

Forum:[Hack Forums](#)

Thread: [HWID Checker Source](#)

frazdaz - May 10, 2011, 4:19 pm

Seen a few threads asking for sources so here's mine without using any external libraries. Wrote it a while ago and, yes, I should've used getaddrinfo instead of gethostbyname.

Yeah I know it's easy to beat, idc.

[Here](#)'s a pastebin link seeing as the word filter managed to fuck up the whitespacing.

```
Code #pragma comment(lib, "Ws2_32.lib")
#include <winsock2.h>
#include <Windows.h>
#include <iostream>
#include <string>
#define recvBuffSize 2000
const std::string host = "site",
                page = "hwid.txt";

int main(void)
{
    HW_PROFILE_INFO hwid;
    WSADATA wsData;
    if (WSAStartup(MAKEWORD(2, 2), &wsData) != 0)
        return 1;

    SOCKET sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
    if (sock == INVALID_SOCKET) {
        WSACleanup();
        return 1;
    }

    int check, i = 0;
    hostent *dns;
    char recieve[recvBuffSize];

    sockaddr_in sin;
    sin.sin_addr.S_un.S_addr = inet_addr(host.c_str());
    sin.sin_family = AF_INET;
    sin.sin_port = htons(80);
    std::string req;
    memset(&req, 0, sizeof(req));
    memset(&recieve, 0, sizeof(recieve));
    req = "GET /" + page + " HTTP/1.1\r\nHOST: " + host + "\r\nUser-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)\r\nConnection: Close\r\n\r\n";
    if (sin.sin_addr.s_addr == INADDR_NONE) {
        dns = gethostbyname(host.c_str());
        if (dns == NULL)
            return 1;
```

```

while (dns->h_addr_list[i]) { //loop through resolutions of hostname
    sin.sin_addr.s_addr = *(unsigned int*)(dns->h_addr_list[i]);
    check = connect(sock, (sockaddr*)&sin, sizeof(sin));
    if (check == SOCKET_ERROR)
        i++;
    else
        break;
}
} else {
    check = connect(sock, (sockaddr*)&sin, sizeof(sin));
    if (check == SOCKET_ERROR)
        return 1;
}

    check = send(sock, req.c_str(), req.size(), NULL);
    if (check == SOCKET_ERROR) {
std::cout << WSAGetLastError();
std::cin.get();
WSACleanup();
return 1;
    }

    recv(sock, recieve, sizeof(recieve), NULL);
    WSACleanup();

    std::string final = recieve; //convert char* to string so we can use string::erase & string::length
    int x;
    for (x = 0; x < final.length(); x++)
        if (final.at(x) == '\r' && final.at(x + 1) == '\n' && final.at(x + 2) == '\r' && final.at(x + 3) == '\n')
            break; //x is the pos where the double CRLF starts

    final.erase(0, x + 4); //remove all characters between start and the final '\n' in the CRLF

    GetCurrentHwProfile(&hwid);
    if (!strcmp((char*)hwid.szHwProfileGuid, final.c_str()))
        std::cout << "Passed HWID";
    else
        std::cout << "HWID Unsuccessful";

    std::cin.get();
}

```

sevetmax - November 19, 2015, 3:17 pm

thank you a lot for this man ! :)

tits4sale45 - November 20, 2015, 5:08 pm

Looks like some good code. Thanks!! ?