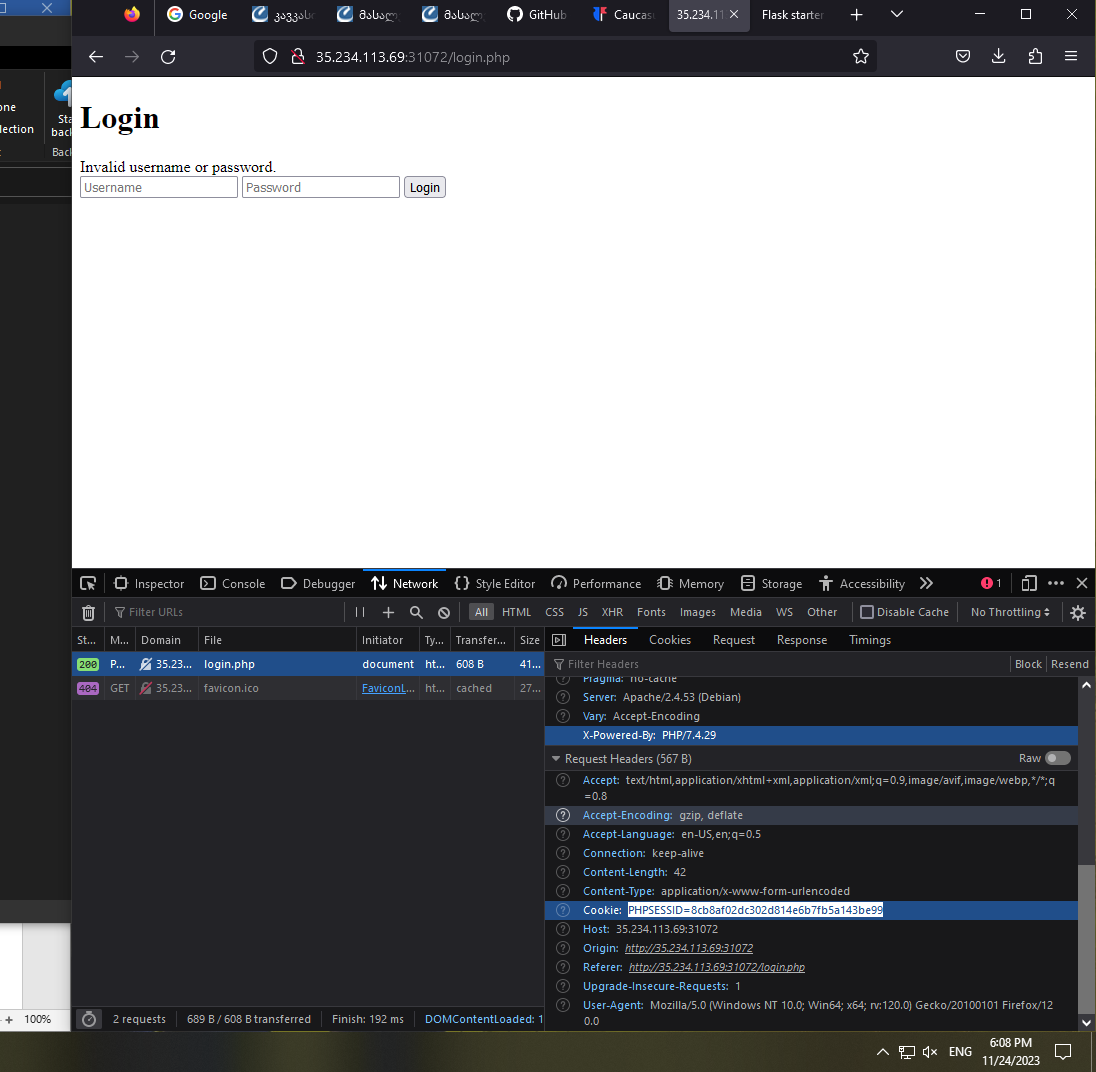
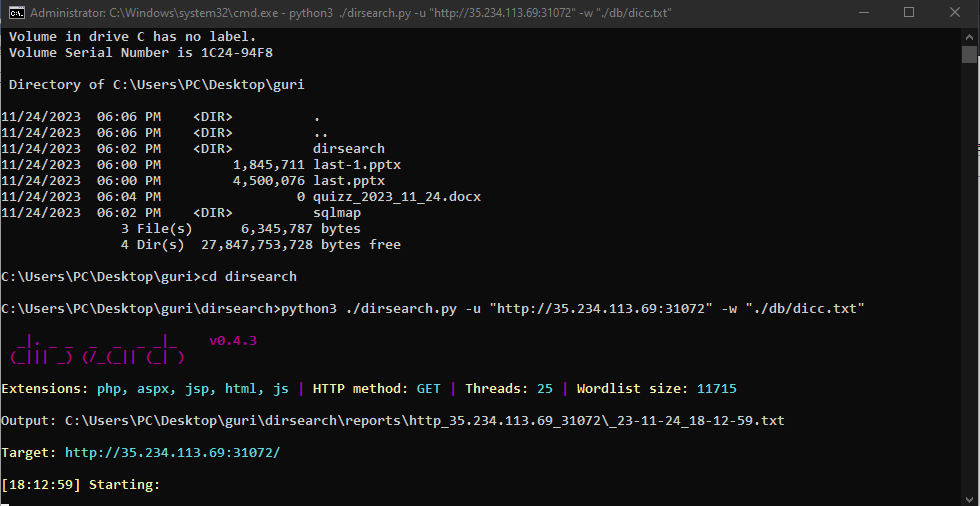
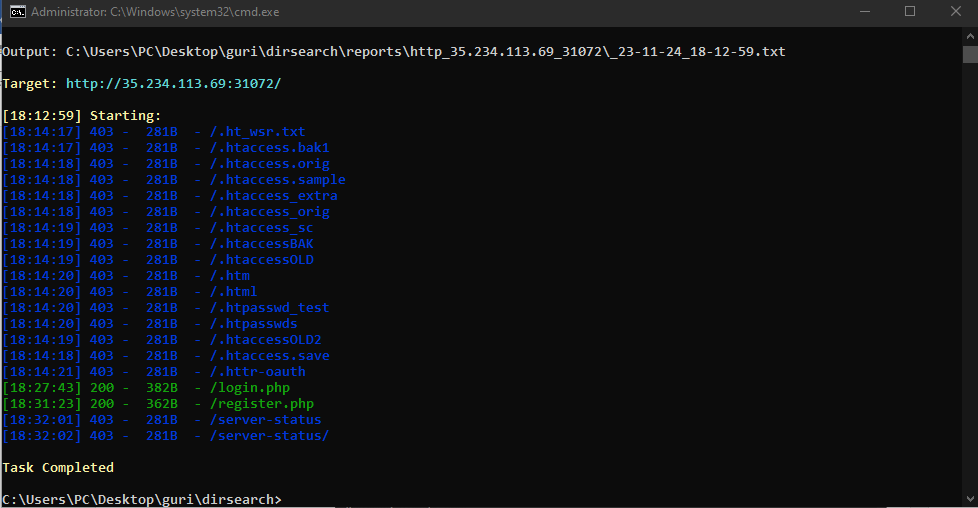
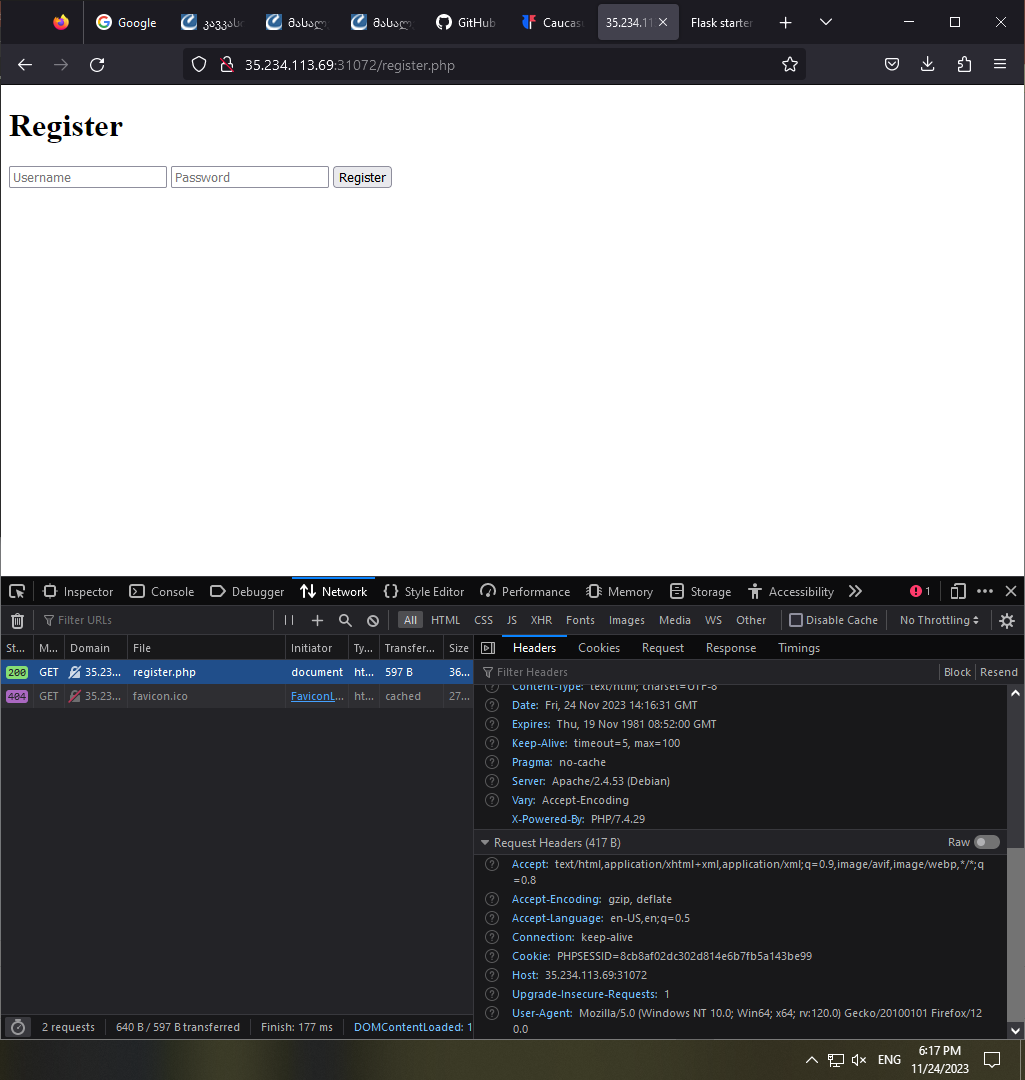
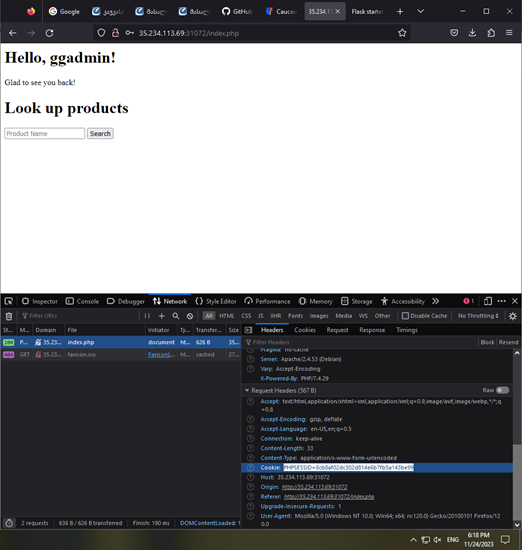
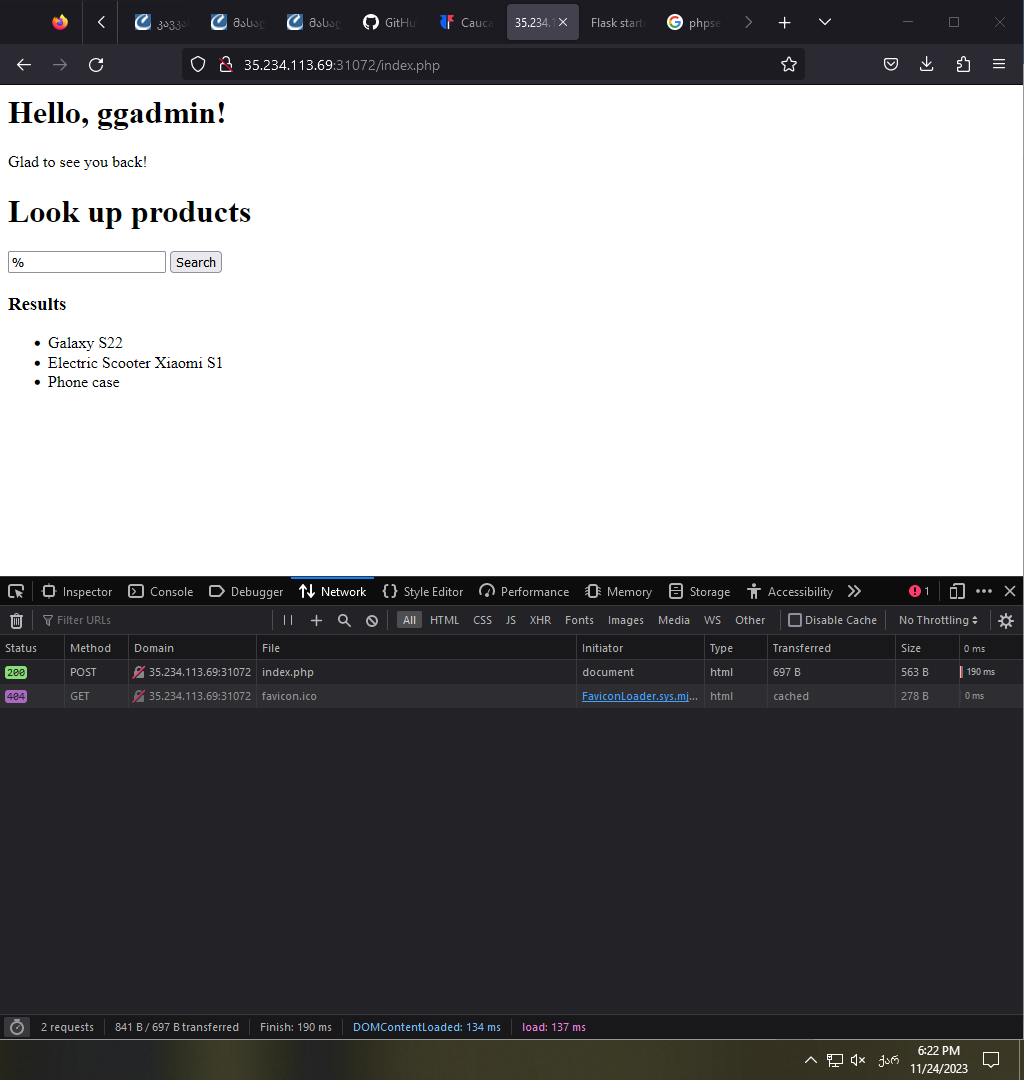
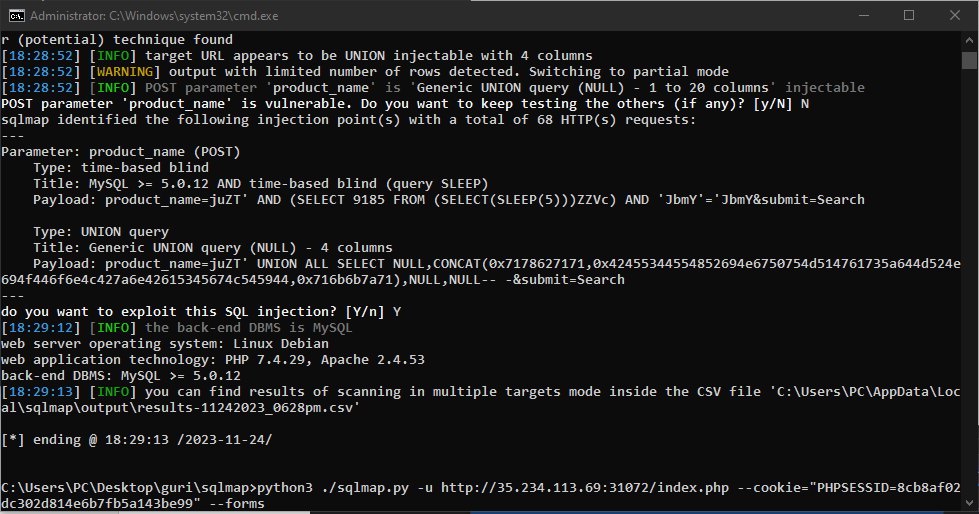
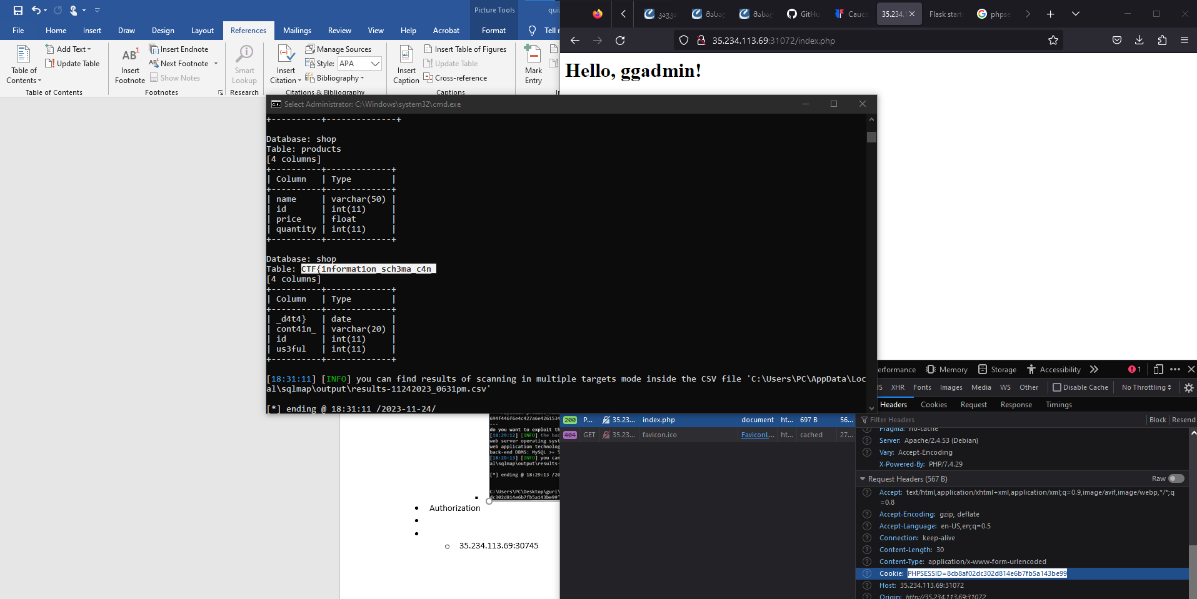
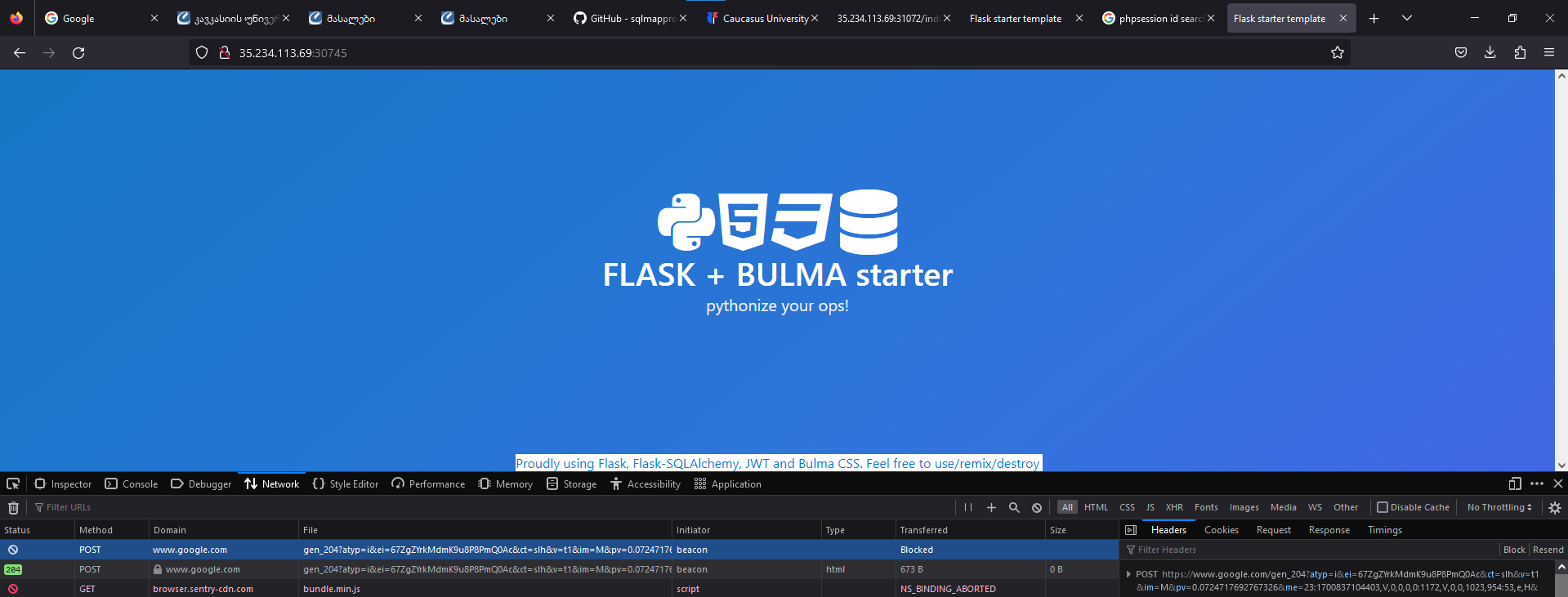
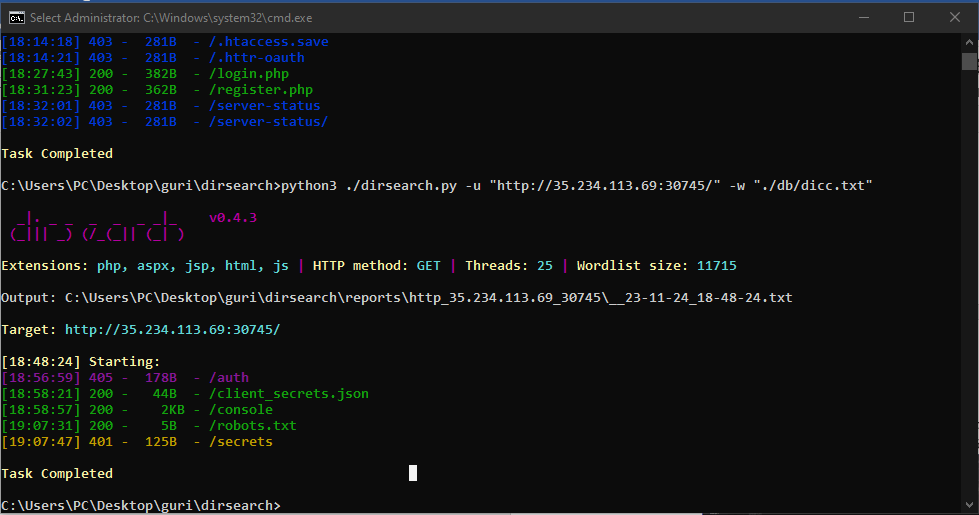
