Préparation a la certification OSCP

Gauvain Roussel-Tarbouriech

Plan/déroulé du cours

Pourquoi l'OSCP

- Certification de relativement bon niveau
- La certification la plus connue d'Offensive Security (les créateurs de Kali Linux)
- Pas une cert "bullshit" a la CISSP, CEH
- Introduit pas mal de concepts vu et revu dans le monde réel
- Problème

Exemples

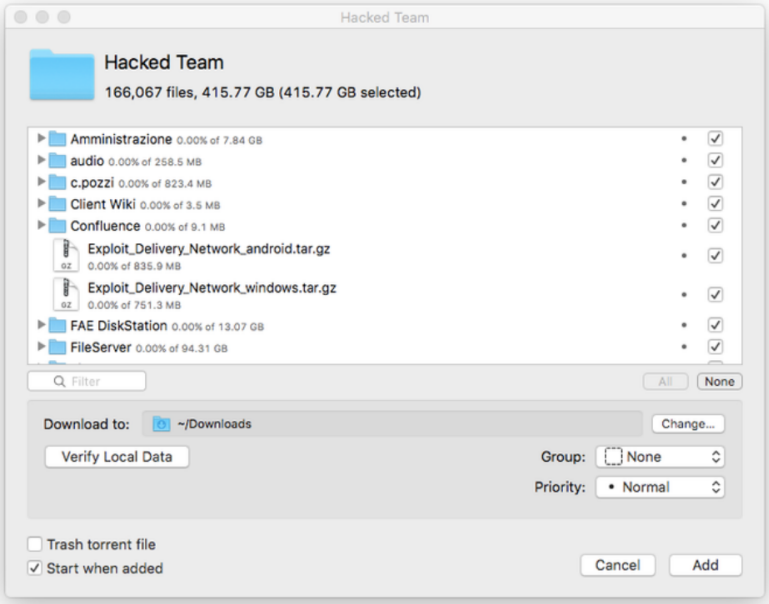
Hacking Team piraté – 400 GB de données dans la nature

@KORBEN — 6 JUILLET 2015

Mise au même rang que Blue Coat, Amesys, Gamma International et Trovicor, la société italienne **Hacking Team** a été classé comme ennemi d'Internet.

Ces derniers vendent à leurs clients une solution de surveillance à distance baptisée Remote Control Systems (RCS) incluant les outils DaVinci et Galileo qui permettent, en autres, de collecter des infos sensibles (emails, historiques d'appels, carnet d'adresses..etc.), les touches tapées au clavier (keylog), de prendre des captures-écran, récupérer l'historique des recherches web, suivre le GPS du téléphone, enregistrer l'audio des appels téléphoniques, activer la caméra du téléphone ou de l'ordinateur, activer le micro de l'ordinateur ou du téléphone pour enregistrer les bruits ambiants ou les conversations...etc etc.

Bref, un beau joujou pour les gouvernements. Seulement, voilà... La société Hacking Team vient de se faire poutrer, se faisant dérober un peu moins de 400 Gb de données sensibles comme des contrats, des factures et de la doc.



Exemples

[BREAKING] Nintendo's Old Files Have Been Stolen; That's Why There's Super Mario 64, Says Report

Tech Times | 05-05

A Nintendo massive breach had happened over the weekend. Over 2TB of old files of the said company have been leaked, according to a report. Interestingly, a day after the said breach, a fan-made port of Super Mario 64 running at 4K resolution on PC was released-- rising the speculations of whether the info came from the said breach.

Exemples

- Bon, vous avez compris l'idée

Objectifs

- Savoir analyser l'infrastructure d'un réseau, trouver les points faibles.
- Savoir faire du pentest web afin d'avoir une porte d'entrée plausible sur des serveurs.
- Savoir faire de l'énumération afin de pouvoir faire une Elévation de privilèges.
- Comprendre les configurations des AD et trouver les problèmes potentiels.
- Savoir exfiltrer des données sans laisser trop de traces.

- En sécurité on a 3 "principaux" domaines: l'exploit, le pentest et la défense.
- L'exploit c'est trouver des bugs dans des programmes, eg des CVEs.
- Le pentest c'est trouver des erreurs d'organisation dans des infras.

Modalités de notation

- Un serveur a infiltrer avec un rapport a rendre pendant les dernières heures de cours.

