<시스템보안> 팀 프로젝트 최종 발표

PE-Analyzer 

Team Interrupt(팀 인터럽트)

리더 유영찬, 강필성, 김태현, 신동규, 이용위, 유승현

목차

프로젝트 PEA 개요

프로젝트 목표

PE(Portable Executable) 파일 포맷을 분석하고
디지털 포렌식에 활용하며 **실무에서도**
충분히 사용될 수 있는 툴을 개발하고자 함.

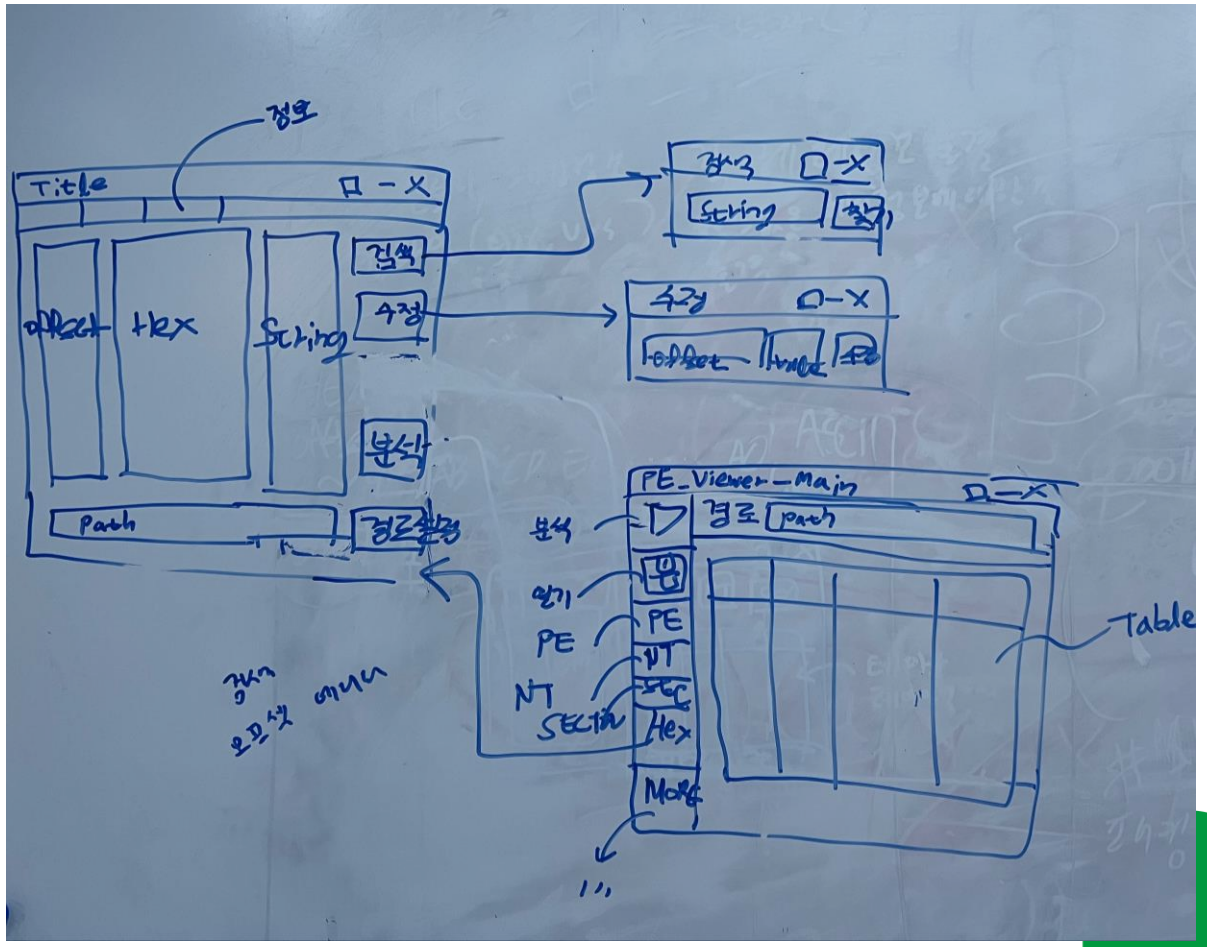
프로그램 설계

프로그램 사용자 타겟

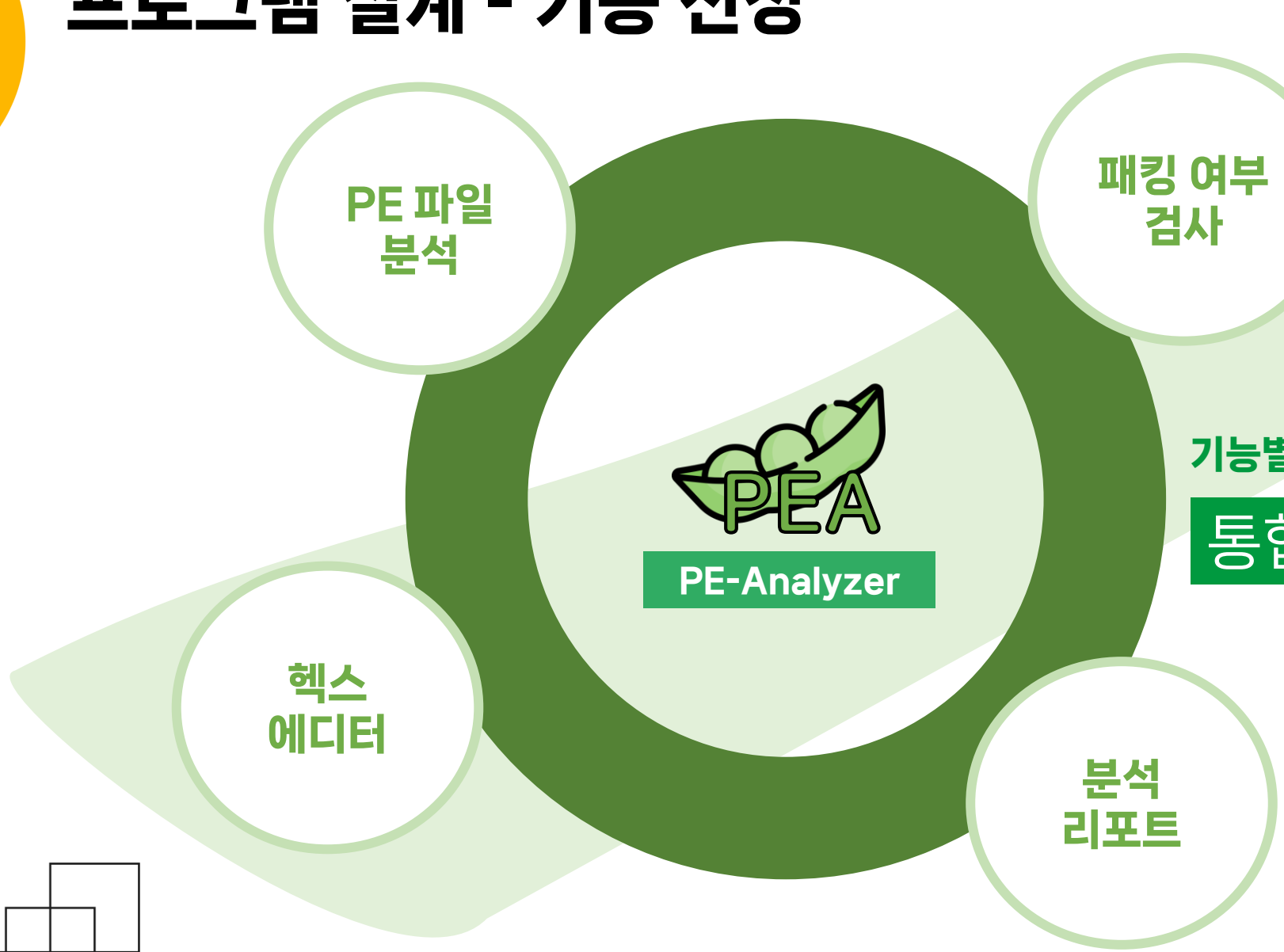
보안 분야 분석 초심자

GUI와 기능의 구성에서의 고려

"직관적이고 단순함."



