L'exploitation

Ian-Kyle Wagner

Club de Sécurité Informatique, Université Laval

Hiver 2021

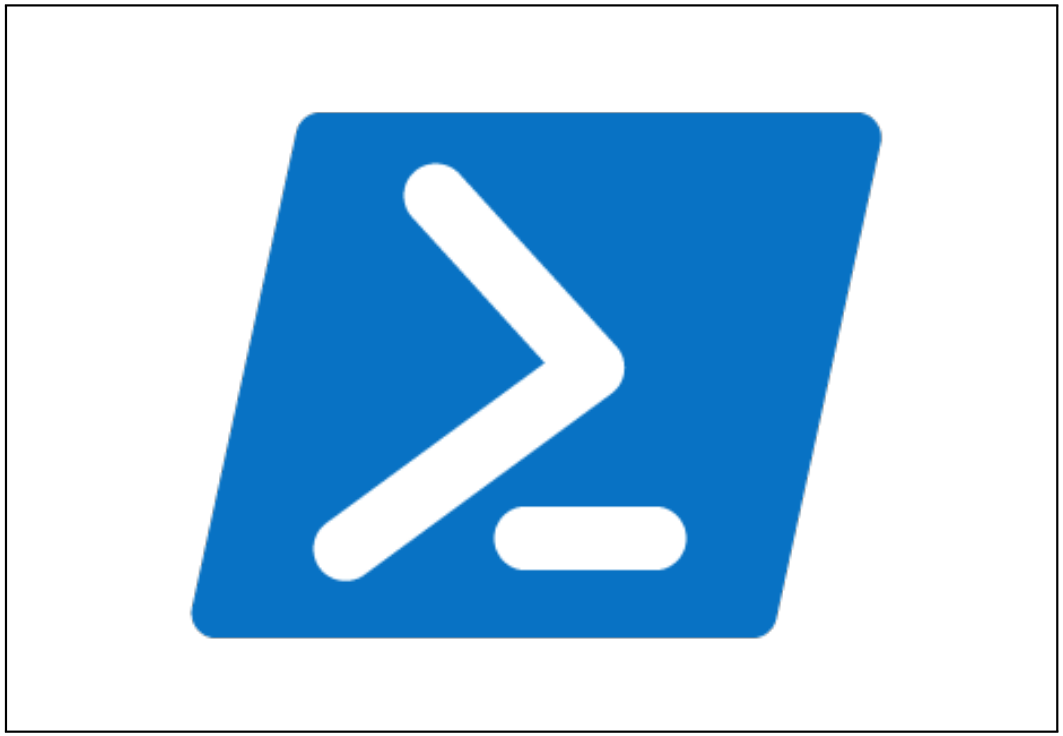
Table des matières

Section 1

Introduction

Introduction à l'exploitation

- utiliser l'espace mémoire d'un programme
- injection de code dans la mémoire débordé dans un programme
- Action non autorisés comme executer ouvrir programmes



Pourquoi exploiter?

- Obtenir l'accès à une machine sans authentification par la courruption de la mémoire logiciel (Buffer Overflow)
- Illustrer les différents les logiciels utilisé incluant les logiciel de protocoles réseau (SMB, inetd, apache, ...)
- Démontrer les vulnérabilités pour audit de sécurité

Section 2

Buffer overflows

Qu'est-ce qu'un Buffer overflow?

- Corruption de la mémoire d'une application
- Ajouter des données supplémentaires qu'un tableau (array) supporte
- Permet d'exploiter le système en ouvrant des processus non authorisés

Comment exploiter une application (processus de base)

- Lire code source ou désassemblé
- Trouver l'adresse commandes vulnérables ex: buf[num] en C
- Injecter à l'adresse de la vulnérabilité pour trouver le nombre de charactères à ajouter
- Insérer les code de contrôle (shellcode) après la séquence de charactères intiaux

Type d'exploits

Les types selon exploitdb:

- Dény de Service (DoS)
- Exploit local (local)
- Exploit à distance (remote)
- Élévation de privilège (Privilege Escalation)

Exploits locaux

- Exploit limité à l'accès direct de la machine seulement.
- Permet d'exécuter n'importe quel autre processus sur la machine exploité
- Utilisé pour démontrer des vulnérabilités sans compromettre le système
- exemple: ouvrir powershell à partir de MS Office en utilisant DDE:
  https://null-byte.wonderhowto.com/how-to/exploit-dde-microsoft-office-defend-against-dde-based-attacks-0180706/

Démo Exploit Local

Exploits à distance

- Exploit permettant à un attaquant d'avoir accès à la machine
- Permet d'exécuter n'importe quel autre processus sur la machine exploité à distance
- Peut être utilisé par un attaquant pour l'exfiltration des données ou l'installation de programmes malicieux
- exemples: MS17-010 (Eternal Blue), CVE-2019-0230

