Ethical hacking

Topic: DLL_Injection and API_Hooking

Submitted by: Laiba Rehman

Submitted to : Sir Bilal

Table of content

DDL Injection

What is DLL Injection?

DLL injection is a technique used in cybersecurity where an attacker manipulates a running process by forcing it to load a dynamic-link library (DLL) file. DLL files are designed to contain code and data that can be used by multiple programs simultaneously. By injecting a DLL into a process, an attacker can execute foreign code within the context of that process.

The Process of DLL Injection

1. **Target Identification**: The attacker identifies the target process where the DLL will be injected.
2. **DLL Creation**: The DLL file containing the malicious code is created.
3. **Process Access**: The attacker opens the target process with specific privileges, such as PROCESS_CREATE_THREAD, PROCESS_QUERY_INFORMATION, PROCESS_VM_OPERATION, PROCESS_VM_WRITE, or PROCESS_VM_READ.
4. **Memory Allocation**: Memory is allocated within the target process using the VirtualAllocEx function.
5. **DLL Loading**: The LoadLibrary function is used to load the DLL into the process's memory space.

Malicious Uses of DLL Injection

For hackers, DLL injection provides several advantages:

- **Access Sensitive Information**: Hackers can extract sensitive data, such as passwords.
- **Modify Behavior**: They can alter the behavior of the target application.
- **Capture Keystrokes**: Keylogging activities can be performed.
- **Gain Persistence**: Injected DLLs can help in maintaining control over the target.
- **Network Propagation**: DLL injection can be used to spread malware across a network.

Defensive Uses of DLL Injection

Cybersecurity researchers and developers can use DLL injection for:

- **Debugging**: Debugging applications to understand their behavior.
- **Reverse Engineering**: Analyzing applications to find vulnerabilities and security weaknesses.
- **Testing Security**: Identifying and fixing potential security issues

Antivirus Detection

Traditional antivirus systems that rely on signature-based detection may struggle to identify DLL injection attacks. Modern antivirus solutions often use heuristic-based detection, which focuses on the behavior of the software rather than just its code. This approach can help in detecting malicious DLL injections by analyzing the actions performed by the software.

—---

API Hooking

Introduction

API hooking is a technique used in software development and debugging that allows developers to modify or monitor the behavior of an application by intercepting function calls.

What is API Hooking?

API hooking refers to the process of intercepting function calls in a software application to alter or monitor its behavior. This technique involves "hooking" into the application's API calls, allowing control over the execution of specific functions.

How API Hooking Works

API hooking typically involves creating a hook function that intercepts a call to a target function. When the target function is called, the hook function is executed first, and it can either modify the call or allow the original function to execute.

Types of API Hooking:

1. **Inline Hooking:**
   - In this method, part of the target function's code is replaced with the hook function's code. The hook function is inserted at the beginning of the target function, altering its execution path.
2. **Import Address Table (IAT) Hooking:**
   - This method involves modifying the Import Address Table (IAT) of the target application so that the application calls the hook function instead of the original function.

3. **VTable Hooking:**
    - This method modifies entries in the virtual table (VTable) used by object-oriented programming languages, redirecting function calls to the hook function.

Use Cases for API Hooking:

Debugging and Reverse Engineering: To understand how an application works internally.
Malware Analysis: To study the behavior of malware by intercepting its API calls.
Performance Monitoring: To monitor and optimize application performance by capturing and analyzing API calls.

Tools and Libraries for API Hooking:

Detours Library: Developed by Microsoft, this library is widely used for API hooking in Windows applications.
EasyHook: A popular open-source library for hooking in .NET applications.
Frida: A dynamic instrumentation toolkit that supports API hooking across multiple platforms.