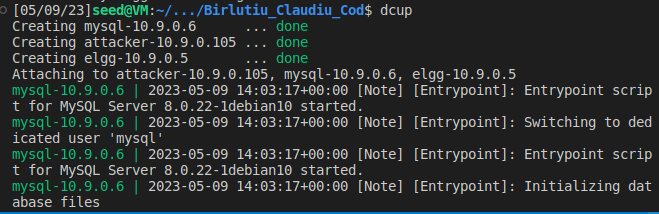
# Raport pentru lucrarea 9: Atacul asupra DNS local

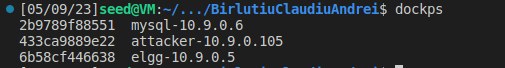
# Autor: Birlutiu Claudiu-Andrei

## Sarcina 1: Observarea cererii HTTP

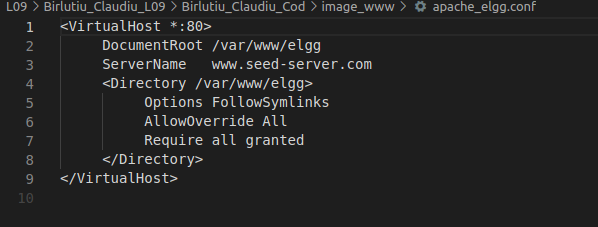
Captura unei cereri HTTP GET și una POST în Elgg.

* În prima faza am construit și pornit containerele necesare desfasurarii laboratorului

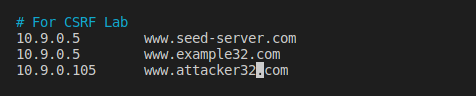


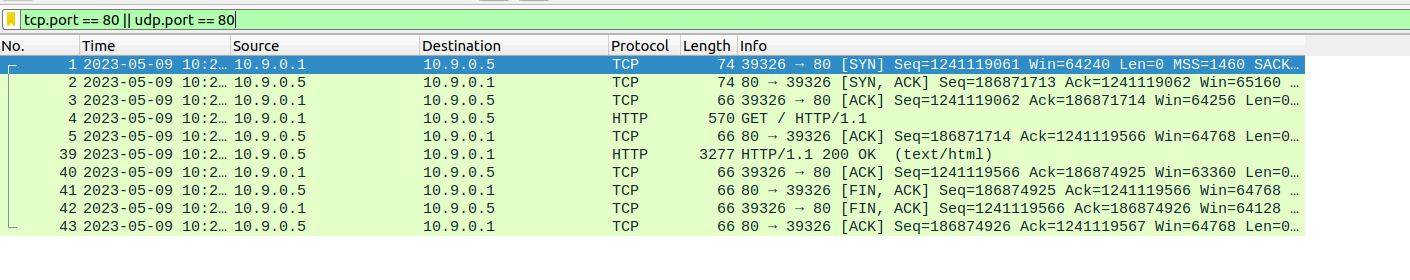


* observam ca sunt în execuție cele 2 servere: **Elgg** (adresa 10.9.0.5) și **mysql**(10.9.0.6)
* aplicația elgg este gazduita de serverul web Apache după cum observam



