Hamster : Outils de BackTrack

Réalisé Par :

M. Sofiane BERABEZ

Master II RIP Décembre 2011

1. Sidejacking avec Hamster et Ferret :

Sidejacking est une méthode passive de reniflement (sniffing) des cookies, puis de les rejouer contre les sites Web pour cloner une session (Gmail, Facebook ou autre) d'une victime. On utilise le terme «sidejacking» pour distinguer cette technique de l'attaque de ***type man-in-the-middle***.

L'avantage de cette méthode est que la victime ne sera pas en mesure de constater si leurs cookies ont été volés.

Pour l'utilisation de hamster avec succès il faut d'abord respecter certaines conditions :

- La première c'est que la victime utilise une connexion ouverte, (exemple : connexion sans fil dans un café), on peut donc écouter passivement les cookies dans ce réseau.
- La seconde est que les cookies utilisés pour identifier la session de la victime ne sont pas crypté par le serveur web.

2. Outils Utilisés :

Comment Sidejacking avec Hamster et Ferret : dans ce qui suit on va faire un tutoriel sur l'utilisation d'un outil dans ***BackTrack 4*** qui est le ***Hamster*** pour accéder à un compte d'une victime qui est sur le même réseau sans connaitre son nom d'utilisateur et mot de passe par un simple vol de cookies en ouvrant les deux paramètres du **Hamster** qui sont ***Ferret & hamster*** tout en les localisant.

- **Hamster** agit donc comme un serveur proxy qui remplace vos cookies avec les cookies de la session de la victime c-à-d réécrire les cookies pour le compte de l'attaquant, hamster manipule toutes les données qui ont été récupérées par **Ferret.**

- **Ferret** (sniffer) analyse les paquets HTTP. Cet outil utilisé pour récupérer des cookies d'une session, il s'exécute au background d'un processus pour la capture des cookies de session qui transitent via un réseau.

Backtrack : Dans cette démonstration nous utilisons Backtrack 4 qui est une distribution linux tournant sur un Live CD qui permet de faire des tests de sécurité sur son réseau.

3. Lancer Hamster:

On ouvre le Shell, D'abord on localise Hamster dans **pentest/sniffers/hamster/**

On lance la commande **./hamster** comme le montre la figure suivante et on remarque bien que navigateur doit être configurer avec un proxy sous l'adresse **1270.0.1** et le port **1234**

