Retour d'expérience sur un "Hack Challenge"

Daniel DUPARD

Enseignant EPITECH

daniel.dupard@nxbp.fr

Présentation du Challenge de Hacking

- Objectifs Initiaux
- Organisation
- Résultat
- Enseignements
- Protection contre les "buffer overflow" du serveur Web
- Conclusion

Objectifs

- Prouver qu'une machine sous Windows 2000 correctement configurée est difficilement attaquable
- Permettre à mes élèves de réaliser un cas concret d'intrusion
- Faire un recensement des attaques et étudier les meilleures attaques

Organisation (1/2)

- Période initiale 3 mois
- Prime initiale 100 euros
  - Pas d'augmentation de la prime avec le temps (prévu initialement, cela a changé en cours de challenge)
- 3 niveaux
  - Compléter une phrase qui commençait pas tous les matins située sur le répertoire du serveur web,
  - Compléter une phrase qui commençait par tous les midis située dans un répertoire autorisé uniquement pour l ’administrateur,
  - Compléter une phrase qui commençait par tous les soirs accessible par un appel système ajouté dans le noyau.

Organisation (2/2)

- 1 Firewall bloquant tout sauf le port 80
  - port 80 autorisé en entrée
  - tous les ports autorisés en sortie
- 1 PC Windows 2000 Pro à jour des derniers patchs + derniers hotfix
  - SP3 + MS 03-005
- 1 serveur Web Abyss 1.1.2
  - pages HTML simples
- 1 système de journalisation externe

Résultats (1/4)

- 3562 tentatives
  - Adresses IP ayant fait des requêtes d’intrusion
- 2 attaquants ont réussi le premier niveau
- Aucun attaquant n'a réussi le niveau 2 ou 3

Résultats (2/4)

- Les 15 premiers jours (prime 100 euros)
  - 2142 adresses IP différentes
  - 2013 tentatives par logiciels disponibles sur Internet
    - Nessus, Saint
  - 64 tentatives par failles connues
    - It is possible for a remote attacker to disclose the contents of arbitrary web-readable files by making a specially crafted web request containing encoded dot-dot-slash (../) sequences.
    - By connecting to the remote web management interface at http://abyss_server:9999 an attacker can use a brute-force method to gain access to the server. There is no delay in a wrong attempt and attackers are given an indefinite number of attempts at entering a valid user and password. Unlike the access.log file for port 80, Abyss has no logging for port 9999. This allows an attacker to perform unseen.
  - 65 tentatives non standard
    - URL malformées
    - Outils d'attaque personnalisés
    - Nessus modifié

Résultats (3/4)

- Les 15 suivants (prime 300 euros)
  - 1167 adresses IP différentes
  - 1050 tentatives par logiciels disponibles sur Internet
  - 3 tentatives par failles connues
  - 9 tentatives non standard

  Nouveauté :
  - 5 tentatives basées sur le désassemblage du code du serveur (=> DoS)

Résultats (4/4)

- Les 15 derniers (prime 1000 euros)
  - 253 adresses IP différentes
  - 227 tentatives par logiciels disponibles sur Internet
  - 10 tentatives par failles connues
  - 12 tentatives non standard
  - 4 tentatives basées sur le désassemblage du code du serveur (=> DoS)

Enseignements (1/2)

- Difficultés à analyser les attaques avec les outils utilisés (Snort, Ethereal)
  - Détecter une attaque
  - Isoler les paquets incriminés
  - Rejouer l'attaque
  - Comprendre l'attaque

- Existe-t-il un outil qui simplifierait la tâche ?

Enseignements (2/2)

- Objectif
  - Protéger de manière générique un serveur Web contre le "buffer overflow"
- Méthode
  - Filtrer au niveau de l’appel à la socket du client tout message de plus de n caractères (n = 100)
- Moyen
  - Réaliser un petit programme externe au serveur qui permet de programmer des filtres de requêtes HTTP
  - Principe de URLScan, SecureIIS, etc.

Conclusions

- Intérêt du challenge
  - Trouver des failles non identifiées
  - Faire tester son logiciel
- Les meilleurs attaquants ont-ils fait le challenge ?
- A quel niveau de prime auraient-ils fait le challenge ?