Training DLL Side-Loading for Red Team Ops

Por Jesús Domínguez

22/11/2022

¡Hola a todos, gracias por tomarse el tiempo de asistir a este taller!

Cualquier duda que tengas siéntete libre de hacérmela saber y con gusto la responderé de la mejor forma posible

$~: whoami

Offensive security researcher en Ocelot team by Metabase Q

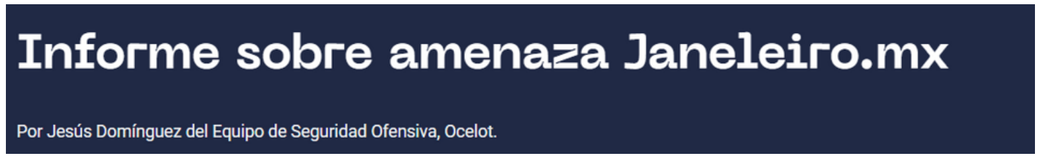
Infosec streamer en twitch.tv/notmalafama directo todos los jueves

Offensive Security Exploit Developer

¿Qué hace un security researcher?

Análisis de malware

Analizar el comportamiento de malware para identificar IOCs TTPs



Detección de vulnerabilidades

Detectar e implementar de vulnerabilidades durante ejercicios de red team

Identificando una DLL vulnerable

TODO LO QUE NECESITAS SABER PARA ENCONTRAR ESTA VULNERABILIDAD

¿Qué es una DLL?

Dynamic Link Library (DLL)

Como su nombre nos indica Dynamic Link Library (DLL) son archivos conocimos como bibliotecas de enlaces dinámicos. Estas bibliotecas ofrecen mediante sus "exports" funciones especificas a las aplicaciones que así lo requieren.

LoadLibrary Windows API

Esta API de Windows es muy popular para cargar en la memoria de un proceso una DLL, esta API recibe el nombre de la DLL que se quiere cargar en la memoria del proceso.

```C++
HMODULE LoadLibraryA(
  [in] LPCSTR lpLibFileName
);
```

Orden de búsqueda de las DLL

Es la forma en que standar que Microsoft define para la búsqueda de una DLL que se quiere cargar mediante LoadLibrary.