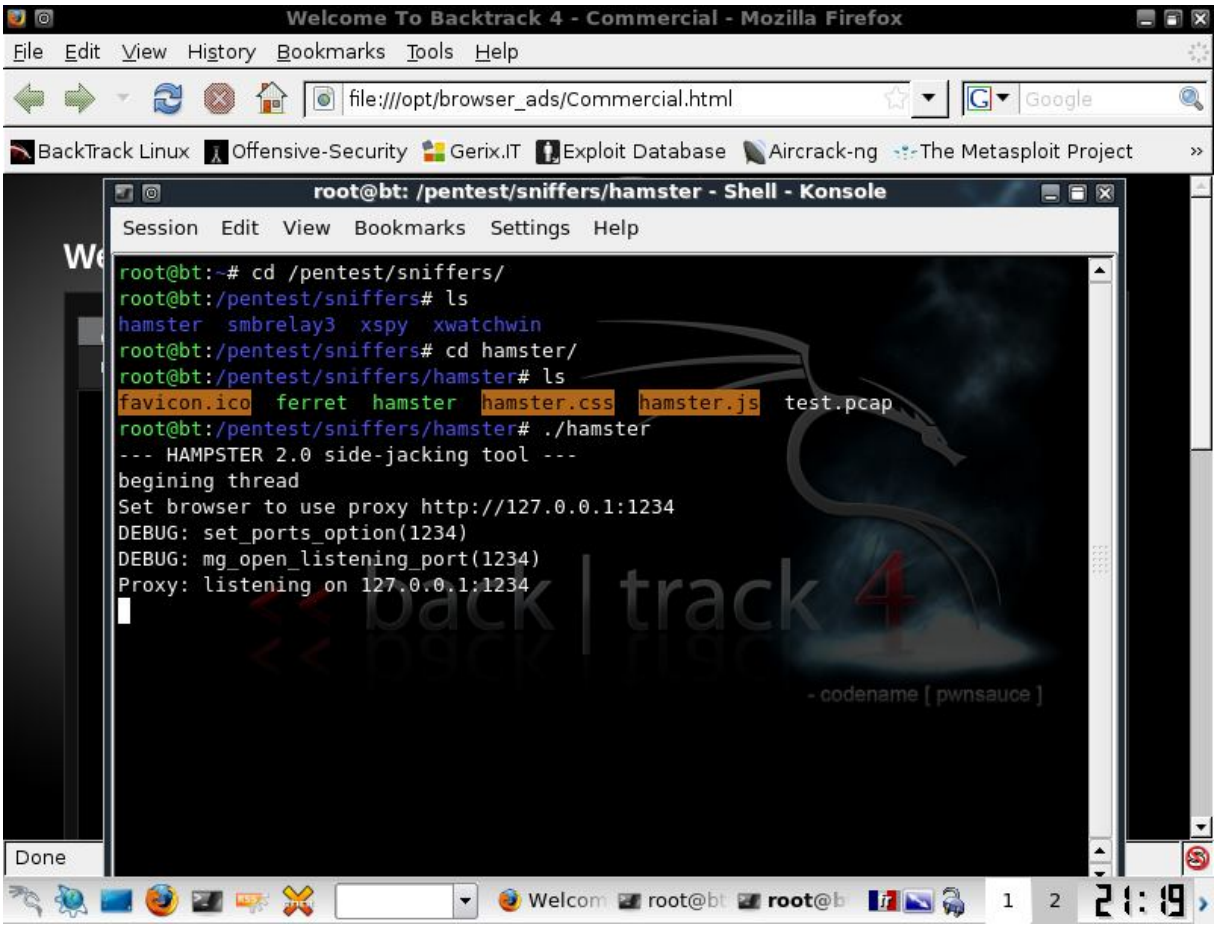


Fig 1: Exécution de Hamster

Configuration de Firefox : Dans Firefox :

Aller dans le menu **Edit -----Preferences**

1. Puis sélectionnez le bouton **Advanced** et ensuite **Network**

- Sélectionner ***Configuration Manuelle du proxy***