Section 3

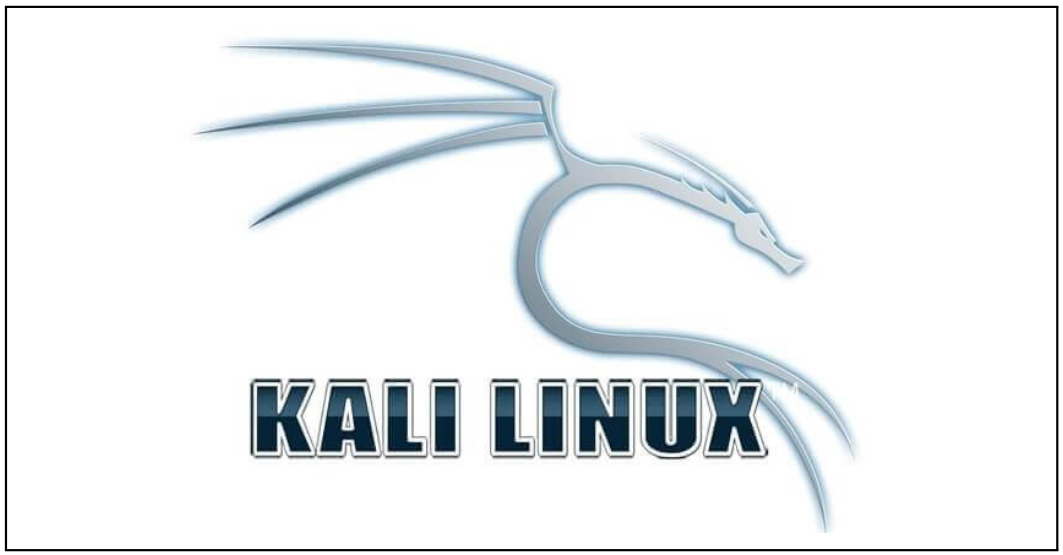
Metasploit

Metasploit

- Framework open source
- permet d'automatiser l'exploitation binaire
- Modulaire et contient plusieurs outils
- Outils: msfconsole, msfvenom, ...
- Sous Kali Linux: service postgresql start
- Pour intialiser après installation: msfdb init

msfconsole

- Terminal interaction entre machines (Meterpreter)
- Permet d'utilisation des modules (exploit)
- module /exploit/multi/handler pour ses propres exploits
- Inclut des outils de postexploitation (Mimikatz, Keylogger, ...)



msfconsole commandes

- set LHOST [ip address]
- set LPORT [port number]
- set RHOST [ip address]
- search [CVE || vulnname || vuln code (ex: MS17-010)]
- set SESSION [num]
- session [-l || -i]
- show [payloads || options]
- use [exploit/OS/software/name]
- set payload [payload directory]
- info
- exploit
- exit
- set target [num]

meterpreter commandes

Les commandes bash standard et les commandes suivantes:

- upload
- download
- edit (modifier fichier avec vi)
- execute -f proc.exe
- shell
- session -l
- del
- hashdump
- getsystem (escalation de privièges)
- use [(auxilary/post) modules]
- clearav
- keyscan_start || keyscan_stop
- run [script]
- idletime
- background

msfvenom

- logiciel de création d'exploits
- encodage possible pour évasion des antivirus
- utilisation différent langages options programmation
- support multiple architectures et platformes cibles (Unix,Windows,...)

Démo Metasploit

Exercices exploitation

Sur le réseau du club de hacking à l'aide de metasploit attaquer les machine virtuelles aux adresse IP suivantes: 10.10.1.6 et 10.10.1.9

Section 4

Antivirus evasion

Problèmes avec l'exploitation

- Peut être facilement détectable
- Surveillance réseau
- Signatures IDS/IPS et antivirus
- Dépend de fichiers pouvant être accessibles

Solutions aux problèmes

- Obfuscation du traffic réseau de l'exploit
- Utilisation de protocoles connus (HTTP(S), FTP, ...)
- Utilisation de ports réseaux communs (80, 443, 22, 21,...)
- Encodage de l'exploit
- Utilisation d'un différent langage de programmation (go, C, python, ...)

Évasion des antivirus

Technique utilisant l'encodage d'un exploit ainsi que des langages de programmes pour le rendre difficile à détecter par des solutions de sécurité comme des antivirus.

Programmes pour automatiser av evasion

Exemple d'outils:

- Veil Framework
- Shellter
- Hyperion
- empire (defunct)
- phantom-evasion (defunct)
- antivirus evasion tool

Veil Framework

- Logiciel libre pour créer des binaires indétectables
- Plusieurs options pour les langages de programmation et encodage
- Seulement capable de créer des exploits de type PE
- site internet: https://github.com/Veil-Framework/Veil

Installation de Veil Framework

Sous Kali Linux:
(sudo) apt update && (sudo) apt -y install veil
/usr/share/veil/config/setup.sh −−force −−silent

