# Code Injection and Hoking Techniques

# Contents

# DLL Injection

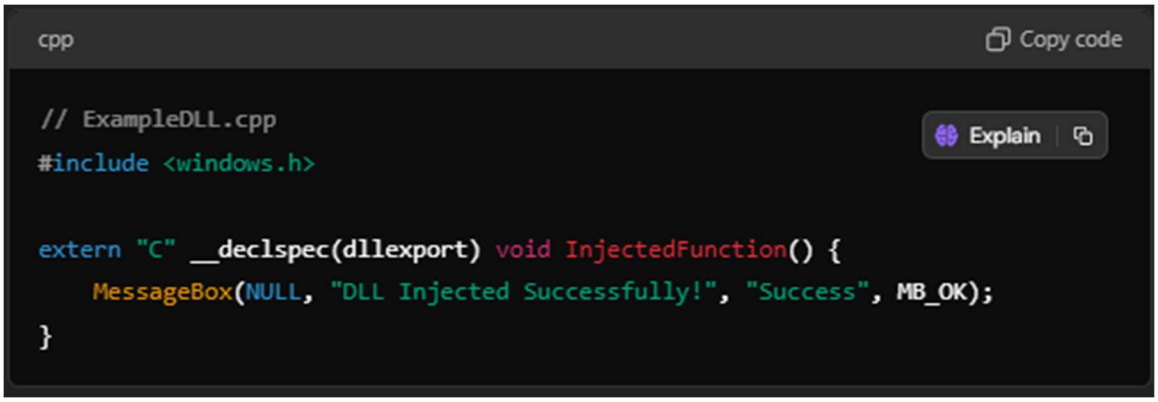
**Definition:**

DLL injection is a technique used to execute code in the address space of another process by injecting a DLL (Dynamic Link Library).

## Examples:

### 1. Simple DLL in C++ :

This shows a message box when injected.