프로그램 설계 - 기능 선정

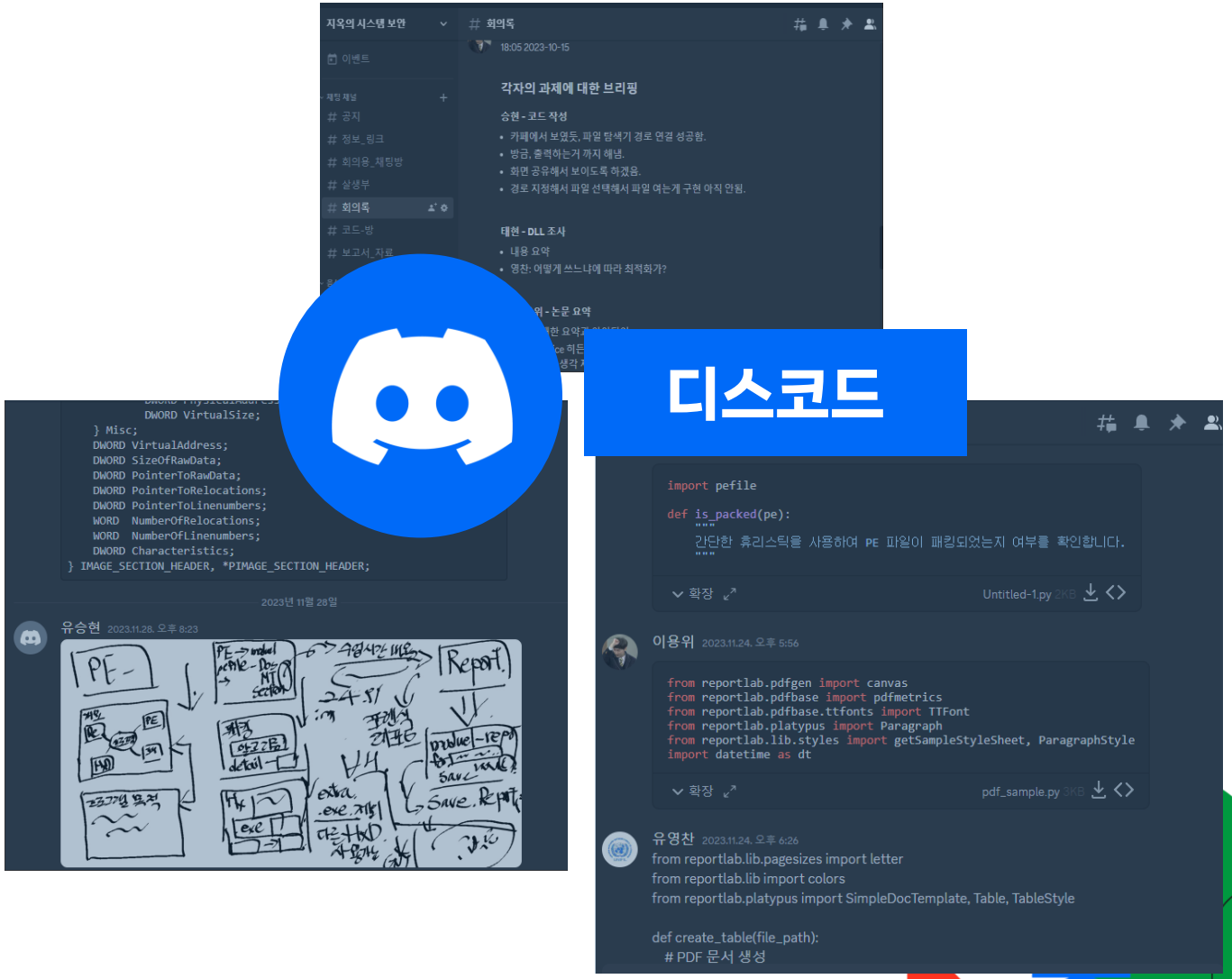


기능별로 파편화된 도구들..

통합해서 손쉽게

프로젝트 PEA 개요 - 협업과 버전 관리

- 음성 채팅 **프로그램인 '디스코드'**를 이용해 **파일관리와 정보공유, 버전관리**를 하였음.
- 회의를 통해 팀원들이 무엇을 공부했고, 어떤 작업을 했는지 공유하며 프로젝트를 진행함.
- **코드를 실시간으로 올려 서로 피드백과 테스트를 실시**
  → 효율적으로 버전관리를 하고자 하였음.
- 디스코드의 기능들을 최대한 활용해, 화면 공유 기능 등 여러 기능으로 프로젝트 관리를 원활하게 함.

