Universitatea Tehnică a Moldovei

Facultatea Calculatoare Informatică şi Microelectronică

Departamentul Ingineria Software și Automatică

**RAPORT**

Lucrarea de laborator nr. 8

# La disciplina „Tehnici de Inginerie Inversa”

A efectuat: st. gr. SI-211 S. Chirita

A verificat: C. Maxim

Chișinău – 2024

Introducere

DLL hijacking este o tehnică prin care un atacator își inserează propriul fișier DLL într-o aplicație pentru a prelua controlul asupra acesteia sau pentru a executa cod malefic. Iată cum ar putea avea loc un atac DLL hijacking în Notepad++ folosind fișierul DLL SensApi.dll:

1. **Identificarea unei aplicații vulnerabile:** Un atacator identifică o aplicație care folosește SensApi.dll și care nu specifică explicit calea completă către DLL-ul respectiv.
2. **Plasarea unui fișier DLL malefic:** Atacatorul plasează un fișier DLL malefic (de exemplu, malware.dll) într-un director în care aplicația vulnerabilă își caută DLL-urile, dar care este accesibil pentru scriere pentru atacator (de exemplu, directorul de instalare al aplicației sau un director temporar).
3. **Pornirea aplicației:** Atacatorul lansează aplicația vulnerabilă.
4. **Încărcarea DLL-ului malefic:** Deoarece aplicația nu specifică calea completă către SensApi.dll, Windows va căuta DLL-ul în mai multe locații, inclusiv în directorul în care atacatorul a plasat DLL-ul malefic. Aplicația va încărca apoi DLL-ul malefic în loc de SensApi.dll.
5. **Execuția codului malefic:** DLL-ul malefic poate conține cod care să preia controlul asupra aplicației sau să execute acțiuni malefice, cum ar fi să fure informații sau să compromită securitatea sistemului.

Pentru a preveni un atac DLL hijacking, dezvoltatorii de software ar trebui să specifice întotdeauna calea completă către fișierele DLL pe care le utilizează în aplicațiile lor. De asemenea, ar trebui să se asigure că utilizatorii nu au permisiuni de scriere în directoarele în care aplicațiile își caută DLL-urile, pentru a preveni plasarea fișierelor DLL malefice în acele directoare.

Cautarea care „dll” este vulnerabil.

Filtrul din Process Monitor afișat în imagine nu are niciun criteriu de filtrare setat în prezent. Acest lucru înseamnă că afișează toate intrările din jurnalele Process Monitor.

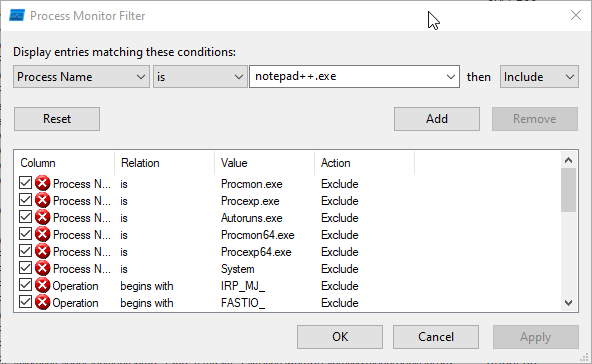


Figura 1 – Proces monitor fara nici un filtru

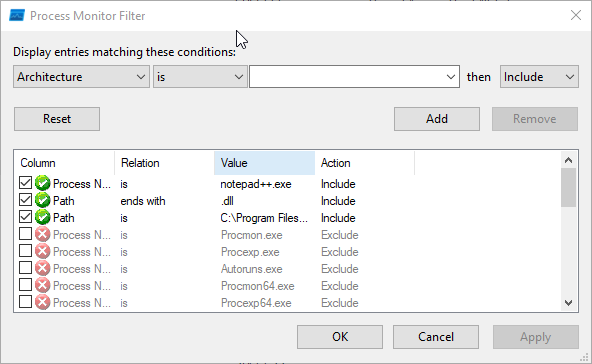
Filtrul din imagine are următoarele criterii setate:

* **Nume proces:** este notepad++.exe
* **Calea:** se termină cu .dll

Acest filtru va afișa toate intrările din jurnalele Process Monitor care îndeplinesc toate cele trei criterii. Cu toate acestea, nu va filtra toate fișierele DLL pentru programul notepad++.