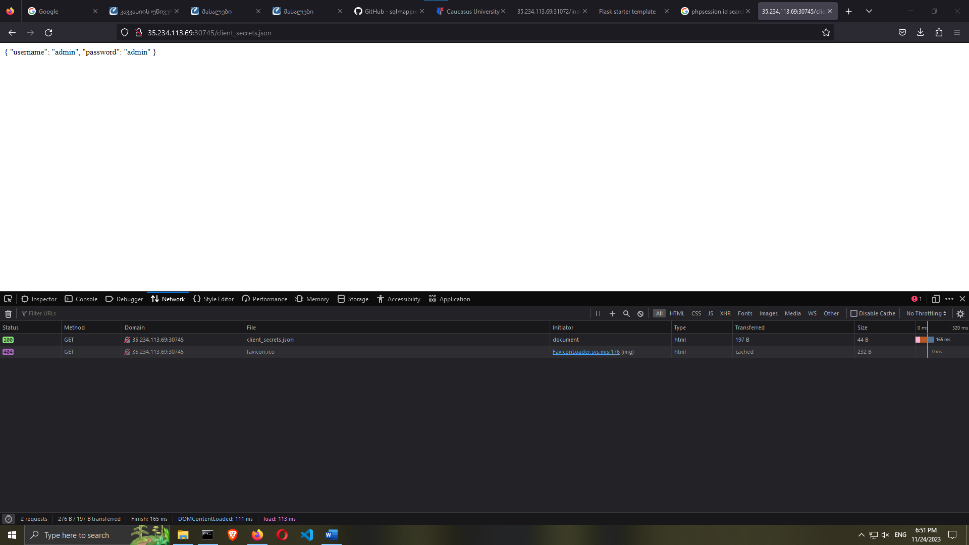
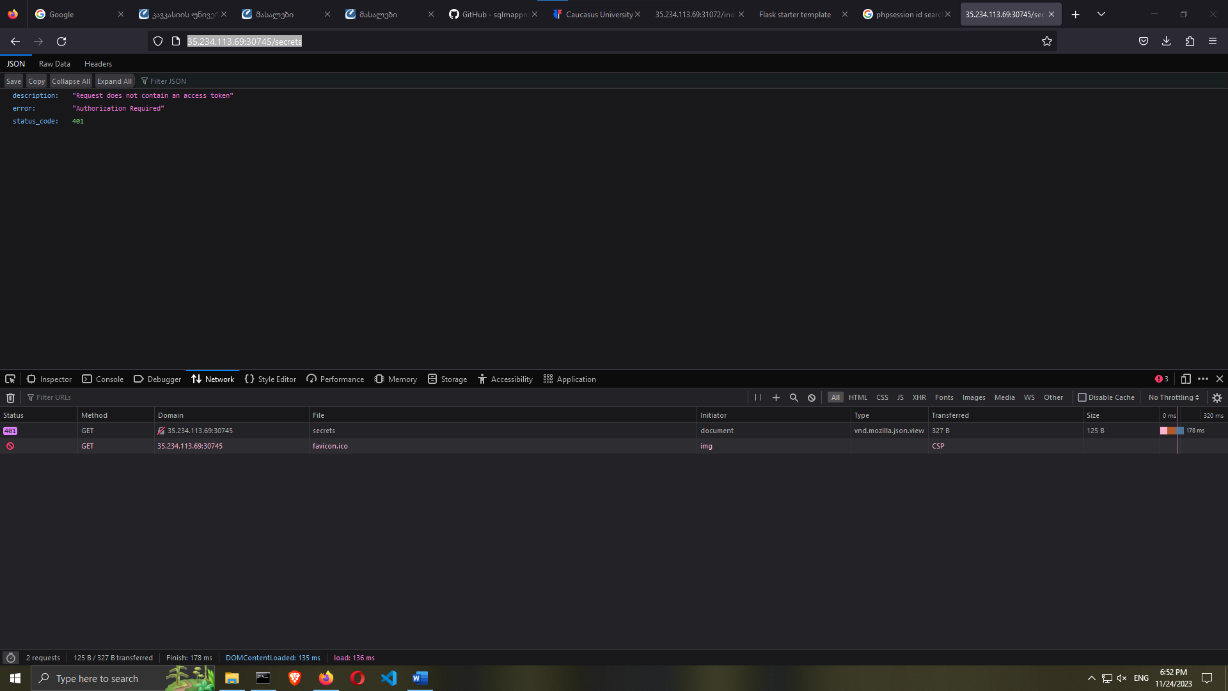
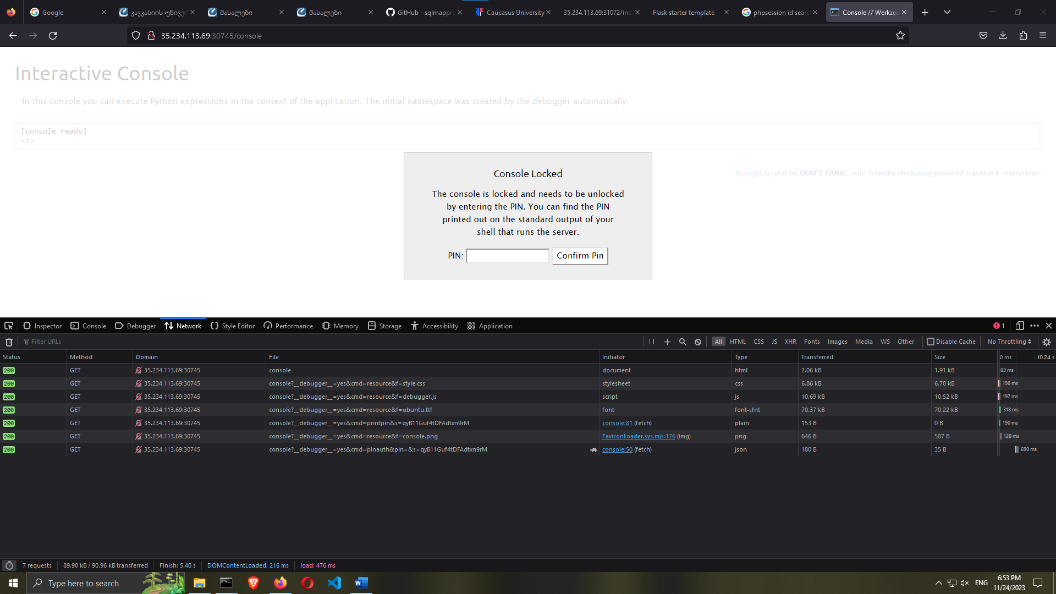
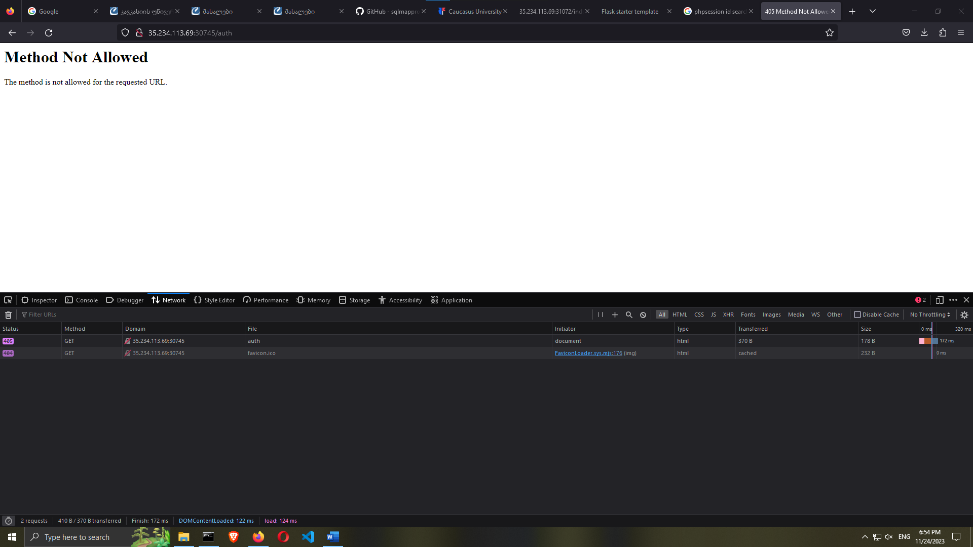
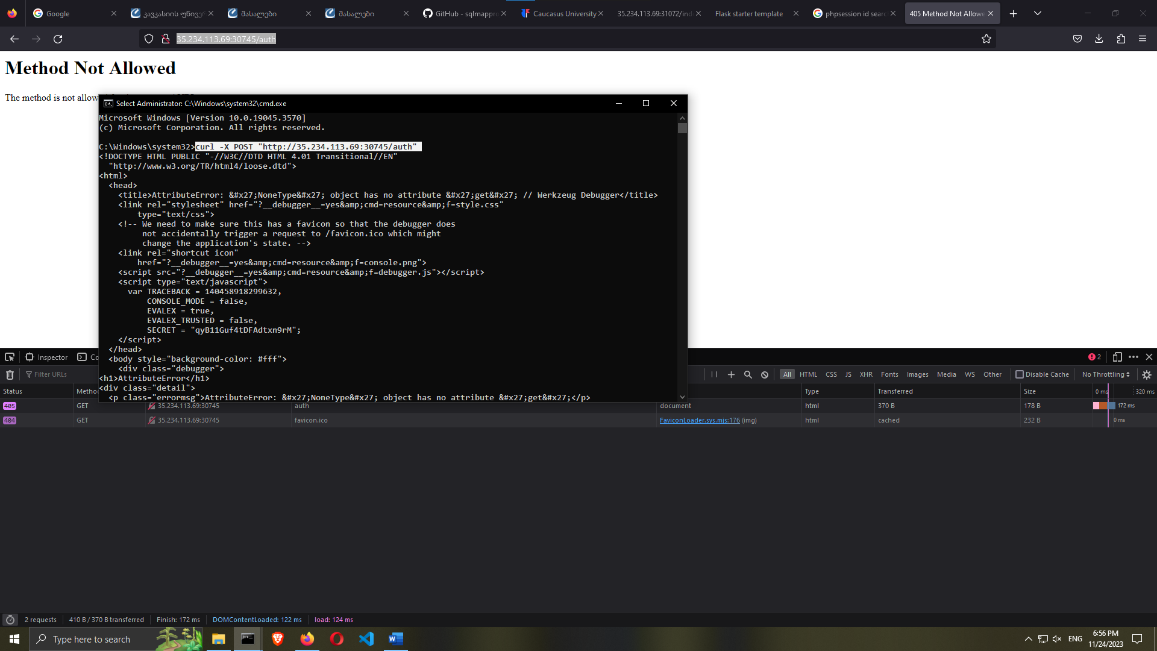
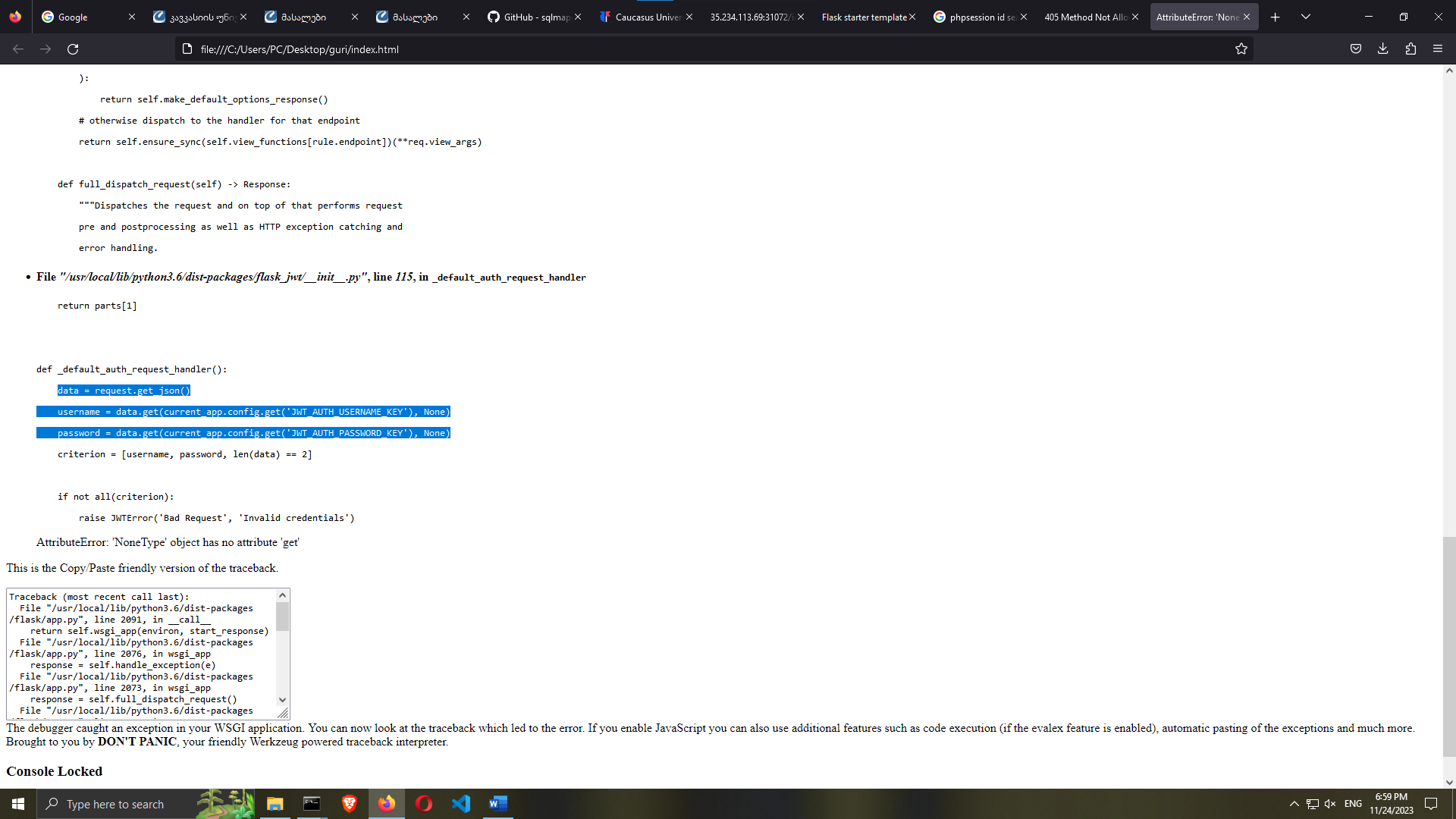
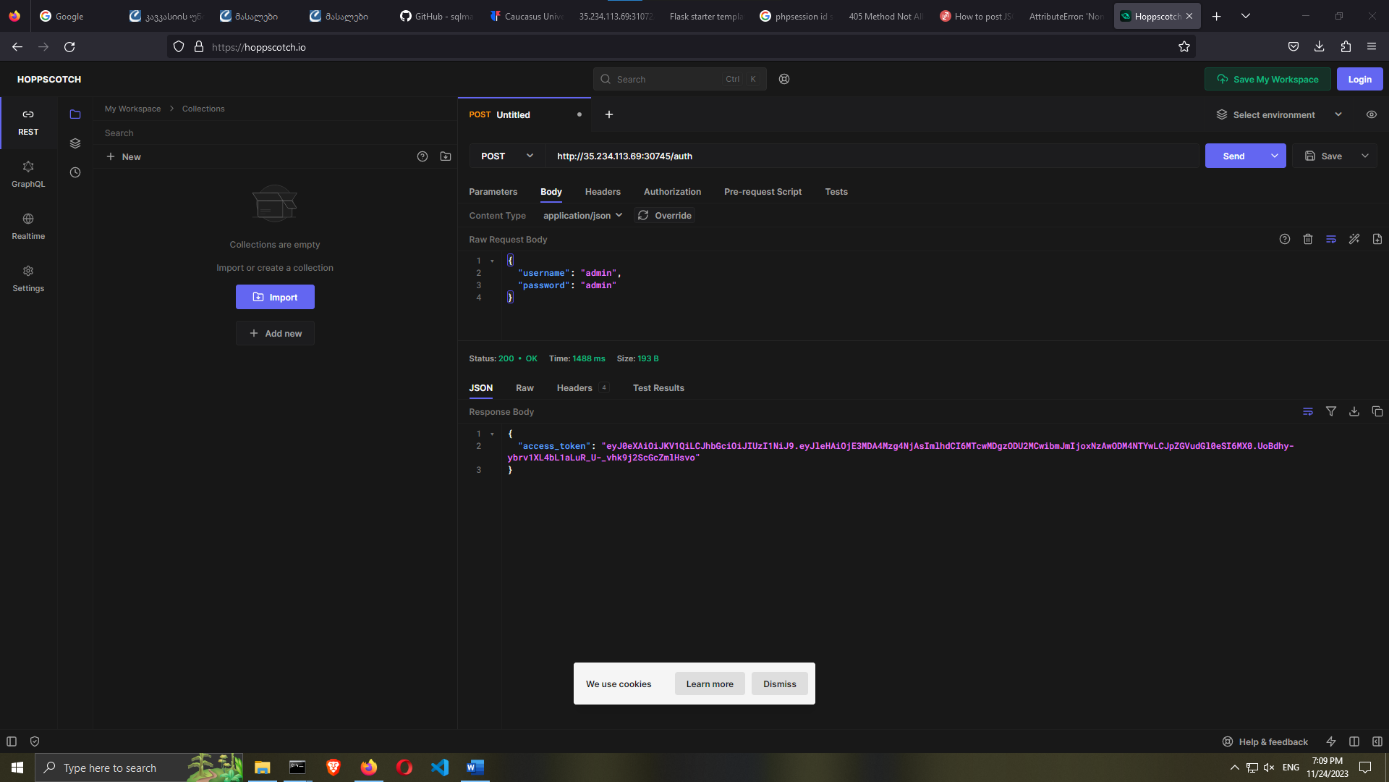
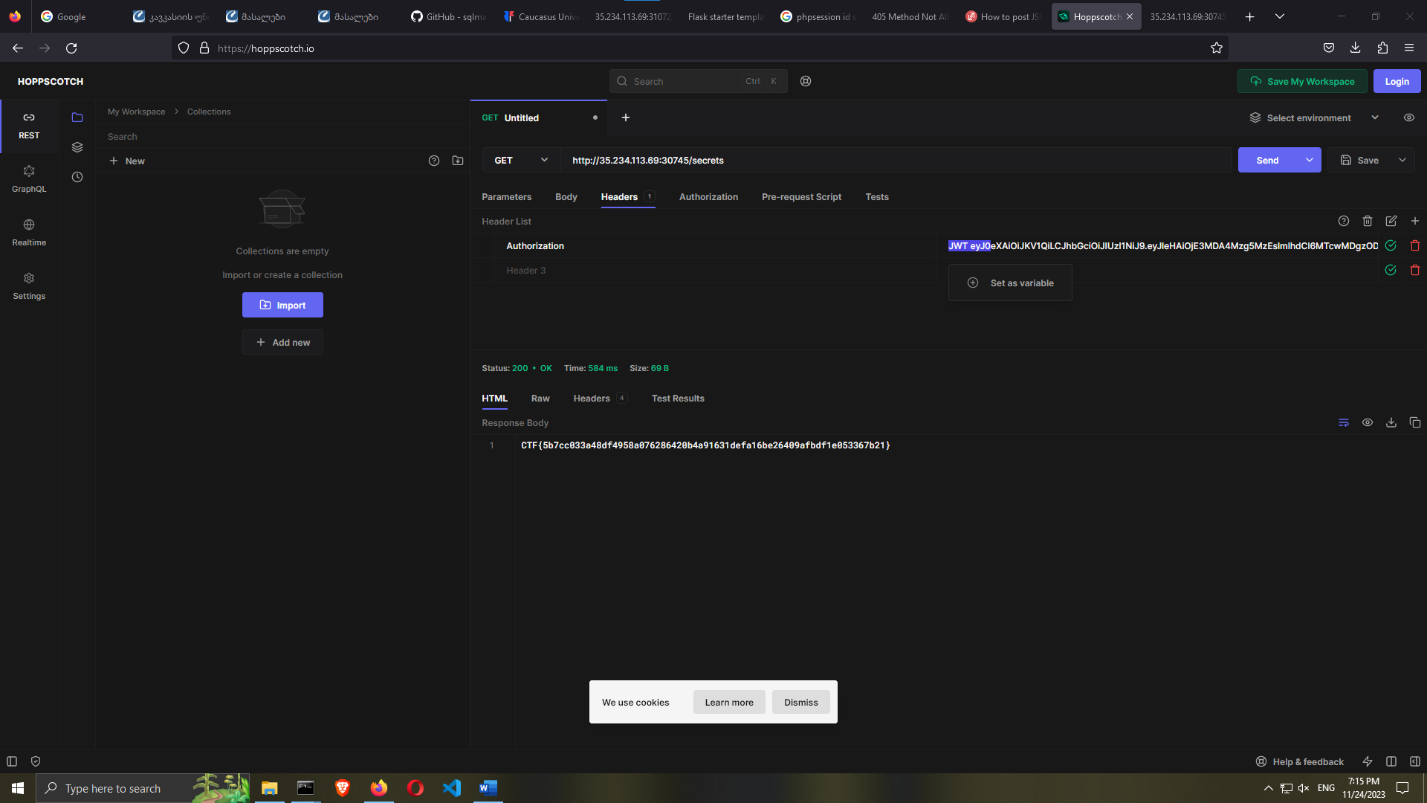
* Schematics
  + 35.234.113.69:31072