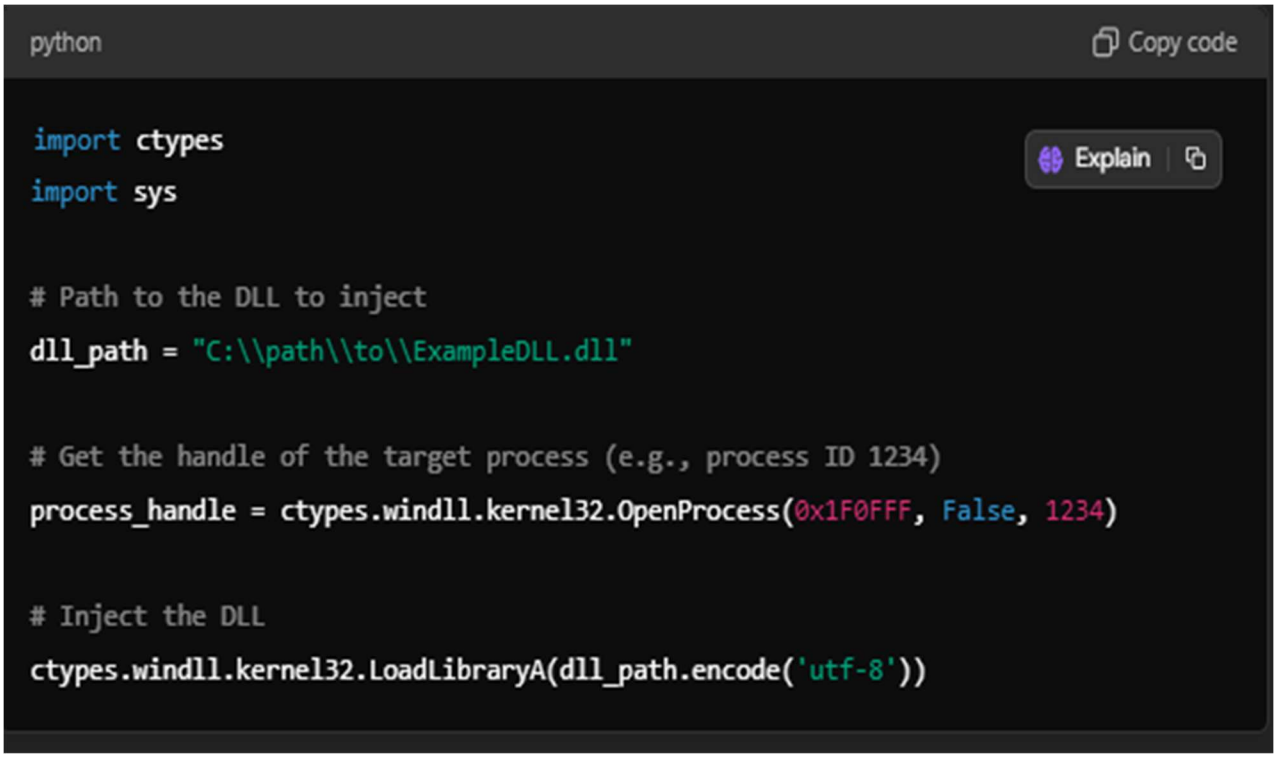

Compile this code into '**ExampleDLL.dll'.**

2. **Inject DLL:** Use a tool like '**DLL Injector**' to inject '**ExampleDLL.dll**' in to a target process.

For example, using **'Process Hacker'** we need to follow these steps:

- Open '**Process Hacker'**.
- Find the target process in the list.
- Right-click the process, select '**Inject DLL'**, and choose **'ExampleDll.dll**'.

**Example Command Using Python:** You can also use a Python script to inject a DLL using the **'ctypes'** library:




# API Hooking

**Definition:**

API Hooking involves intercepting calls to an API function and redirecting them to custom code.

1. **Simple API Hook in C++:**

This intercepts **'MessageBox'** calls and modifies the message:

```cpp
// HookExample.cpp
#include <windows.h>

// Original MessageBoxA function pointer
typedef int (WINAPI *MessageBoxA_t)(HWND, LPCSTR, LPCSTR, UINT);
MessageBoxA_t OriginalMessageBoxA = NULL;

// Hook function
int WINAPI HookedMessageBoxA(HWND hWnd, LPCSTR lpText, LPCSTR lpCaption, UINT uType) {
    return OriginalMessageBoxA(hWnd, "Hooked Message!", lpCaption, uType);
}

void InstallHook() {
    // Get the address of MessageBoxA
    HMODULE user32 = GetModuleHandle("user32.dll");
    OriginalMessageBoxA = (MessageBoxA_t)GetProcAddress(user32, "MessageBoxA");

    // Install the hook (this is a simplified example, real hooking requires more step
    // You may use libraries like Microsoft Detours or MinHook for a complete solution
}
```

2. **Using Cheat Engine:**
   - Open **'Cheat Engine'** and attach it to the target process.
   - Use the **'Memory View'** to find the API function (e.g **'MessageBoxA'**).
   - Write and execute a script to modify tha API call (e.g using Cheat Engine's Lua scripting feature).

**Example Command Using Cheat Engine:**

```lua
-- Example Lua script for Cheat Engine
local address = 0x12345678 -- Address of MessageBoxA in the target process
writeBytes(address, 0x90, 0x90, 0x90, 0x90) -- NOP (No Operation) out the original code
```

## Tools:

1. **Process Hacker:**
    - **Purpose:** Manage processes and their modules.
    - **Features:** View detailed information about processes, modules, and perform DLL injection.
2. **Cheat Engine:**
    - **Purpose:** Debug and modify running processes.
    - **Features:** Memory scanning, code injection, API Hooking, and debugging.

## Conclusion:

DLL injection and API Hooking are advanced techniques for modifying and analyzing software behavior. BY injecting DLLs or hooking API functions, you can alter how applications work. Practical examples using tools like Process Hacker and Cheat Engine demonstrate these techniques in action.