- Saisir ce HTTP Proxy: **127.0.0.1** et le Port: **1234**
- Cocher Utiliser ce proxy pour tous les protocoles cliquer sur ok.

On doit avoir un écran comme ci-dessous :

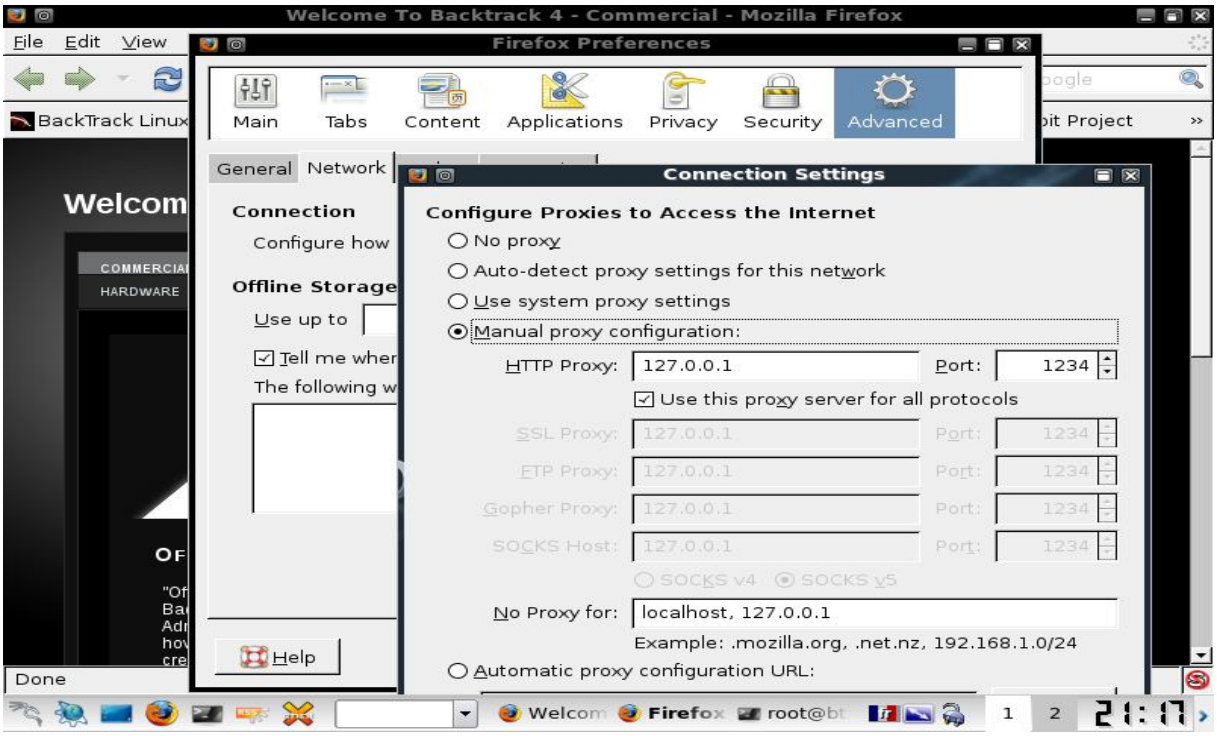
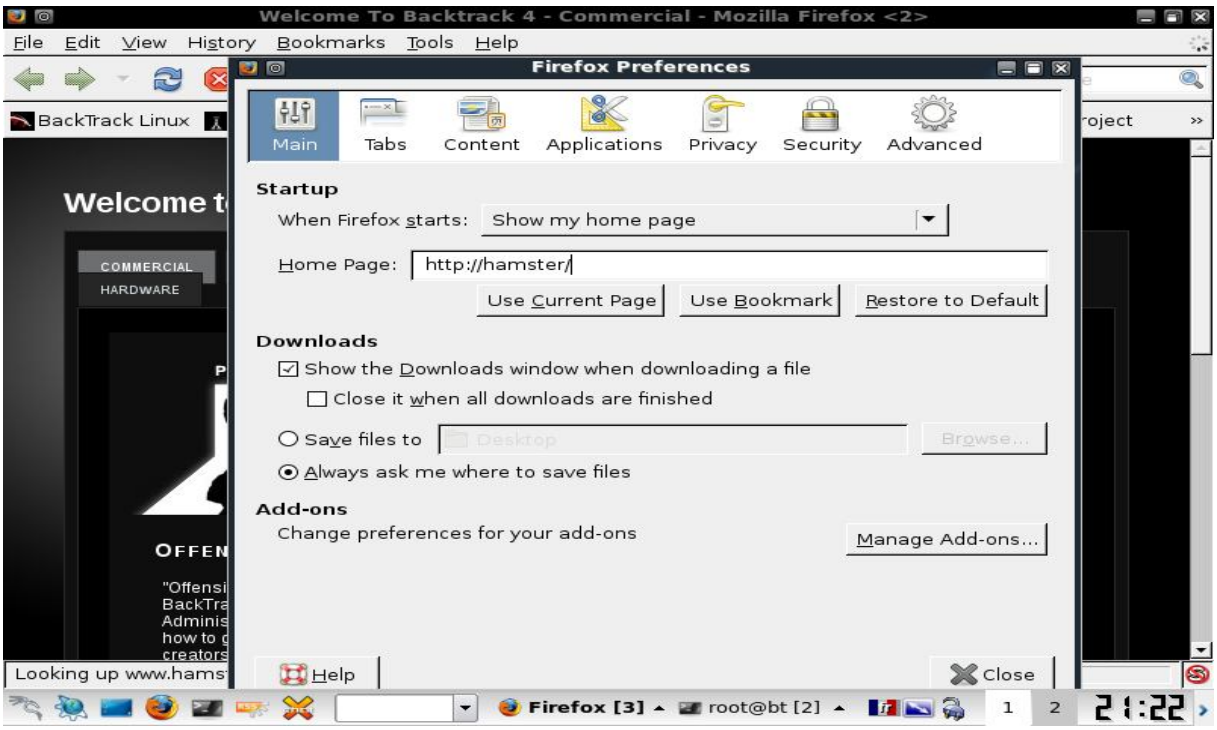