  + ვუყურებთ გვერდს, ვფიქრობთ რაიმე ლოგინი და პაროლი ხომ არ შეიძლება გავარტყათ, ეს წარუმატებლად სრულდება, მაგრამ შეგვიძლია რექვესთს გავყვეთ, ნუ აქ ჩანს php ვერსია და რაღაც php-ის სესიის აიდი, ნუ ამის გამოყენება შეგვიძლია
  + 
  + რახან გავრცელებულიცაა მარტივად დავაბრუტფორსებთ ვორდლისტით და dirsearch-ით:
  + 
  + ნუ, ამან შედეგი გამოიღო, ღიაა რეგისტრაციის ფორმა:
  + 
  + 
  + დავრეგისტრირდეთ და წესით რაიმე საინტერესოსკენ მივდივართ უკვე
  + ვლოგინდებით და ვპოულობთ პროდუქტების სერჩბარს და ვამჩნეთ რომ სესიის აიდი შეიცვალა:
  + 
  + ნუ სადაც სერჩბარია და სადან ქივორდი „products“, იქვეა sql injection-ის vulnerability, დავტეტოთ და დავრწმუნდეთ: % ვცადოთ, და სხვადასხვა მზა ექსპლიტები რაც დაგვარწმუნებს იმაში, რომნამდვილად მოწყვლადია:
  + 
  + ნუ რამდენიმე მუშაობს. Sql vulnerability + php = sqlmap
  + გასათვალისწინებელია, რომ აქ ერთი ვარიერი გადავლახეთ და დავლოგინდით, სერერმა კი ქუქიში სესია მოგვცა, ამის გამოყენებაც მიზანშეწონილი იქნება:
  + ნუ მოვსინჯოთ სხვადასხვა კონფიგურაციით:
    - 
    - დავამატოთ columns და ვნახავთ საინტერესო ცხრილს:
    - 
    - ეს აშკარად ფლეგის დასაწყისია, ჩავიხედოთ ამ ცხრილში

და ვაწყობთ ფლეგსაც სვეტებით:

**CTF{1nformat1on\_sch3ma\_c4n\_cont41n\_us3ful\_d4t4}**

**აი ფლეგიც**

* Authorization
  + 
  + ნუ მინიშნება გვხვდება ფლასკზე და JWT-ზე, ნეთვორკში საინტერესო არაფერია, ამიტომ ვიწყებთ დირექტორიების ბრუტფორსით:
  + 
  + ვიპოვეთ რამდენიმე საინტერესო სახელის მქონე ენდფოინთი დადავტესტოთ:
  + ნუ ერთზე ვიღაცის credential-ები დევს: 
  + არის secrets, რომელზეც წვდომა არ გვაქვს:
  + არის კონსოლი
  + 
  + რომლის პინიც ასევე არ ვიცით
  + საინტერესოა კიდევ: 
  + ეს მინიშნება, რომელიც გვეუბნება, რომ სხვა მეთოდით მივაკითხოთ. ვცადოთ post: 