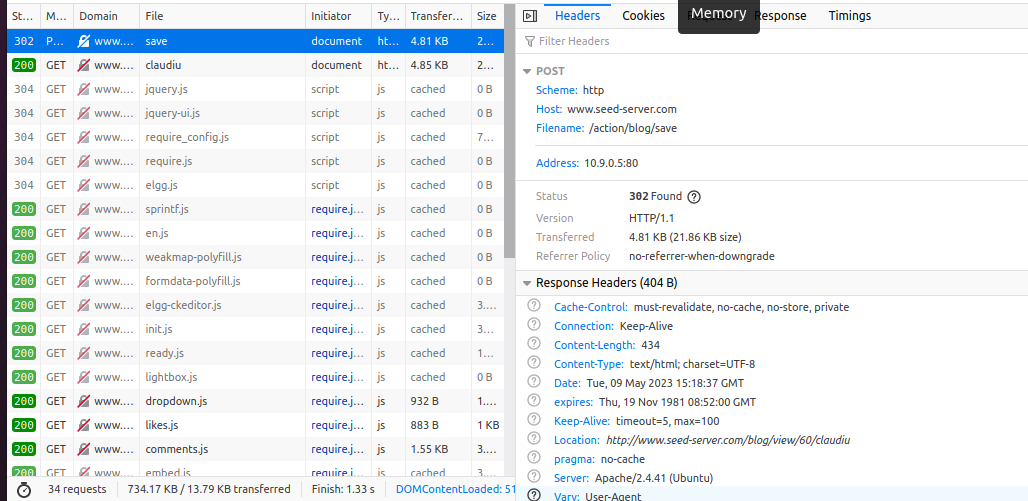
* site-ul rău intenționat este susținut prin containerul attacker care ruleaza pe ip-ul 10.9.0.105
* modificam intrările din etc/hosts precum am configurat containerele



* accesam din browser [**www.seed-server.com**](http://www.seed-server.com/)și am capturat în wireshark traficul tcp pe portul 80
* din Web Live observam o cerere GET executata cu succes și cum arata aceasta împreuna cu headerele

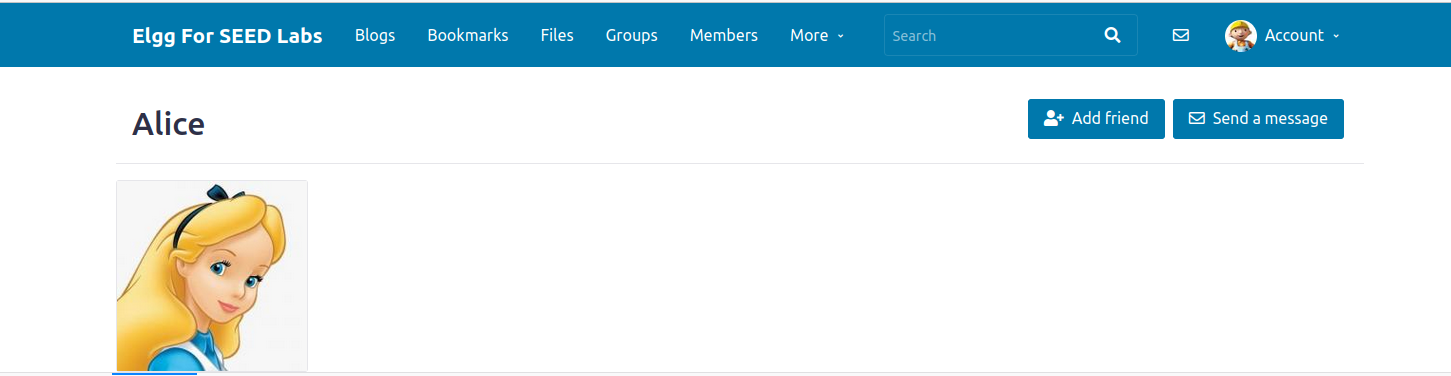


* pentru post am încercat login-ul pe un cont de admin cu
  + - username-ul: **admin** și parola: **seedadmin**
* de remarcat este faptul ca **Content-Type** este un *form-data* și **accept** *application/json*
* în exemplele de mai sus am observat headerele atasate cererilor HTTP pentru cele 2 tipuri: get și post
* cu developer tool -> network am urmărit headerele din cadrul adăugării unui blog



