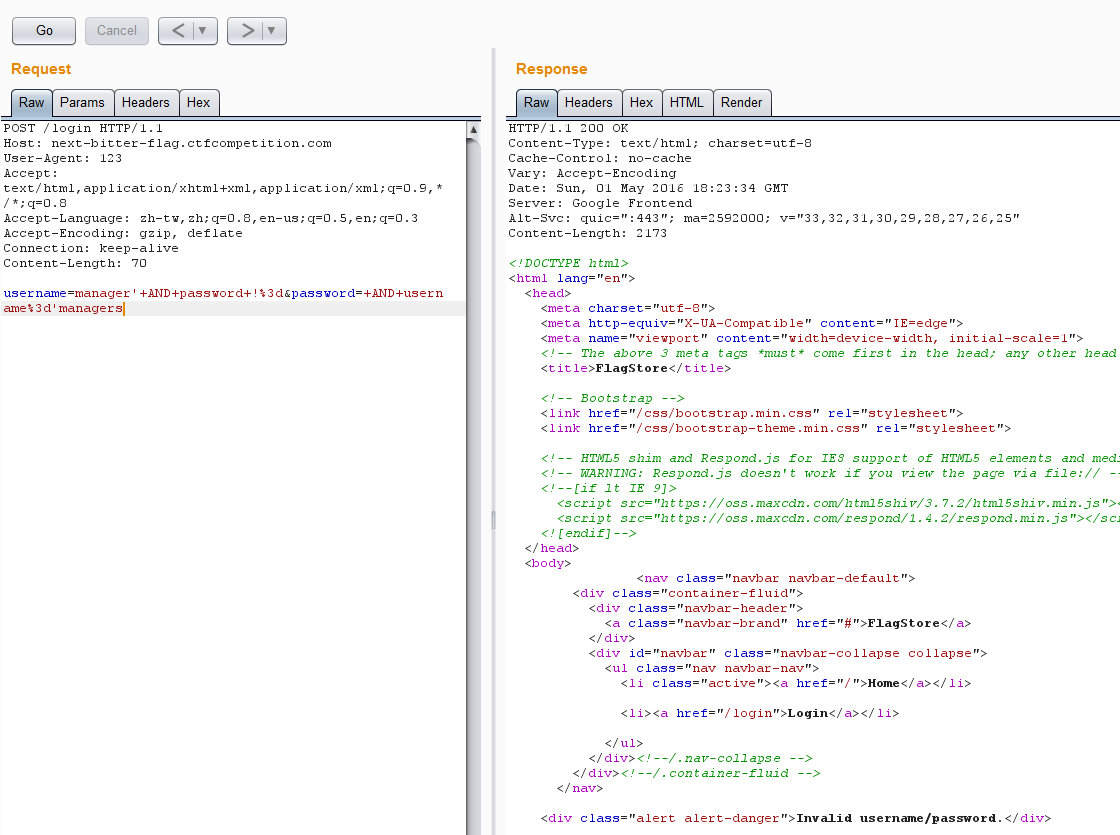
SELECT \* FROM User WHERE username = 'manager' AND password !=' AND password = ' AND username = 'manager'

簡單來說，我把它原本的 AND password = ' 的語句變成了一個單純的字串，並把他給 password != 判斷式，達到這個sub-condition恆True的目的。