**Motivul:**

Criteriul "Calea se termină cu .dll" va filtra doar fișierele care au extensia .dll. Cu toate acestea, programul notepad++ poate încărca și alte tipuri de fișiere DLL, cum ar fi fișierele DLL cu nume care nu se termină cu .dll. De asemenea, programul notepad++ poate încărca fișiere DLL din alte locații decât C:\Program Files.



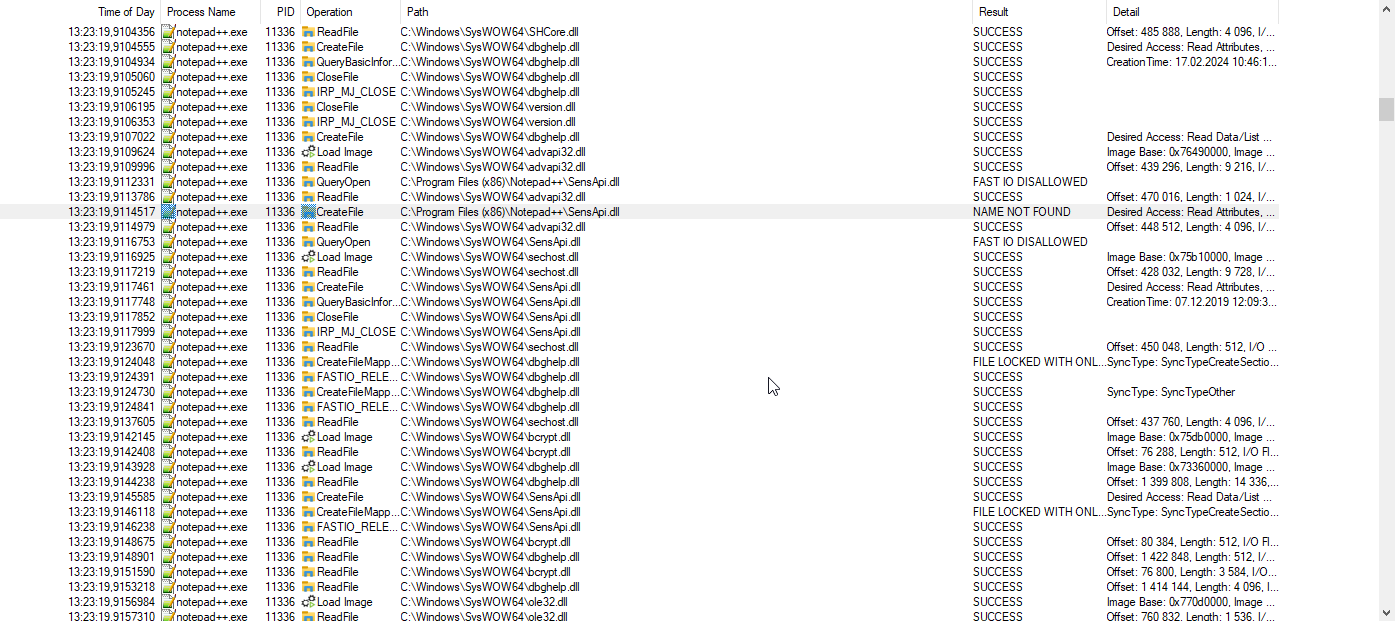
Figura 2 -Proces monitor cu fitru pentru Notepad++

Figura 3 – Detectarea fisierului vulnerabil

Conform imaginii furnizate, s-a identificat o vulnerabilitate în fișierul DLL ****SensApi.dll****. Fișierul este căutat mai întâi în directorul curent și, dacă nu este găsit, este încărcat din directorul ****SySWOW64****.