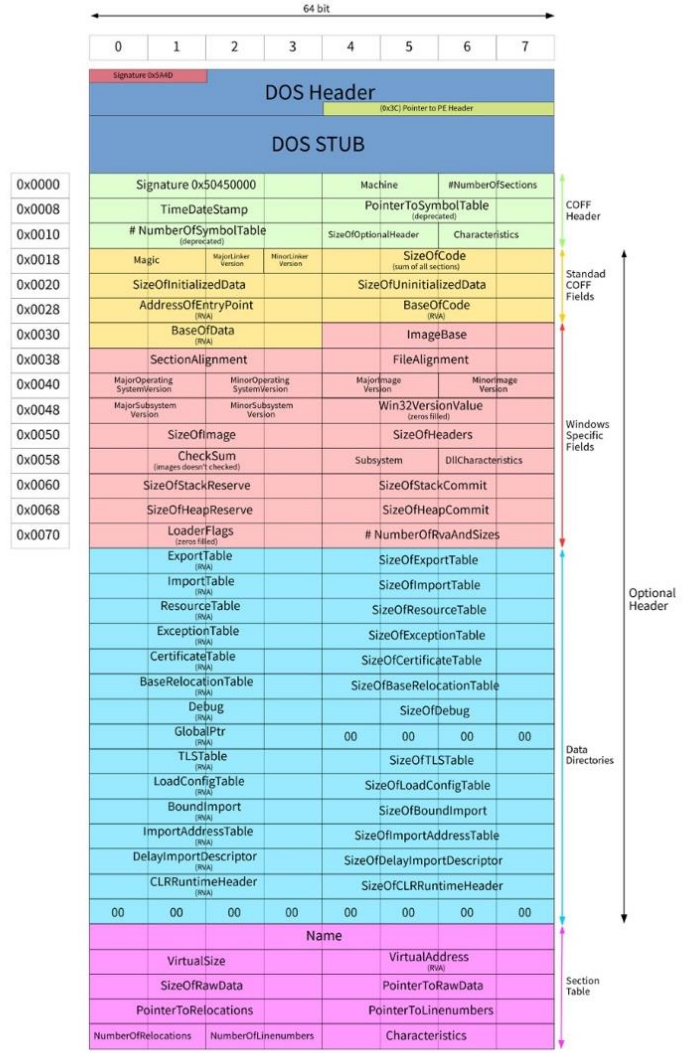


PE 파일 구조

- **PE파일**
  - 윈도우 운영체제에서 우리가 만들고 사용하는 파일이 다른 윈도우 운영체제의 PC로 옮겨져도 실행이 가능하도록 만들어 놓은 포맷 혹은 파일