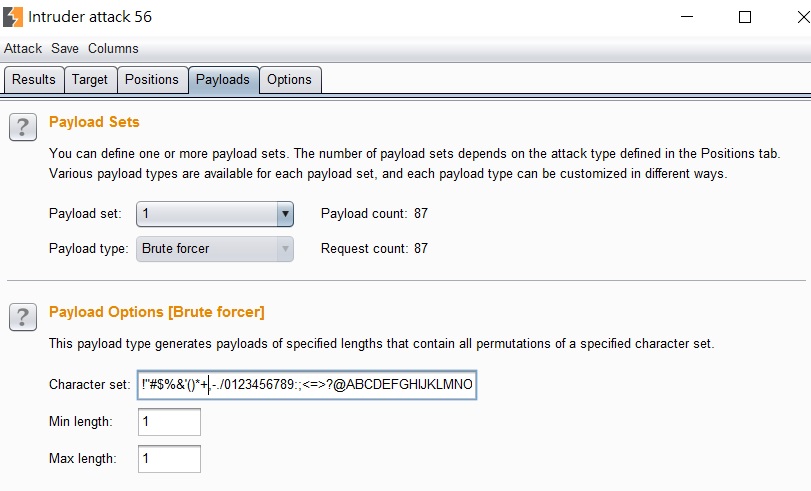
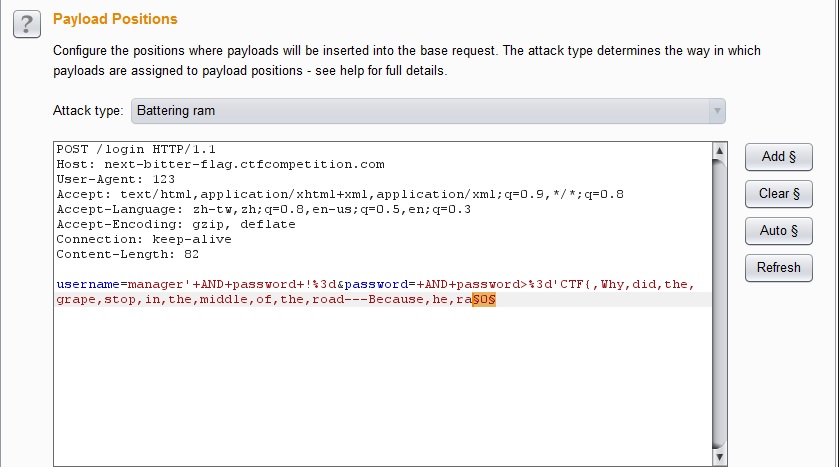
1. 當我用此輸入拿key時，居然出現 invalid password錯誤(囧)，猜測程式可能有對password的內容作非法字串比對，但沒有跳出錯誤訊息代表這個QUERY是可以正常運作的。



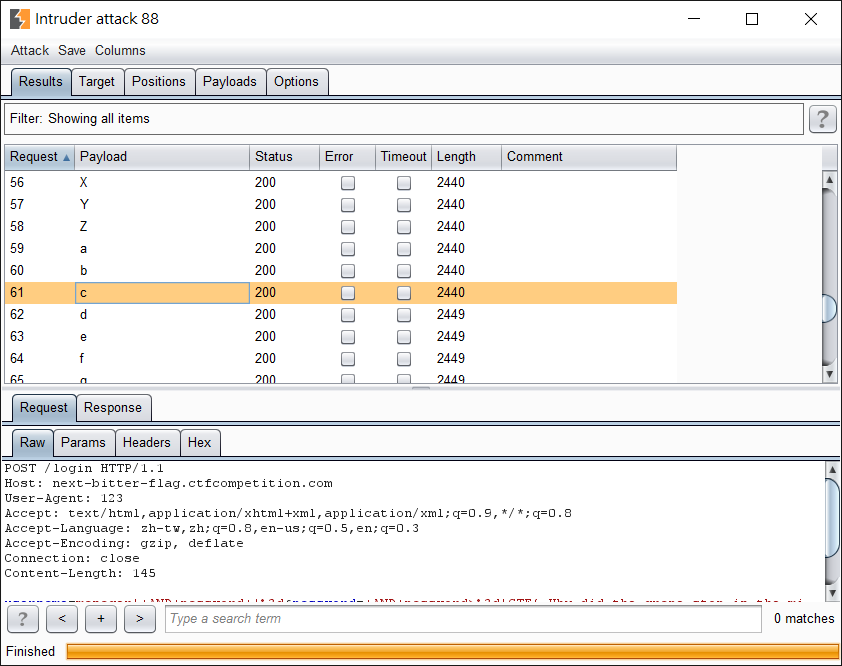
1. 再驗證一下用一個恆FALSE的 condition，結果會出現 Invalid username/password，恭喜你無法登入成功(誤)



1. 這是可以拿來做 boolean- based SQLi的條件!! 現在又回到最一開始的問題，GQL沒有像substring這些函式可用，甚至我要用LIKE 都無法，於是我GOOGLE一下這種情況該怎麼處理，結果在這篇找到一個解法，利用 >= 還有 < 達到字串LIKE的功能。 (https://groups.google.com/forum/#!topic/google-appengine-python/pw6jJdwUOAU)
2. 接下來就是苦功了，我本來以為密碼會很短，所以就只開了burp suite intruder頂一下。payload選用 bruteforce，字元從 32(空白鍵) 列舉到 127 (~) 。然後就從 SELECT \* FROM User WHERE username = 'manager' AND password !=' AND password = ' AND password>= '$a$ ，依序往下，結果沒想到password超長阿 ORZ，再給我一次機會，我應該會寫一個python 用 binary search 加速。