#include <windows.h>

#define DLLEXPORT extern "C" \_\_declspec(dllexport)

DLLEXPORT void calc (void);

bool is\_running;  // once

 bool APIENTRY DllMain(HANDLE hModule, DWORD dwReason, LPVOID lpReserved)

 {

  if( !is\_running ){

  is\_running = true;

    calc();                           //Acesta este ca un PoC(Proof of Concept =)) Start calculator

    HANDLE hmod;

    hmod = LoadLibrary("SensApi\_correct.dll");

  }

  return TRUE;

 }

DLLEXPORT void calc(void)

{

  HWND Hide;

  AllocConsole();

  Hide = FindWindowA("ConsoleWindowClass", NULL);

  ShowWindow(Hide,SW\_SHOW);

  system("calc");

 }

Acest cod DLL încearcă să ruleze Calculatorul Windows (calc.exe) atunci când este încărcat într-o aplicație. Funcția DllMain este utilizată pentru a verifica dacă DLL-ul este în curs de rulare pentru prima dată și, în acest caz, deschide o fereastră de consolă și lansează calc.exe folosind funcția system("calc"). Este important de menționat că acest cod este doar un exemplu demonstrativ și nu ar trebui să fie utilizat în mod real, deoarece poate fi considerat un comportament dăunător sau nedorit.

# Makefile for SensApi.dll project

all: SensApi.dll

SensApi.dll: SensApi.o

    dllwrap SensApi.o -o SensApi.dll

SensApi.o: SensApi.cpp

    g++ -c SensApi.cpp -o SensApi.o

clean:

    del SensApi.o

Această fișier Makefile este folosit pentru a compila un fișier sursă C++ (SensApi.cpp) într-un fișier DLL (SensApi.dll) folosind un compilator GNU. Iată o scurtă descriere a fiecărei secțiuni din fișierul Makefile:

1. **Regula all:** Această regula specifică că atunci când rulezi make all, Makefile-ul va compila SensApi.dll.
2. **Regula pentru SensApi.dll:** Această regula specifică că fișierul DLL SensApi.dll trebuie să fie creat din fișierul obiect SensApi.o folosind dllwrap.
3. **Regula pentru SensApi.o:** Această regula specifică că fișierul obiect SensApi.o trebuie să fie creat din fișierul sursă SensApi.cpp folosind compilatorul g++.
4. **Regula clean:** Această regula specifică că atunci când rulezi make clean, fișierul obiect SensApi.o va fi șters.

Pentru a utiliza acest Makefile, asigură-te că ai instalat GNU Make și un compilator C++ (de exemplu, g++) pe sistemul tău. Apoi, poți rula make all pentru a compila fișierul DLL SensApi.dll sau make clean pentru a șterge fișierul obiect SensApi.o.

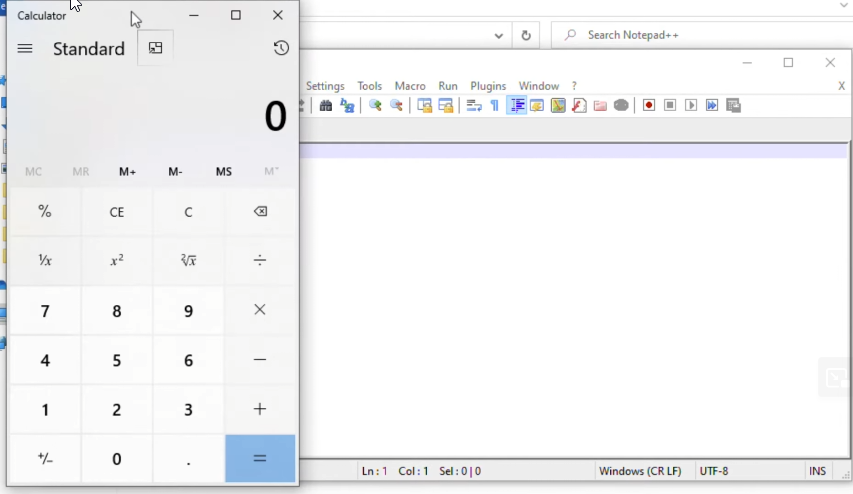


Figura 4 – PoC

Concluzie: În concluzie, pentru a proteja aplicațiile împotriva acestui tip de atac, este recomandat să specificați întotdeauna calea completă către fișierele DLL pe care le utilizați în aplicații și să restricționați permisiunile de scriere în directoarele în care aplicațiile își caută DLL-urile. De asemenea, este important să monitorizați activitatea aplicațiilor și să implementați măsuri de securitate suplimentare pentru a detecta și a preveni astfel de atacuri.