Fig 2 : Configuration de proxy de Mozilla

2. Dans **Main,** dans le champ de la page d'accueil (home page), changer la page d'accueil pour **http://hamster/** afin qu'il s'ouvre automatiquement lorsque vous lancez le navigateur.



Et pour faire fonctionner Hamster on ouvre notre navigateur (qui est sur http://hamster) et l'interface web de Hamster nous apparait :

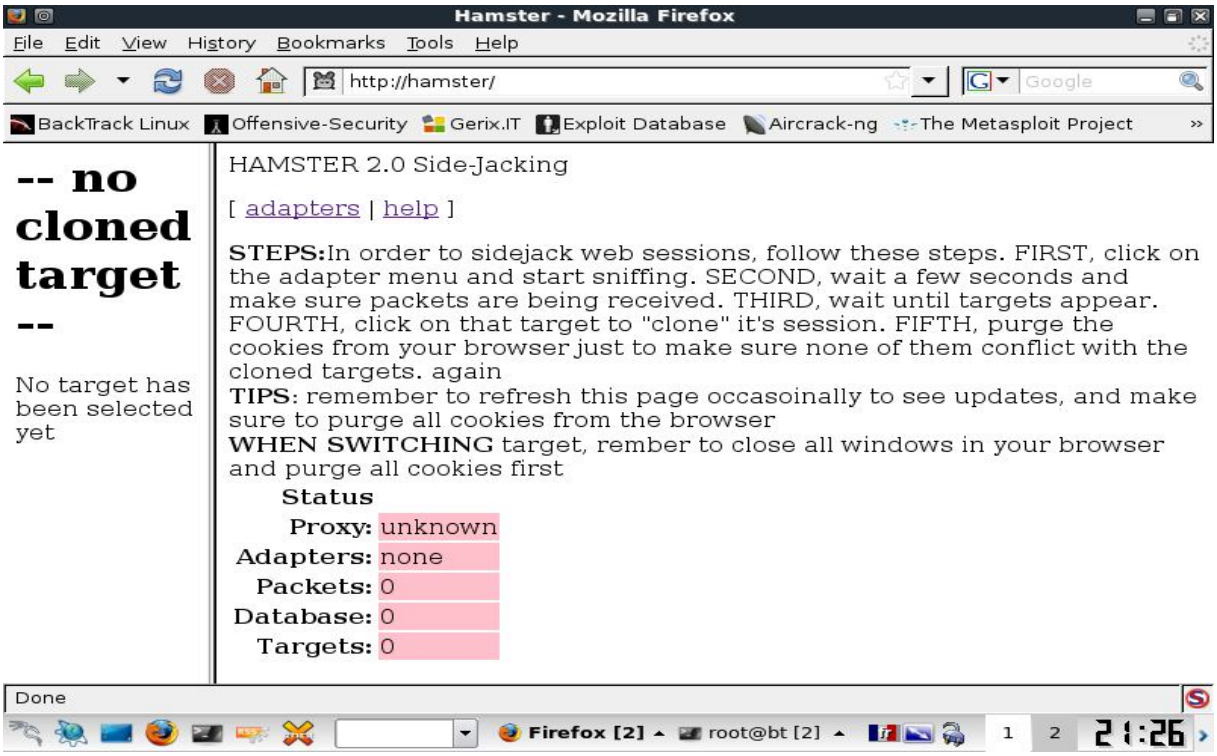


Fig 4 : interface web de Hamster

Puis on clique sur adapters pour choisir notre **adapter (interface)** qui lance lui-même **ferret** en background pour commencer le sniffing et pour cela on doit d'abord connaitre les noms des ***adapters (interfaces)*** qui sont présents en lançant la commande ***ifconfig***

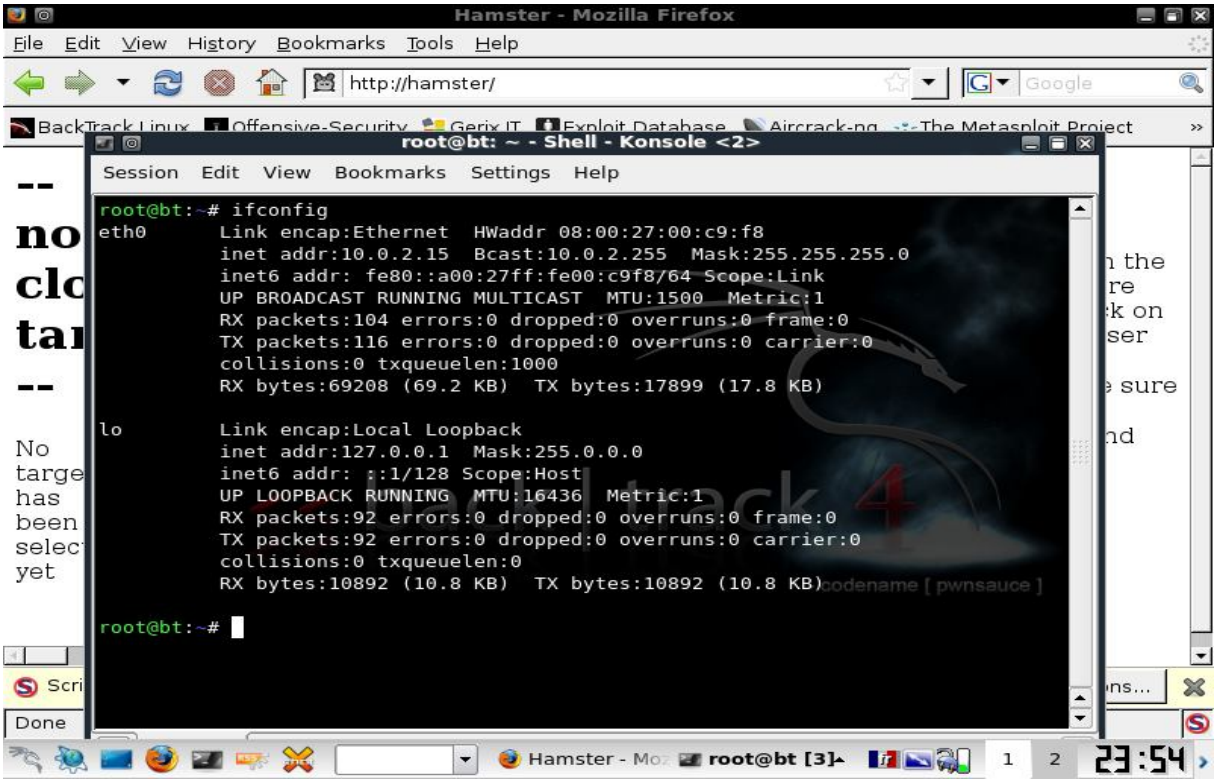


Fig 5 : Adapters (interfaces)

On va choisir notre adapter eth0 et on clique ***sur submit query*** et on aura la figure suivante et en background on aura ferret activé

