💻 Writeup – SwampTech Solutions – SwampCTF 2025

Morad Halmi – Ecole2600 – Web Exploitation

🎯 Objectif

Le challenge consistait à exploiter un portail d'administration web et à obtenir le flag final. Le contexte était basé sur une interface administrateur d'une entreprise fictive nommée **SwampTech Solutions**, avec des éléments PHP classiques.

🔍 Étape 1 – Inspection de la page

Dès le chargement de la page `http://chals.swampctf.com:49367/`, j'ai inspecté le code source de la page. On y trouvait une **commentaire HTML** intéressant :

```
<!-- TEST USER CREDENTIALS -->
<!-- guest:iambutalowlyguest -->
```

Grâce à cela, je me suis connecté avec les identifiants `guest:iambutalowlyguest`, ce qui m'a permis d'accéder au dashboard `/dashboard.php`.

🧪 Étape 2 – Bypass Auth et accès admin

En observant les cookies avec **Burp Suite**, j'ai remarqué qu'un champ `user` contenait une valeur en **MD5**. En le bruteforçant localement avec `echo -n "admin" | md5sum`, j'obtiens :

```
21232f297a57a5a743894a0e4a801fc3
```

Je remplace donc la valeur du cookie `user` par ce hash. Une fois le cookie modifié, j'ai accès à la page `adminpage.php` avec le message "Welcome, Admin!".

🧩 Étape 3 – Interaction avec la fonctionnalité de vérification de fichier

Sur le panneau admin, il y avait une section "Check if a File Exists". J’ai testé avec différents noms, dont `flag.txt`.

Résultat : le site m’indique que le fichier **flag.txt existe**.

🚀 Étape 4 – Récupération du flag

J’ai ensuite fait une requête GET directe vers `/flag.txt` tout en gardant le cookie `user` à la valeur MD5 de "admin".

La réponse du serveur m’a retourné le flag en clair :

```
swampCTF{WeIrD_F0RmATs_<r>_FuN}
```

✅ Conclusion

Ce challenge était un bon mélange entre :

- Inspection manuelle dans le HTML (commentaire caché)
- Manipulation de cookies pour simuler un compte admin
- Bypass via **hash MD5 statique**
- Interaction simple avec une fonction `file_exists()` côté serveur

🏁 Flag final

```
swampCTF{WeIrD_F0RmATs_<r>_FuN}
```