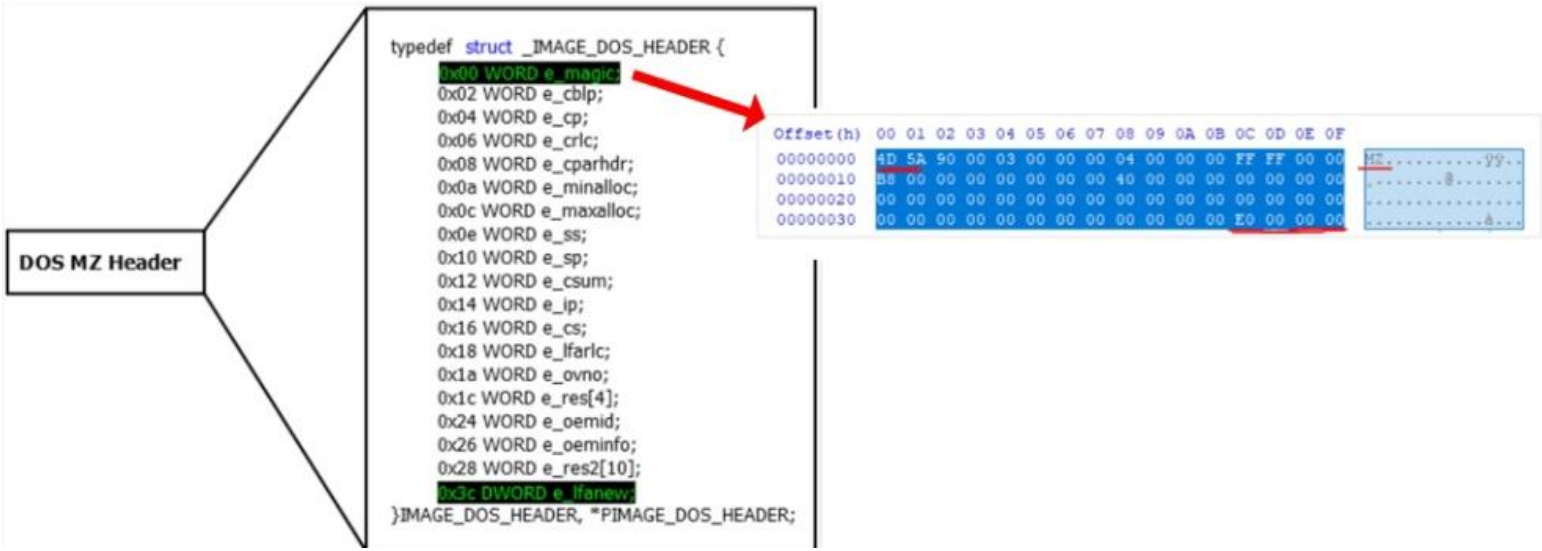
  →**"윈도우 상에서 배포 및 실행을 용이하게 만든 파일 포맷."**

- **PE파일 구조**
  - 다양한 정보들이 PE 헤더에 구조체 형식으로 저장



DOS 헤더

- 중요한 부분은 **DOS Signature 값인 "e_magic"**.
- 0x00에 위치한 이 값을 통해 실제로 해당 파일이 PE 파일인지 판단 가능.
- → PE 파일의 경우,e_magic에 4D 5A(MZ) 값으로 고정.
- 즉, e_magic 값이 4D 5A여야만 PE 파일로 인식해 실행 가능.



NT 헤더 & Section 헤더

- **NT헤더:** 크게 file 헤더, optional 헤더로 분류 가능.
  - file 헤더 구조에는 실행 가능한 CPU, 섹션의 수, 파일을 빌드한 날짜를 담는 정보가 있음.
  - optional 헤더에서는 가장 중요하게 볼 부분은 "IAT"
    - IAT: DLL 동적 라이브러리를 사용하기 위해 필요한 테이블.
      어떤 DLL혹은 함수(API)를 사용하는지 알 수 있기 때문에 분석하려는 PE파일이 어떤 행위를 할 수 있는지 파악 가능해서 중요.
- **Section 헤더:** PE 파일의 중요 데이터들이 카테고리(섹션)별로 저장된 헤더