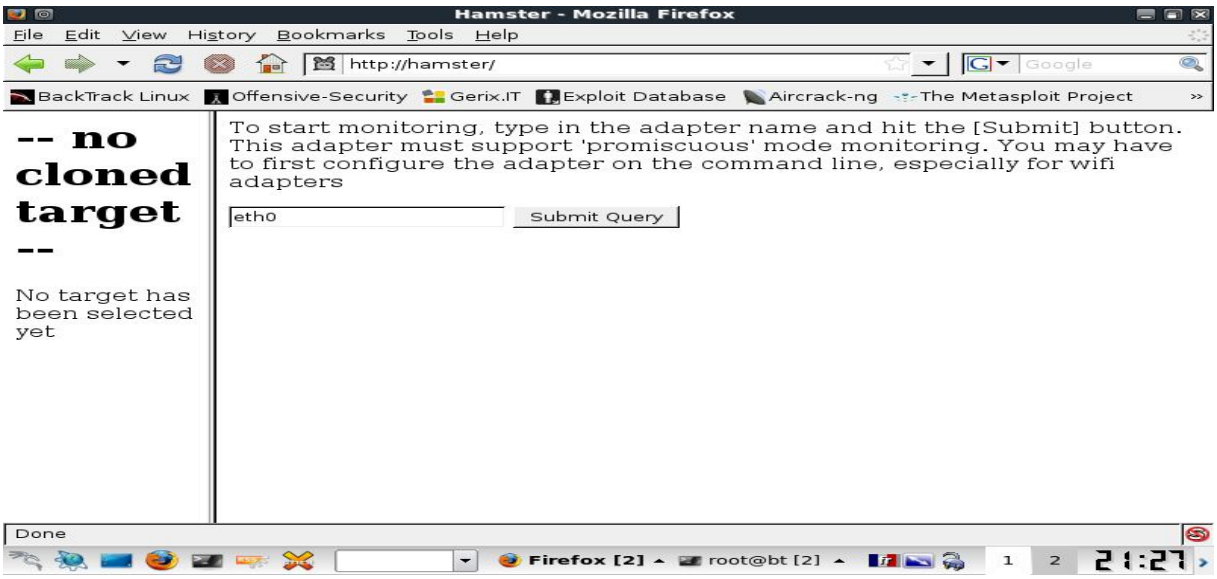


Fig 6 : choix de l'adapters (de l'interface)

Et la figure suivante nous montre l'exécution de ferret en background et début du sniffing sur l'interface **eth0**

