# p6ac4et

**Description du challenge :**

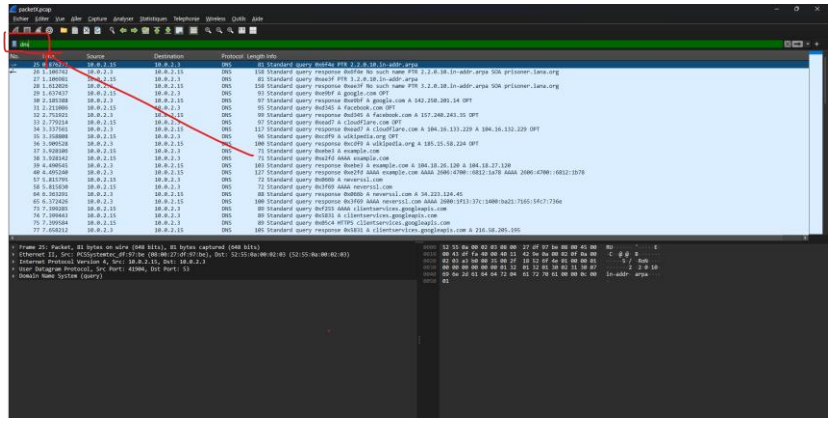
Le PCAP fourni contient un mélange de trafic réseau légitime : TCP, HTTP, ARP, DNS de sites connus, et quelques requêtes DNS supplémentaires vers le domaine.cscc.net.
Parmi ce “bruit”, certaines requêtes DNS contiennent le flag fragmenté et encodé en Base64.

Le joueur doit collecter uniquement les requêtes DNS vers .cscc.net, assembler les fragments dans l’ordre chronologique, puis effectuer un Base64 décode pour obtenir le flag.

**Étapes de résolution**

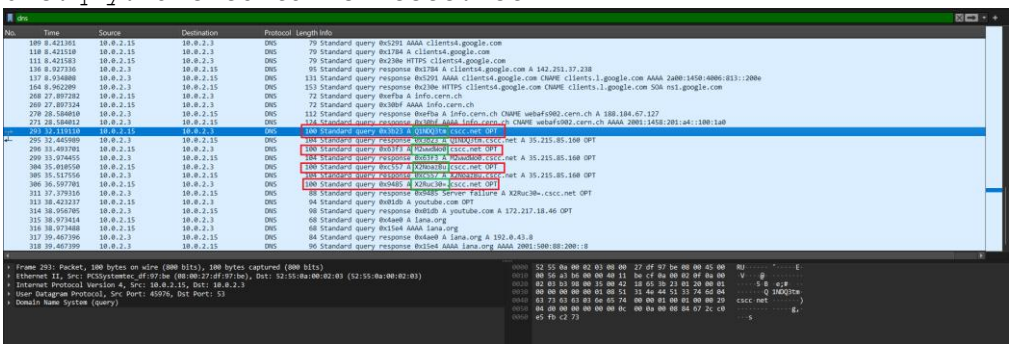
Ouvrir le PCAP dans Wireshark
Filtrer uniquement le trafic DNS :

dns



Identifier les requêtes pertinentes
Filtrer les DNS contenant .cscc.net :

dns.qry.name contains "cscc.net"



Exemples extraits du PCAP :

Q1NDQ3tm.cscc.net
M2wwdWo0.cscc.net
X2NoazBu.cscc.net

X2Ruc30=.cscc.net

**Collecter les fragments**

Respecter l’ordre chronologique (Time)

**Les concaténer dans le même ordre :**

Q1NDQ3tmM2wwdWo0X2NoazBuX2Ruc30=

**Décoder le Base64 Dans un terminal :**

echo "Q1NDQ3tmM2wwdWo0X2NoazBuX2Ruc30=" | base64 -d

**Résultat**

CSCC{f3l0uj4_chk0n_dns}

**Astuces / pièges à éviter**

Les autres requêtes DNS, HTTP, TCP ou ARP sont du bruit : elles ne contiennent pas de flag.

Certains fragments Base64 “faux” peuvent apparaître dans d’autres domaines (.cscc.com, .fake.net) ou dans le trafic HTTP/ARP.

Ne jamais décoder avant d’avoir assemblé tous les fragments.

L’ordre des fragments est critique ; la synchronisation temporelle est la clé.

✅ Flag
**CSCC{f3l0uj4_chk0n_dns}**

Author : **felouja**