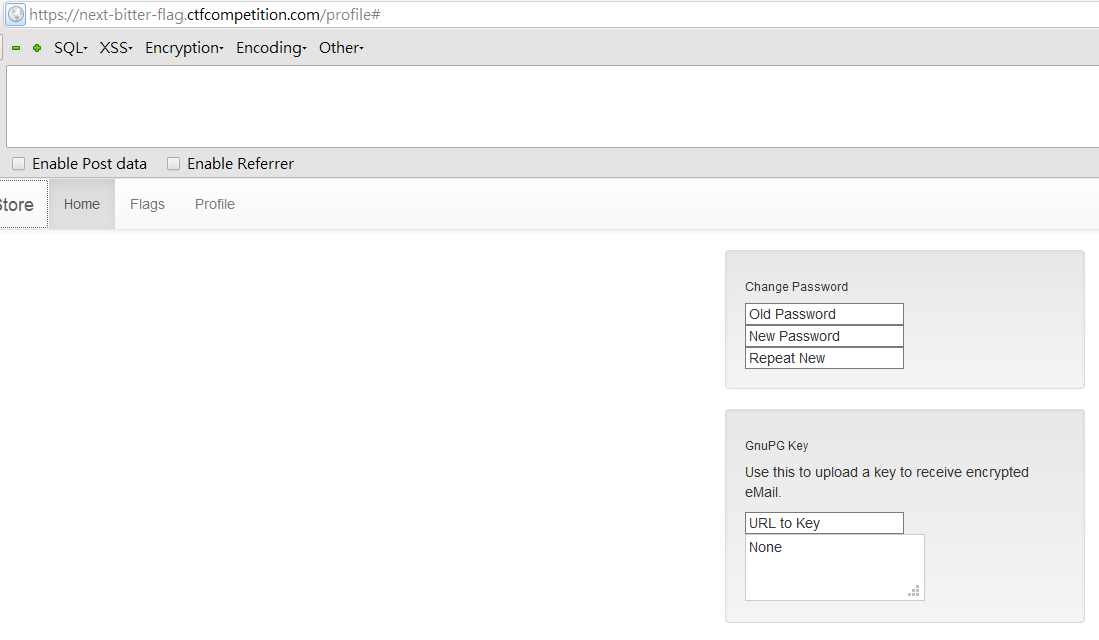
1. 利用回應的長短來判斷內容是甚麼，從2440 -> 2449 的轉界點 (回應 invalid password 跟 invalid username/password 長度的差別)，藉此猜完KEY。



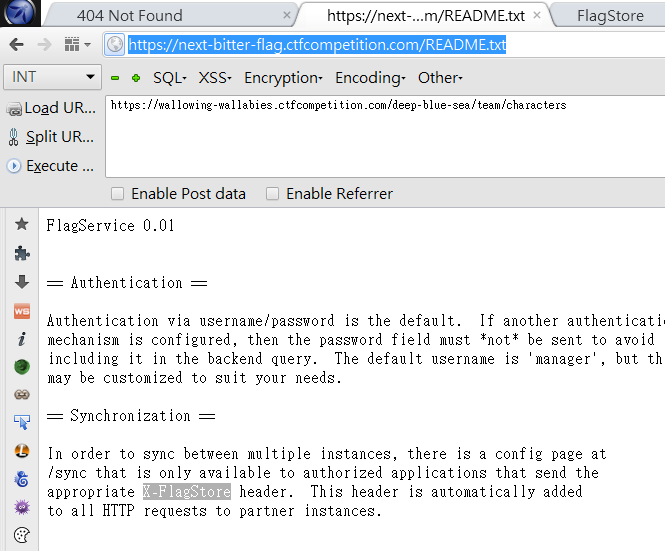
1. 最後獲得 CTF{,Why,did,the,grape,stop,in,the,middle,of,the,road---Because,he,ran,out,of,juice,}，也就是這一題的Key

FSS – Electric Boogaloo

1. 這題倒也簡單了，是上一題的延伸，先用此帳號/密碼登入，可以看到一個介面。 看到這畫面其實滿容易被誤導到RFI 或 command injection去，但這題沒這麼難。



1. 一個PT工作者在RECON時，看一下 robots.txt是很合理的，看一下發現有一個README.txt ，從裡面可以發現有一個叫做 /sync的檔案，但是要X-FlagStore標頭設對才能存取，這其實我在解上一題就挖出這個了，只是一直沒用到他，現在幾乎可以確定這一定是第二題的題示，可是X -FlagStore的值從哪邊來呢?



1. 疑 !? Server端會不會有，阿 PROFILE的頁面是不是可以輸入網址，假設SERVER有此Header，我又讓他自己存取自己的 /sync檔案，我是不是就有機會讀出內容了呢!
2. Bingo! 獲得 key CTF{@What@did@the@father@tomato@say@to@the@baby@tomato@whilst@on@a@family@walk---Ketchup@}