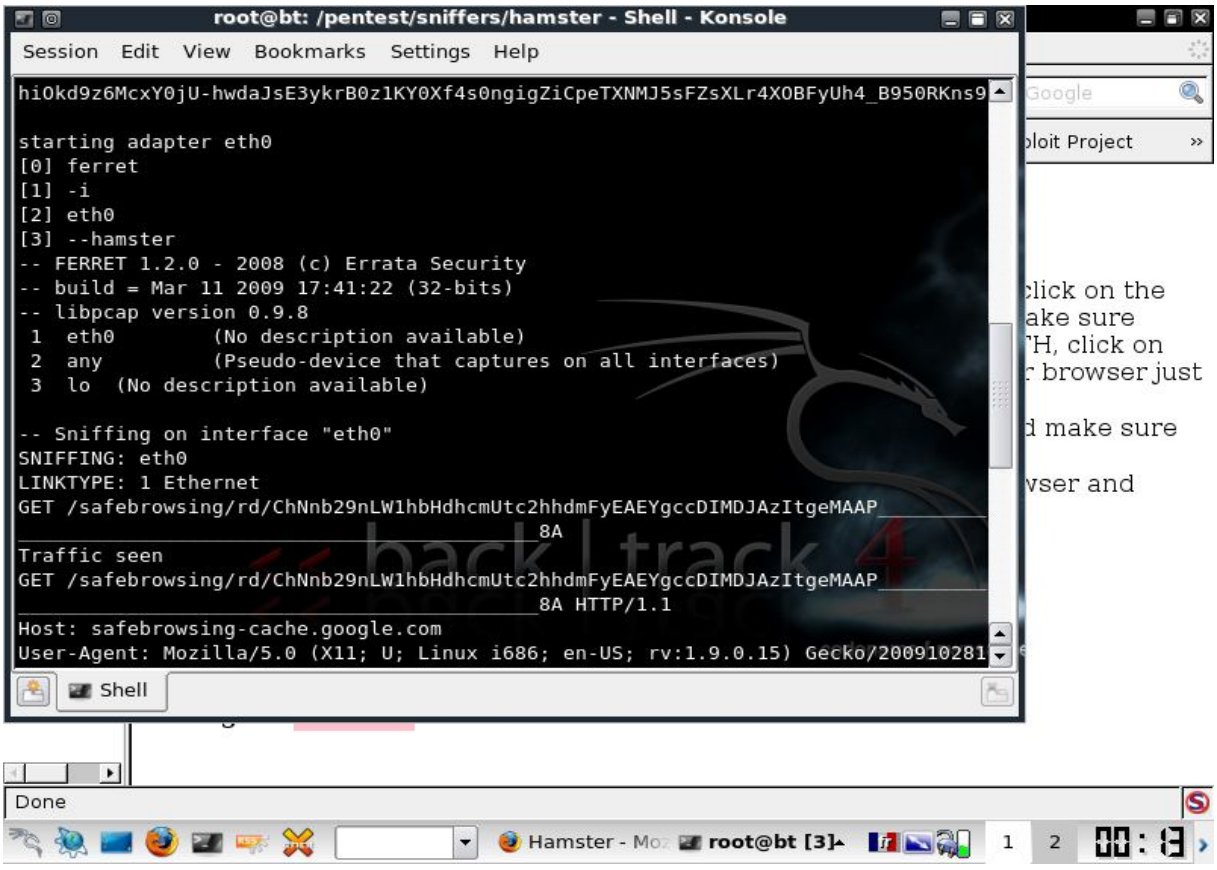


Fig 7 : Exécution en background de **Ferret** et le début de sniffing

Maintenant après avoir lancé Hamster et Ferret et après configuration de notre navigateur on va se connecter sur un compte on s'authentifier et récupérer ces cookies, pour les rejouer après. J'ai choisi de me connecter au site de POF.com pour faire mes tests, je lance http://www.pof.com/ et je m'authentifie avec un login ***Sofiane_Hmaster*** et un mot de passe, après avoir ouvert mon compte sur POF je réactualise http://hamster/

Fig 8 : Ouverture d'une session Compte POF

Sur la figure suivante qui est l'interface web de l'outil Hamster on remarque que l'interface sur laquelle on est en train d'écouter est affichée ainsi que le nombre de paquet et le nombre qui transitent sur notre réseau.

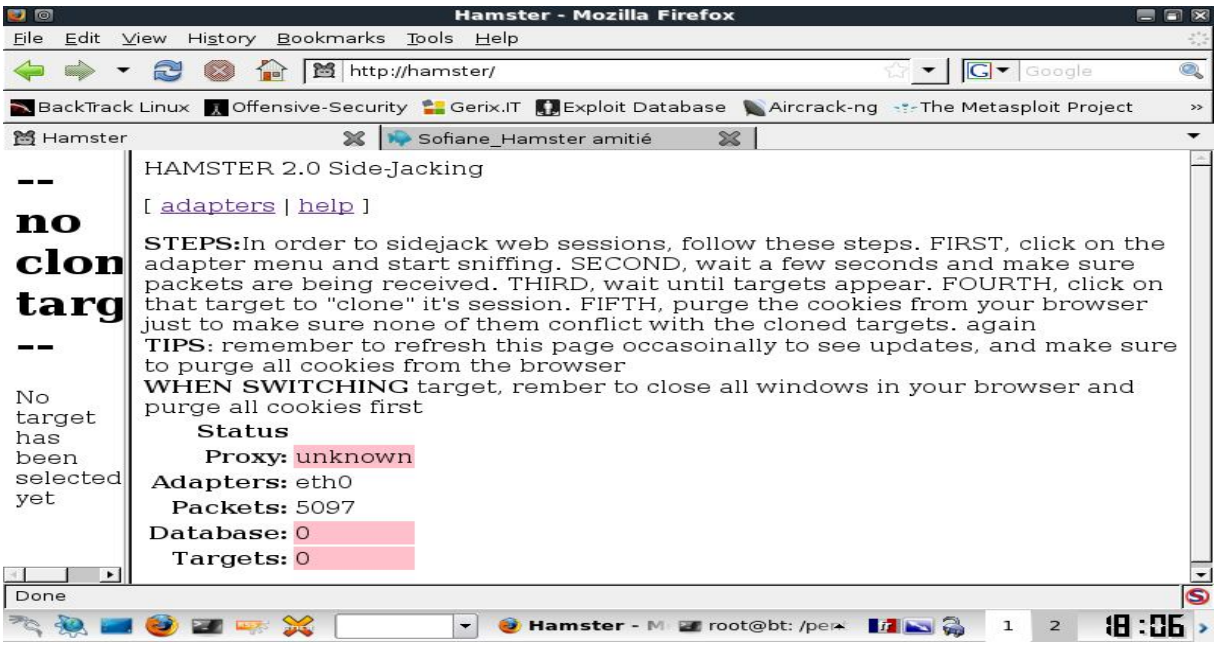


Fig 9 : Interface web du Hamster avec l'interface écouté et le nombre de paquet transitant