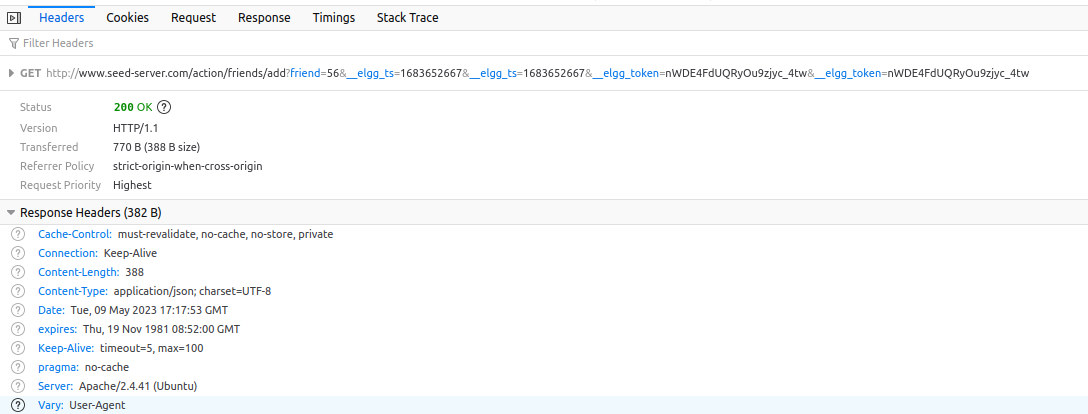
## Sarcina 2: Atac CSRF folosind cererea GET

* Scopul este de a-l adauga pe Bobby listei de prieteni ai lui Alice fără ca aceasta din urma sa își dea consintamantul -> prinț intermediul unui atac CSRF
* se va construi un link astfel încât Alice să fie redirectionata spre pagina atacatorului; ea trebuie sa aibă o sesiune Elgg deschisă în browser ca atactul sa funcționeze;
* pentru început am investigat cum arata o cerere de **add friend**

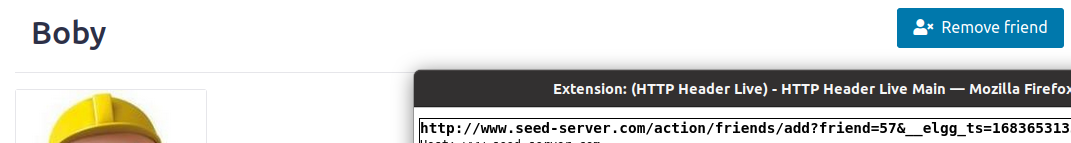
****

* cererea arata în felul urmator
  + - http://www.seed-server.com/action/friends/add?friend=56&\_\_elgg\_ts=1683652479&\_\_elgg\_token=BaFvpwErogzNHfMAF8iAoA&\_\_elgg\_ts=1683652479&\_\_elgg\_token=BaFvpwErogzNHfMAF8iAoA

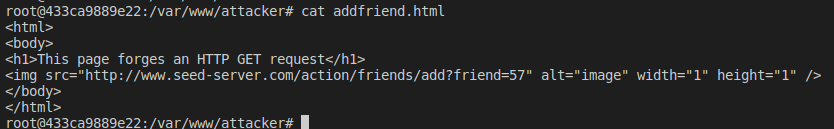
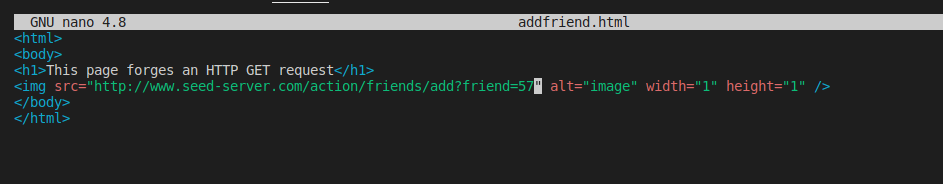