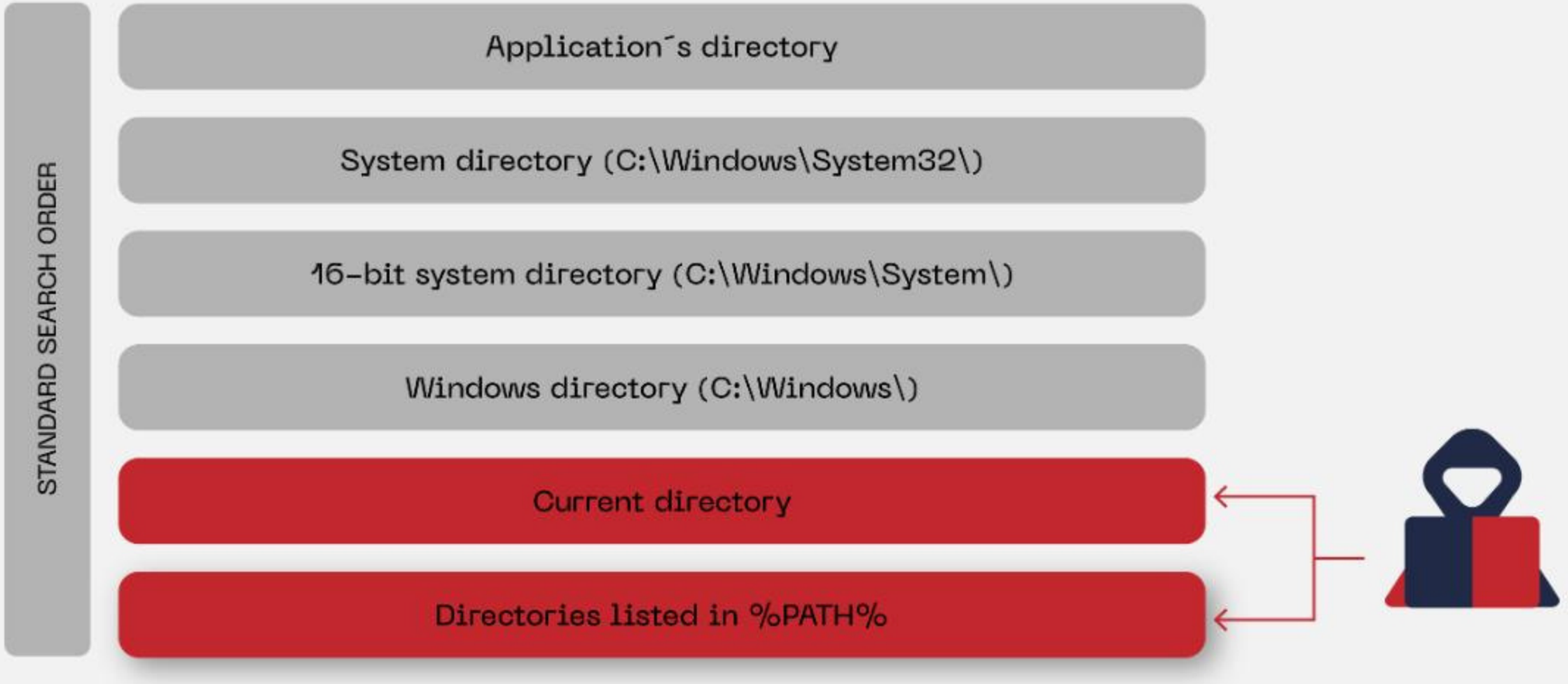


Figure 1.Traditional DLL search

Usando ProcMon para encontrar una DLL vulnerable

Process Monitor aka procmon es una herramienta de Windows que nos muestra en tiempo real el sistema de archivos, llaves de registro y process/thread activity.

Explotación

TODO LO QUE NECESITAS SABER PARA EXPLOTAR ESTA VULNERABILIDAD

Arquitectura básica de la DLL

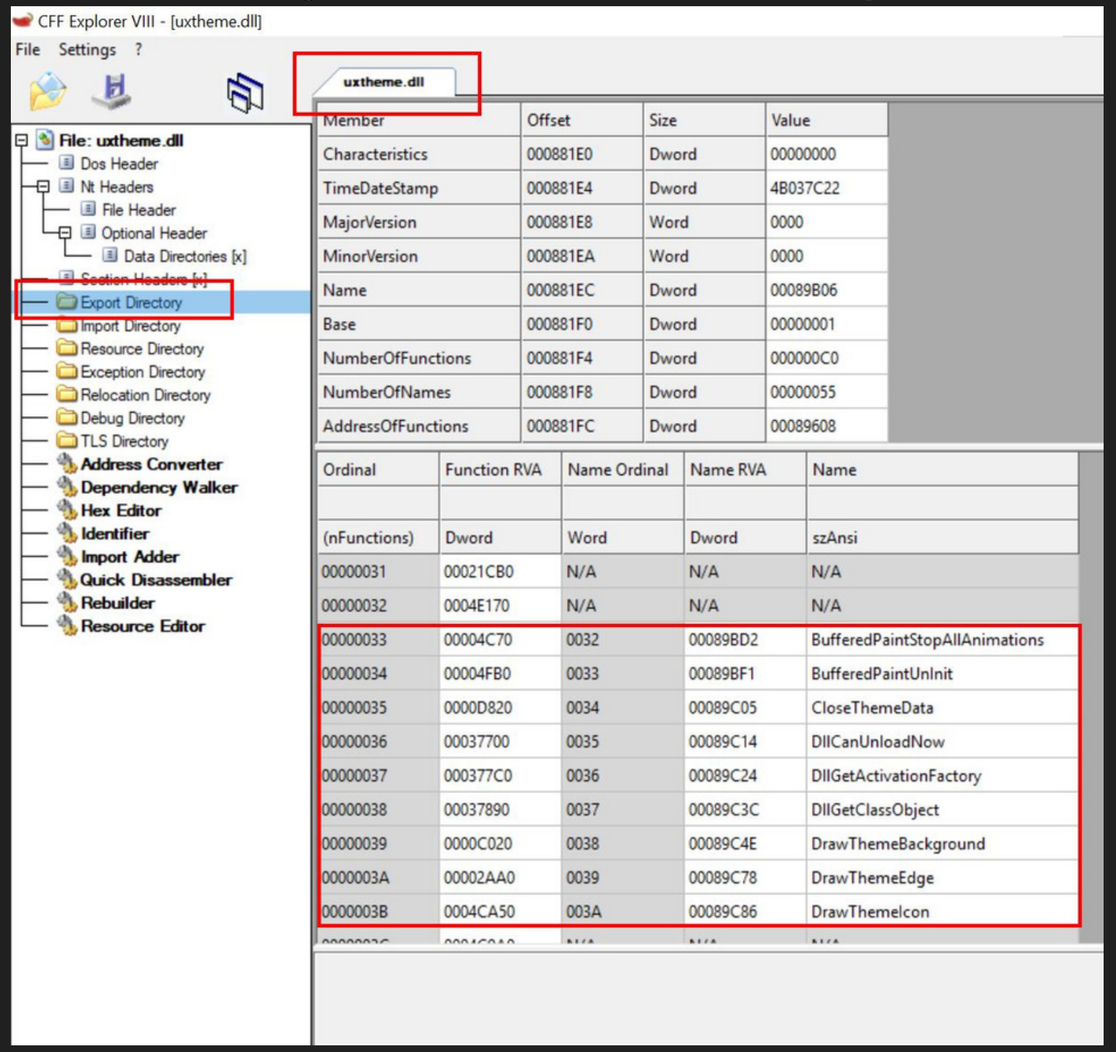
La parte más importante de la DLL se le conoce como Entry-Point.