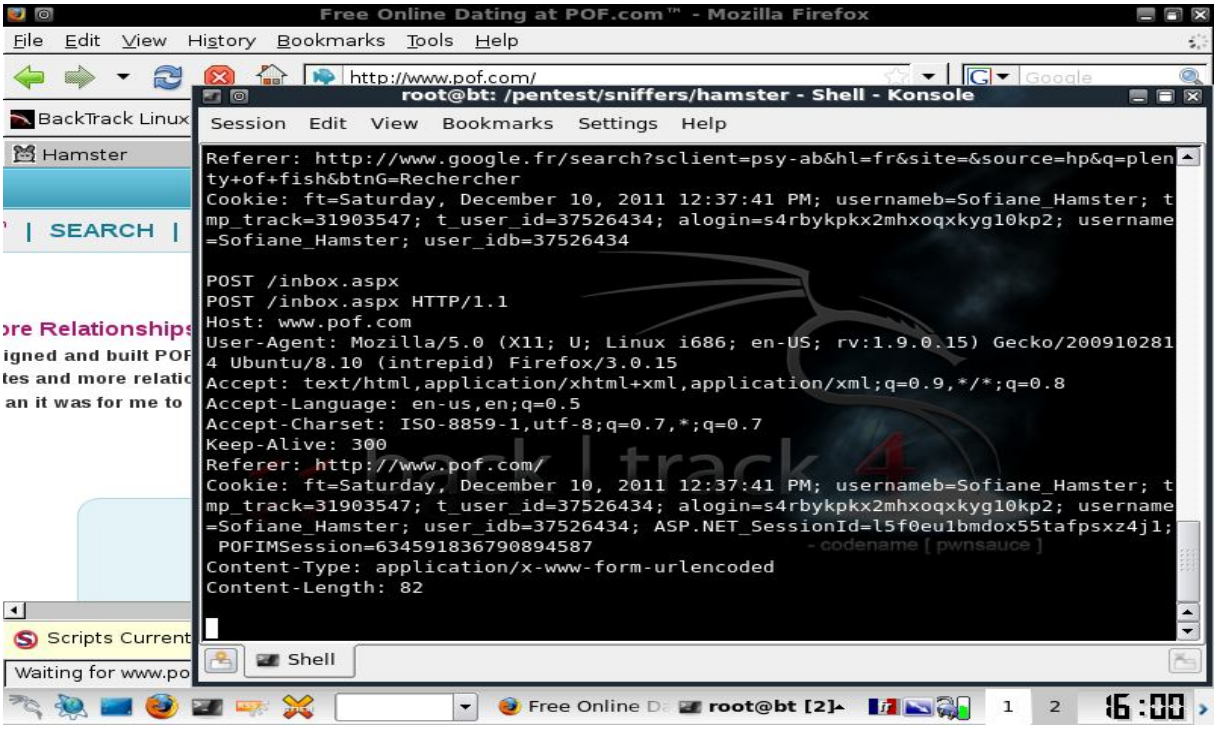


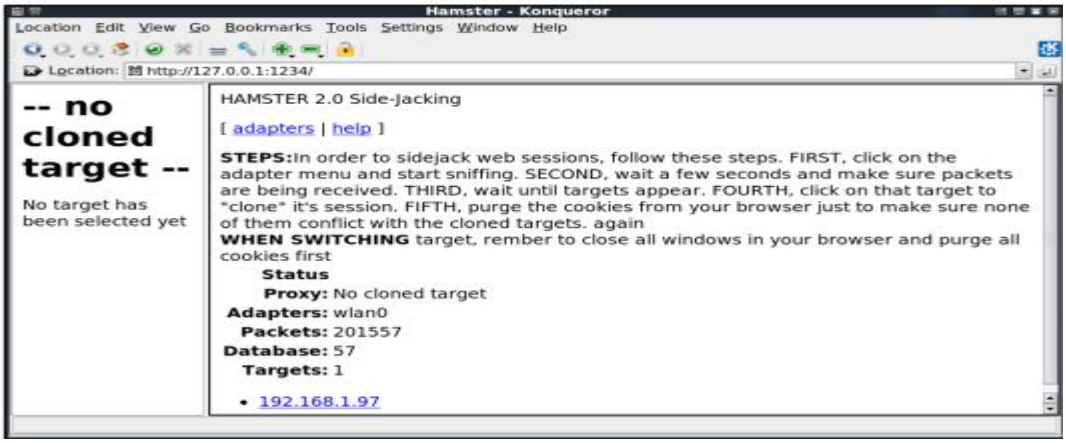
Fig 10: les paquets sniffés et les cookies récupérés

La figure ci-dessus montre les paquets sniffés et tous les cookies récupérés par **Ferret** et qu'on pourra utiliser pour accéder mon compte POF sans s'authentifier à nouveau. Et c'est le but de l'outil **Hamster.**