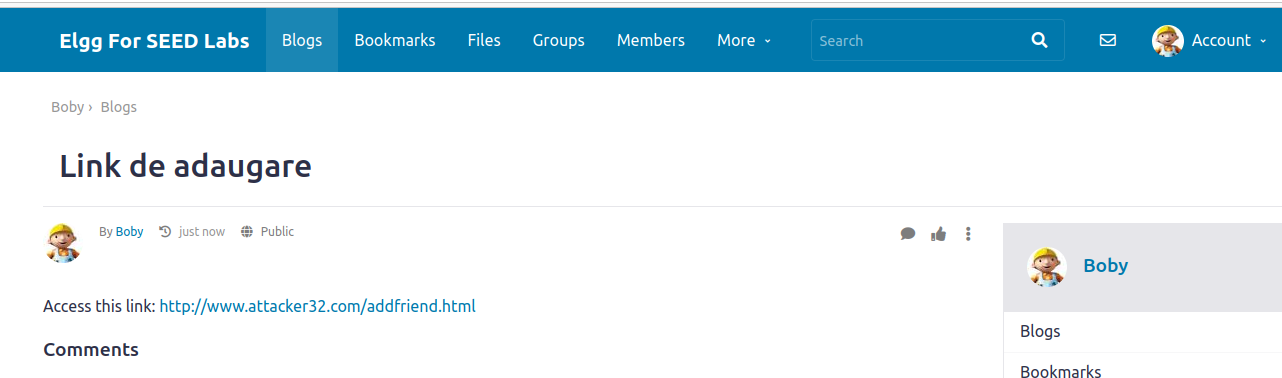
* luam link-ul: <http://www.seed-server.com/action/friends/add?friend=56> ;
  + - 56 – e id-ul lui Alice; pentru a afla id-ul lui Bobby putem să ne logam pe contul lui Alice și îl aduagam pe Boby în lista de prieteni – urmărim în http live header și vedem ca id-ul lui Boby este : 57



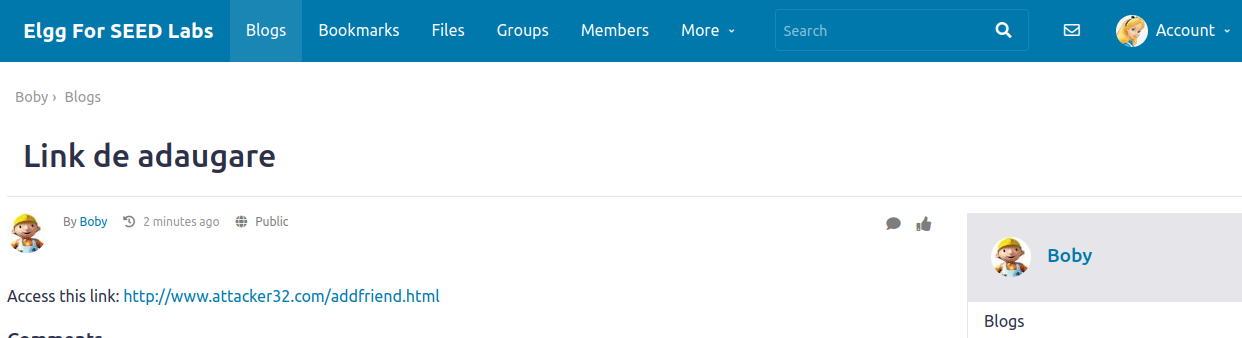
* link-ul pentru adugarea lui Boby(fara paramateri legați de token și elg ts) este:
  + - [http://www.seed-server.com/action/friends/add?friend=5](http://www.seed-server.com/action/friends/add?friend=56)7 ;
* vom adauga pe pagina atacatorului addfriend.html linkul acesta încorporat într-un tag de **img** cum este recomandat pentru a se executa cererea **GET** se poate observa mai jos; ne conectam pe containerul atacatorului



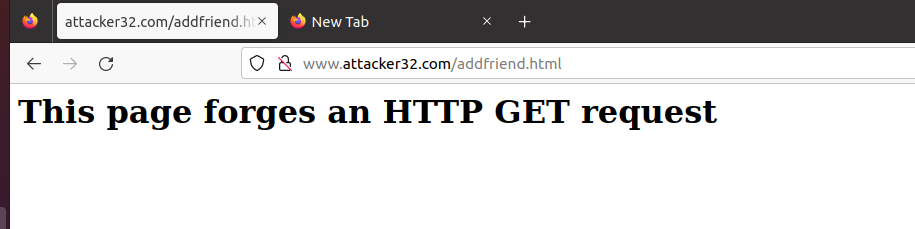
* Boby posteaza link-ul spre pagina rău intenționată astfel încât Alice sa aibă acces la el; de exempu posteaza un blog cu linkul: <http://www.attacker32.com/addfriend.html>