La DLL cuenta con distintos escenarios donde la DLL va a ser llamada, estos son:

* DLL_PROCESS_ATTACH - El proceso carga la DLL
* DLL_THREAD_ATTACH - El proceso crea un nuevo thread
* DLL_THREAD_DETACH - El thread termina
* DLL_PROCESS_DETACH - El proceso cierra el handle de la DLL

Analizando los exports de la DLL

Además del DLL Entry-Point una DLL tiene funciones que expone para su uso por atrás aplicaciones, estas funciones son mejor conocidas como "exports".



Caso de uso: FileZilla DLL side-loading

FileZilla DLL side-loading

Para este caso usaremos la última versión del popular software de FTP FileZilla, a continuación vemos los pasos a seguir:

1. Mover la DLL a un directorio diferente al original
2. Agregar FZ_DATADIR con el path C:\Program Files\FileZilla FTP Client a las variables de entorno de Windows
3. Ejecutar procmon para ver las DLL que se cargan
4. De las DLL que salen seleccionamos libfilezilla-32.dll
5. Revisamos exports con CFF explorer
6. Con el script en python obtenemos los exports que vamos a pegar en el proyecto de visual studio
7. Abrimos un nuevo proyecto de DLL en Visual Studio 2022
8. Pegamos los exports
9. Compilamos y probamos

Mitigación

MITIGANDO DLL SIDE-LOADING

Recomendaciones para la mitigación

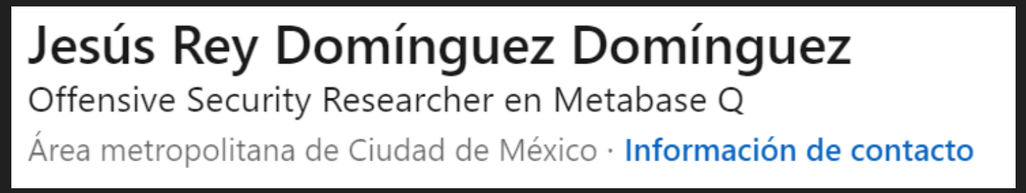
* Usar el path completo para llamar a la DLL
* Emplear herramientas automatizadas para detectar esta vulnerabilidad
* Usar alguna solución que sea capaz de bloquear DLL maliciosas cargadas por software legítimo

¡Muchas gracias por haber llegado al fina!

Me puedes encontrar así

Twitch / Youtube

LinkedIn



Twitter @chucho_domz