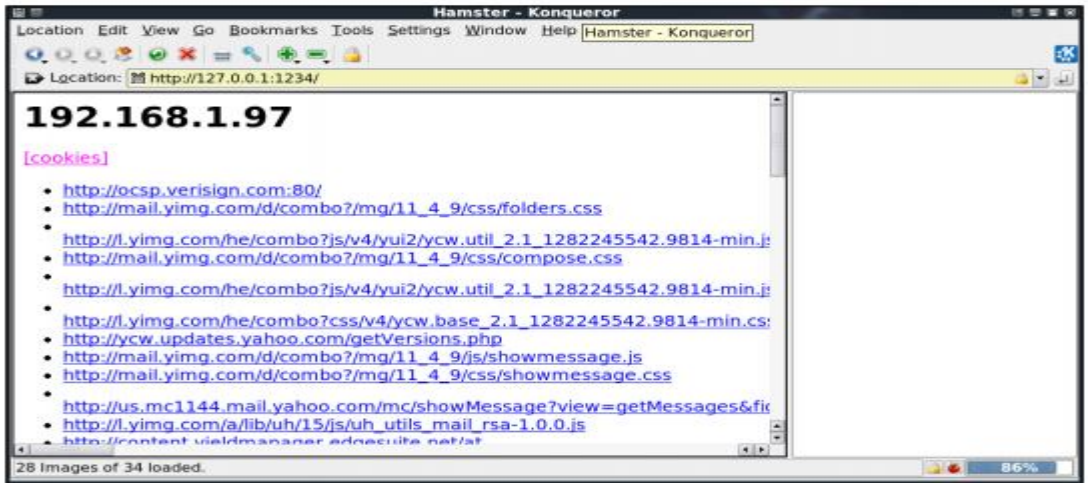
Comment peut-on rejouer ces cookies via l'interface web et accéder à un compte ?

Si notre pc est équipé d'une carte réseaux qui permet le mode promescous on pourra rejouer les cookies directement via l'interface web de Hamster sans avoir à les remplacer manuellement de la manière suivante :

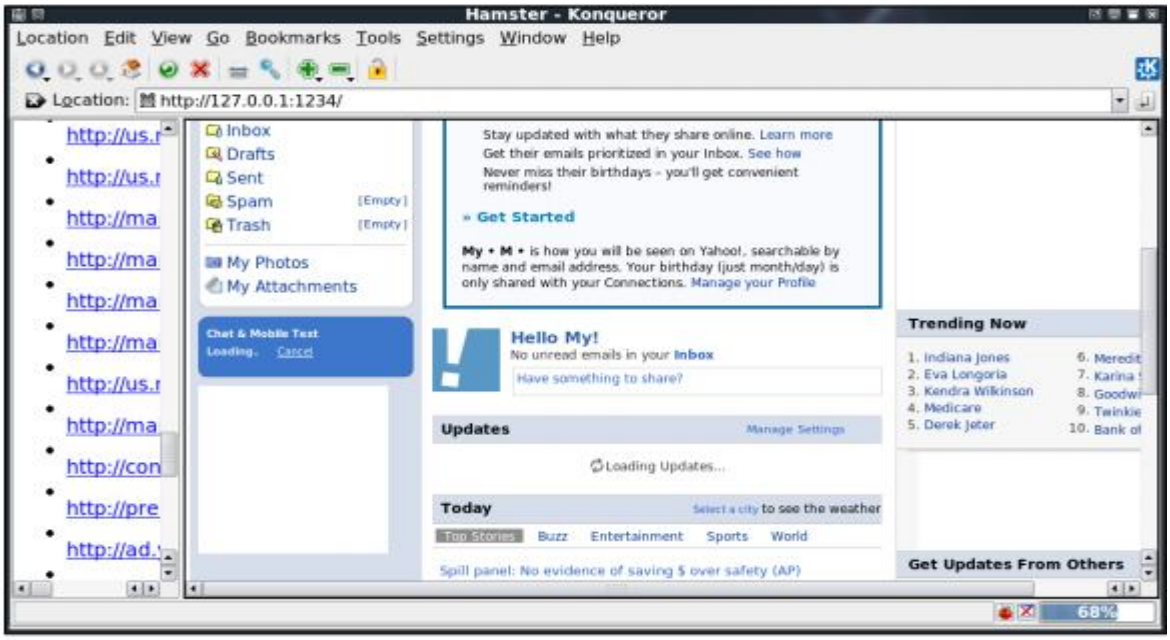
Après avoir s'authentifier à un compte on réactualise http://hamster/ et cette page doit nous apparaitre (Image prise dans le livre : BackTrack 4: Assuring Security by Penetration Testing des auteurs Shakeel Ali Tedi Heriyanto)



Après une petite attente, on verra l'apparition d'adresses IP. On clique sur l'adresse IP de la victime afin de cloner ses sessions. Dans la fenêtre de gauche, on voit les sites que la victime est en train de visiter.



Et pour accéder au compte Yahoo de la victime on n'a qu' cliquer sur le line approprié et on aura accès à sa messagerie comme suit :



Conclusion : Protégeons-nous

La leçon à tirer de cette étude est que les utilisateurs ne peuvent vraiment pas se permettre d'être laxiste sur le cryptage des données envoyées sur les réseaux ouverts sans fil et aussi de se déconnecté a chaque fin d'utilisation d'une session sur le net.