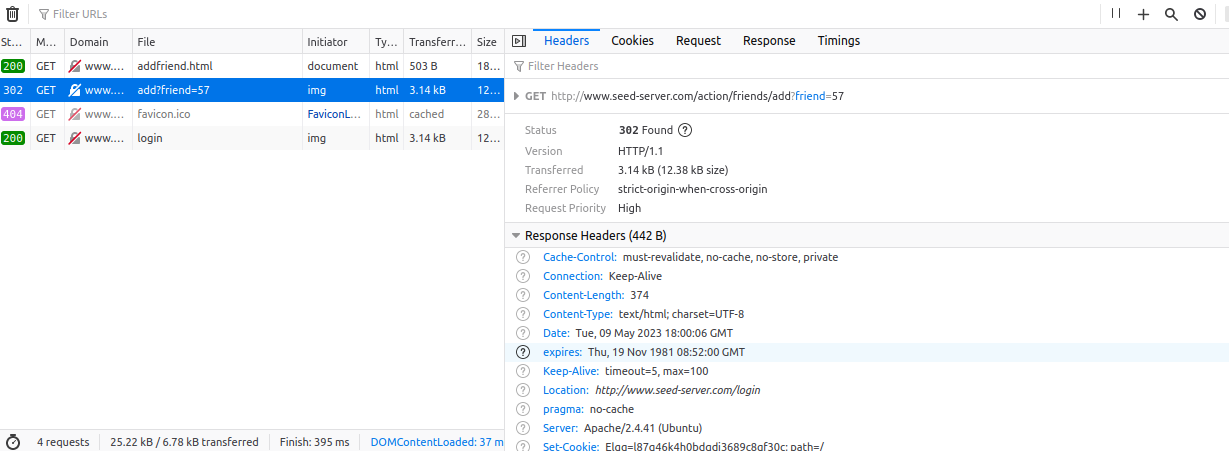
* ne logam cu contul lui Alice și accesam linkul



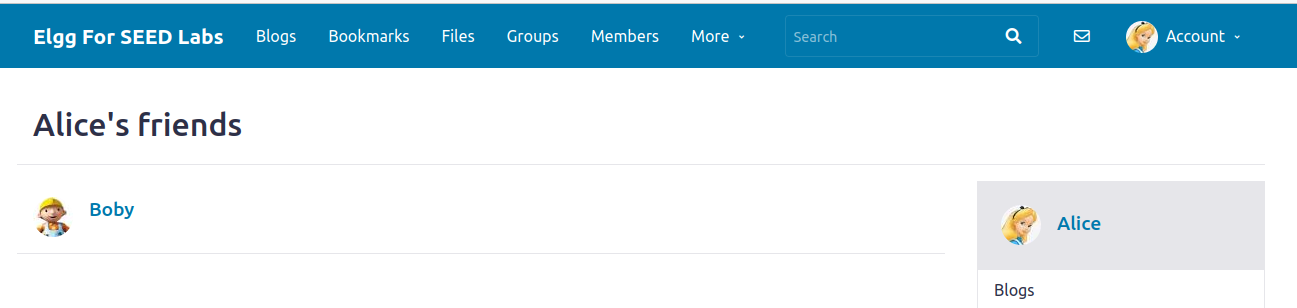
* ne va redirectiona spre:



* observam ca s-a executat cererea GET de adaugare prieten:



* și când ne întoarcem înapoi pe contul lui Alice observam ca Boby a devenit prietenul ei



## Sarcina 3: Atac CSRF folosind cererea POST

**Întrebarea 1**

**Întrebarea 2**

## Sarcina 3: Implementarea unei contramăsuri pentru Elgg