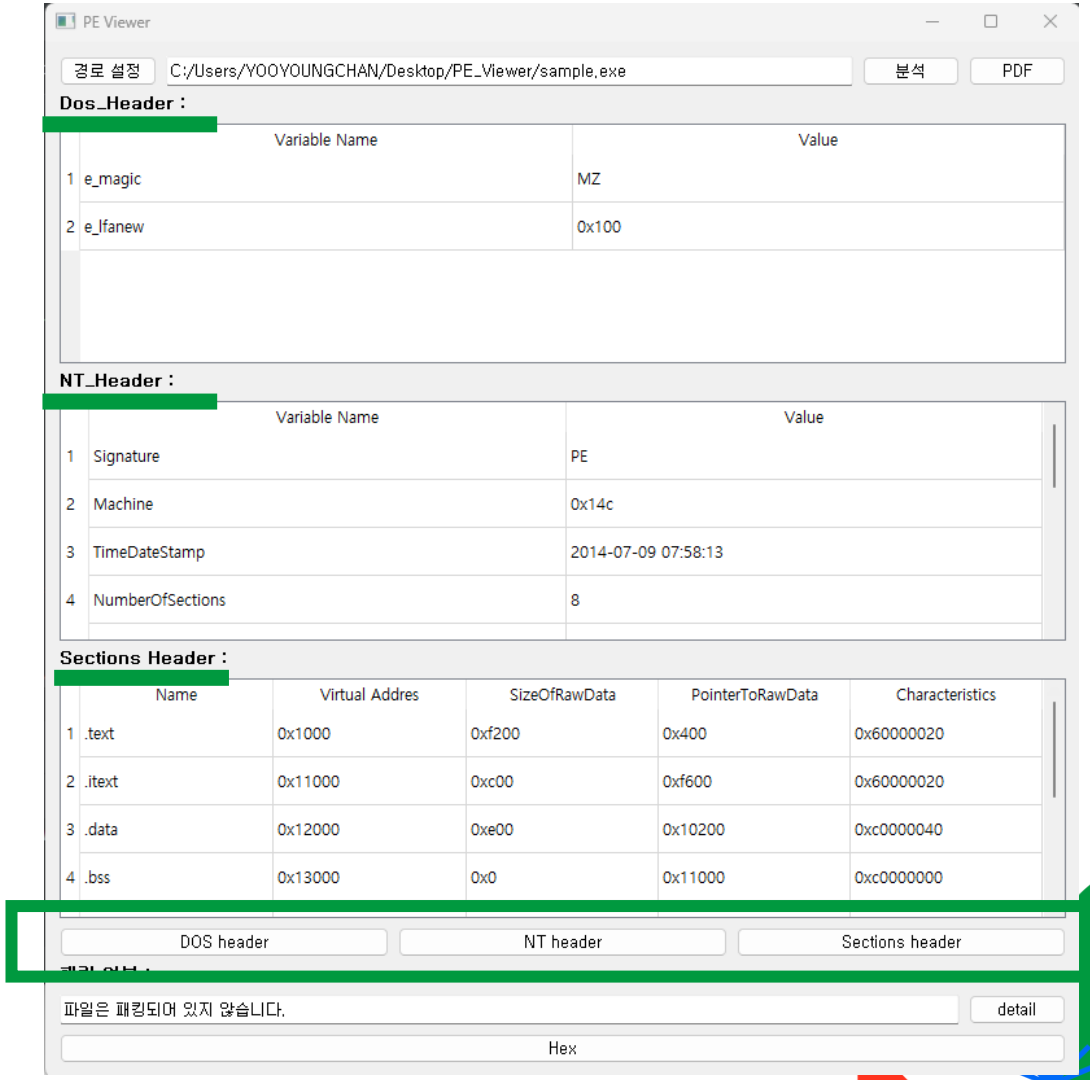
File Header

```
typedef struct _IMAGE_FILE_HEADER {
① WORD   Machine;
② WORD   NumberOfSections;
③ DWORD TimeDateStamp;
    DWORD PointerToSymbolTable;
    DWORD NumberOfSymbols;
    WORD   SizeOfOptionalHeader;
    WORD   Characteristics;
} IMAGE_NT_HEADER, *PIMAGE_NT_HEADER;
```

```
typedef struct _IMAGE_SECTION_HEADER {
  BYTE Name[IMAGE_SIZEOF_SHORT_NAME];
  union {
    DWORD PhysicalAddress;
    DWORD VirtualSize;
  } Misc;
  DWORD VirtualAddress;
  DWORD SizeOfRawData;
  DWORD PointerToRawData;
  DWORD PointerToRelocations;
  DWORD PointerToLinenumbers;
  WORD NumberOfRelocations;
  WORD NumberOfLinenumbers;
  DWORD Characteristics;
} IMAGE_SECTION_HEADER, *PIMAGE_SECTION_HEADER;
```