  + ნუ ასე წაკითხვა გაჭირდება და ვნახოთ მარკაპში, შევქმნათ html-ის ფაილი და შიგნით ჩავწეროთ ეს კონტენტი:
  + **curl -X POST "http://35.234.113.69:30745/auth" > index.html**
  + აქ ვხედავთ, რომ ფლასკის \_\_init\_\_ ფაილი ელოდება json-ს:
  + 
  + Username და password ვიცით, ერთ-ერთ ფაილში ეწერა, რომელში იქამდე ჩავიხედეთ
  + curl -X POST url -H 'Content-Type: application/json' -d '{"username":"admin","password":"admin"}'
  + ან postwoman:
  + 
  + საინტერესოა, ტოკენი დაგვიგენერირა, ახლა შეგვიძლია ამის გამოყენებით იმ secrets path-ზეც ჩავიხედოთ, გადაცემის მექანიზმი მარტივია, მინიშნება იყო რომ jwt-ზე საუბარი და სტრუქტურითაც jwt-ია, ვცადოთ ვარიანტები:
  + Postwoman თავად გვთავაზობს jwt-ით ავტორიზაციის ჰედერის შაბლონს (ვაკეთებთ ჰედერში, რადგან დეფოლტად ბრაუზერი get-ით ადგება):
  + 
  + და ფლეგიც ბრუნდება - **CTF{5b7cc033a48df4958a076286420b4a91631defa16be26409afbdf1e053367b21}**