Matériel nécessaire

- Préférable: Une VM Kali Linux afin d'éviter les temps de compilations des inévitables gentoo-istes.
- Agréable: un OS avec un noyau Linux [2]
- En vrac: `nmap`, `owasp-zap`/`Burp Suite`, `firefox`, `gobuster`/`dirsearch`, `BlueHound`, `sqlmap`,`hashcat` et plus encore.

[2] Non, WSL1, ça compte pas trop :P. WSL2 ça peut avoir ses bugs.

Où trouver les ressources ?

- Le discord du cours, que je vais de ce pas vous donner.
- Le lien du cours pour toutes les slides/ressources qu'on a vu en cours: https://code.govanify.com/govanify/esgi-oscp.
- Les listes awesome ctf/security sur GitHub et lire beaucoup de writeups :D

Plan du cours

- Introduction au Pentest web
- Introduction a l'exploit binaire/reverse et au reverse engineering[3]
- Introduction a l'Elevation de privilèges et a la configurations de serveurs + AD
- Beaucoup de pratique :)

[3] https://code.govanify.com/govanify/esgi-re/

Le Pentest web

Le Pentest web

Je ne sais connais malheureusement pas votre niveau ou vos connaissances en web donc on va devoir repartir depuis le début

Architecture d'une page web

- De l'HTML qui contient le "coeur" du document
- Du CSS qui mets en page l'HTML
- L'HTTP, le protocole de communication avec des serveurs web
- JavaScript qui permets de modifier l'HTML et le CSS et de communiquer avec des serveurs

Verbes HTTP

- GET, obtenir une ressource
- HEAD, meme chose que GET, mais sans la ressource, probablement inutile pour nous
- POST, envoyer une ressource
- PUT, remplace une ressource
- DELETE, supprimer une ressource
- CONNECT, pour se connecter a un tunnel, probablement inutile pour nous
- OPTIONS, pour savoir quels verbes sont supportés
- TRACE, renvois la ressource envoyée
- PATCH, pour modifier une ressource

Bon, tout ça c'est un peu abstrait, voici ce que ça donne dans la réalité:

```
GET / HTTP/2
```

L'HTML

Les ressources du protocole HTTP sont souvent au format HTML. Je ne vais pas expliquer l'HTML ici vraiment, donc pour ceux qui ne savent pas ce que c'est clic droit sur firefox, inspecter l'élément ou demandez moi de l'aide après les slides! Sinon en gros c'est un format d'organisation textuelle par balise genre `<h1>test</h1>`

Le CSS & JavaScript

En gros meme chose que l'HTML mais pour la mise en page de l'html, eg: `.c{color:#747369}` mets le tag de classe c avec la couleur RGB en hex.

Le JavaScript quand a lui est un langage de programmation de typage faible qui peux faire des appels réseaux et modifier le CSS ou l'HTML.

Les cookies

Il s'agit de données personelles, généralement stockant avec quel compte vous etes connecté etc. C'est sauvegardé du coté client et est envoyé lors de requetes HTTP.

Problème de sécurité commun de structure

En connaissant juste les bases on peux déjà introduire 2 problèmes de sécurité majeurs qui arrivent de temps a autre:

Les serveurs gérant mal certains verbes HTTP et vous donnant un accès non voulu a des ressources, eg si un serveur ne vous autorise pas a obtenir des ressources via GET mais vous les renvois avec un POST.

Les commentaires! Trop souvent dans du code de production vous verrez des commentaires vers une interface admin, certaine fois avec des données sensibles.

Il s'agit plus d'énumération, mais le fichier robots.txt a la racine d'un site web peux vous apprendre pas mal de choses

XSS

L'HTML est un langage de tags. On peux rajouter du texte utilisateur dans de l'HTML, eg pour dire votre nom d'utilisateur. Que se passe-t-il si ce texte continent de l'HTML?

Si l'HTML n'est pas filtré, vous pouvez remplacer des parties de la page web par du contenu que vous controlez, ce qui contient du javascript. Le javascript peux faire des requetes réseaux et a accès a vos données personelles. F.

Il y a 3 types de XSS:

- Reflected, ou l'utilisateur envois une requete avec de l'html et le serveur lui rajoute
- Stored, ou le serveur a stocké l'HTML dans une base de donnée et le renvois a l'utilisateur
- DOM, ou le javascript qui modifie votre page est vulnérable