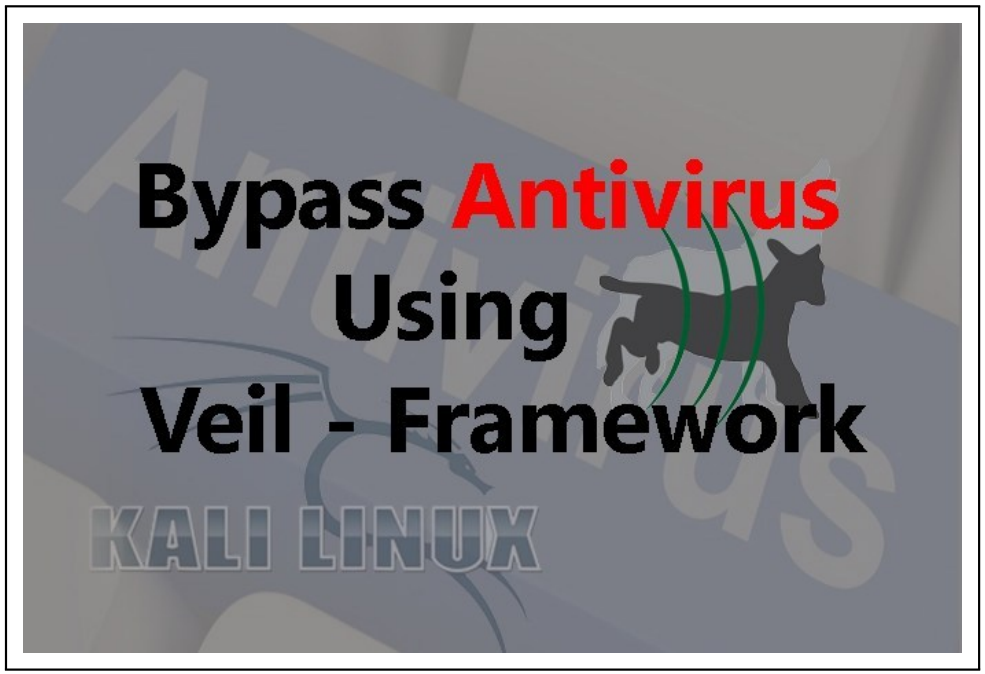
Shellter

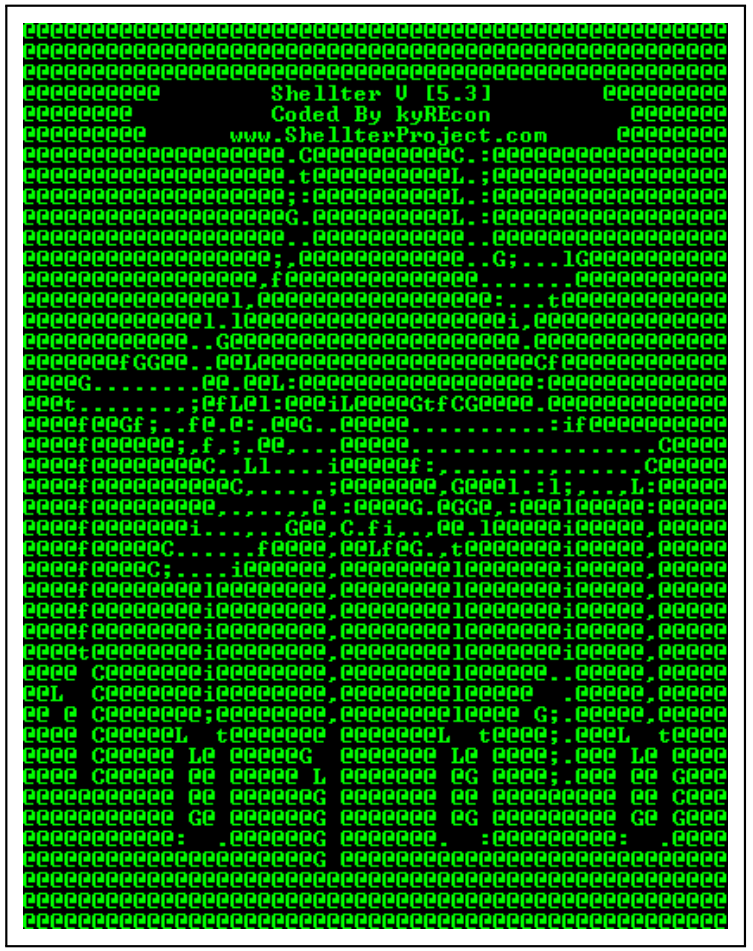
- Logiciel libre pour ajouter des exploits à des installateurs
- Ne fonctionne pas avec certains types d'installateurs (ex: VLC)
- Options d'évasion pour éviter la détection
- Seulement de modifier des binaires de type PE
- site internet: https://www.shellterproject.com/

AntiVirus Evasion Tool

- framework pour l'évasion des antivirus
- https://github.com/govolution/avet
- git clone https://github.com/govolution/avet.git && cd avet
- sudo bash setup.sh
- python3 avet.py

Démo Evasion AV





Exercices d'évasion

1. Créer un exploit à l'aide d'un des outils et exploiter une machine virtuelle Windows avec un AV.
2. Télécharger l'installateur de WinRAR, le modifier avec Shellter et exploiter la même machine virtuelle.