

Korp\_station

Writeup pour Fancy Blog



The image shows a challenge interface with a dark background. At the top, there's a header with 'Challenge' in a red-bordered box and '26 Solves' next to it. The main title 'Fancy Blog' is in large white font, with '200' below it. The challenge is presented in two languages: French and English. The French text describes a discussion about infrastructure attacks and asks for an audit. The English text is a translation of the French one. At the bottom, there's a 'Flag' input field and a 'Submit' button.

Challenge 26 Solves

## Fancy Blog

### 200

[FR]

La discussion que nous avons découverte porte sur des attaques contre des infrastructures.

Réalisez un audit des applications de ces infrastructures afin d'identifier les vulnérabilités potentielles avant qu'elles ne soient effectivement exploitées par les pirates.

[EN]

The discussion we uncovered revolves around attacks on infrastructures.

Conduct an audit of these infrastructure applications to identify potential vulnerabilities before they are actually exploited by hackers.

<http://qualif.hackerlab.bj:4500>

Author: r3s0lv3r

Flag Submit

Le lien nous mène vers un site nous m-ne vers un site simple avec trois pages index,articles et contact.html.

Ce n'était pas vraiment un défi difficile mais j'ai passé beaucoup de temps à identifier la vulnérabilité.

**1ère tentative** : xxs dans le formulaire de contact pour obtenir le cookie (perte de temps)

## Contact us

**Name**

**Email**

**Message**

Submit

**2ème tentative** : Path Traversal dans le paramètre page (perte de temps)

qualif.hackerlab.bj:4500/?page=....//....//....//....//etc/passwd

### 3ème tentative : XSS dans le paramètre page

En faisant le path traversal j'ai constaté ce commentaire dans la réponse

```
<!-- Page flag.txt does not exist! -->
```

Ce qui montre clairement que le paramètre page était vulnérable à du Reflected XSS car il renvoie directement les données que nous avons soumis dans la réponse. J'envoie une alerte classique et je constate que notre entrée est filtrée et qu'on ne pouvait pas utiliser directement les scripts habituels. [Évasion du filtre OWASP pour XSS](#) nous donne des commandes à essayer pour échapper ces filtres. Après plusieurs essais

GET /?page=--><script>alert('XSS')</script> HTTP/1.1 voilà la commande que j'ai envoyée pour obtenir le flag HLB2024{XSS\_INJ3CT10N\_1n\_C0MM3N7\_8898})69}