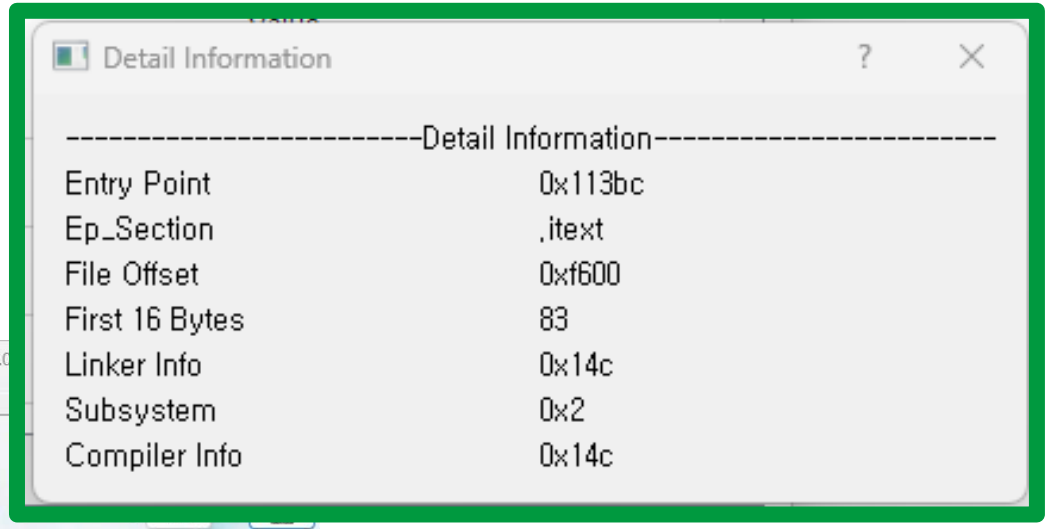
기능 소개 - [1] 헤더 정보 출력

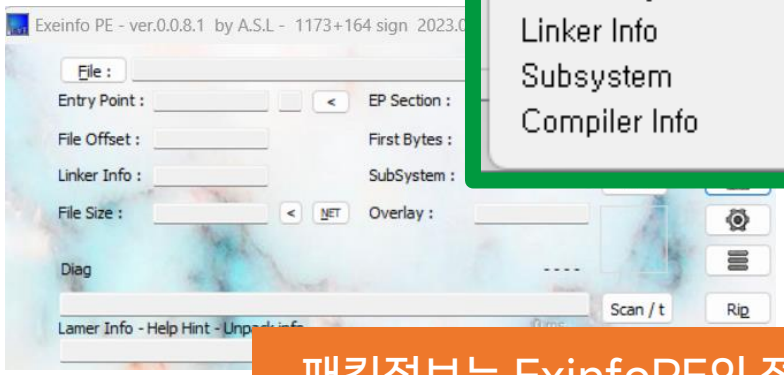
- 응용 프로그램(.exe, .dll, 등..)을 불러옴.
- 불러온 파일은 '분석' 버튼을 눌러 분석함.
- 분석에 성공한 경우, "성공".
  실패 시 오류 발생 메시지 출력
- **각 헤더의 버튼을 눌러 정보 조회 가능**
  - **Dos Header**
  - **NT Header**
  - **Sections Header**



기능 소개 - [2] 패킹 정보 출력

- **패킹 여부 출력 기능 제공**
- 자세한 정보는 'detail' 버튼을 누르면 새 창에 다음의 정보들을 출력
  - 엔트리 포인트
  - ep 섹션
  - 파일 오프셋
  - 첫 16 바이트 정보
  - 링커 정보
  - 컴파일 정보

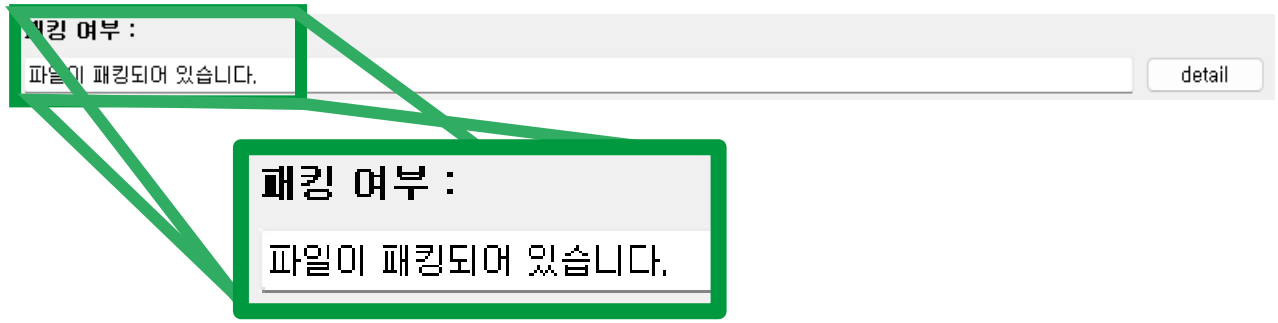


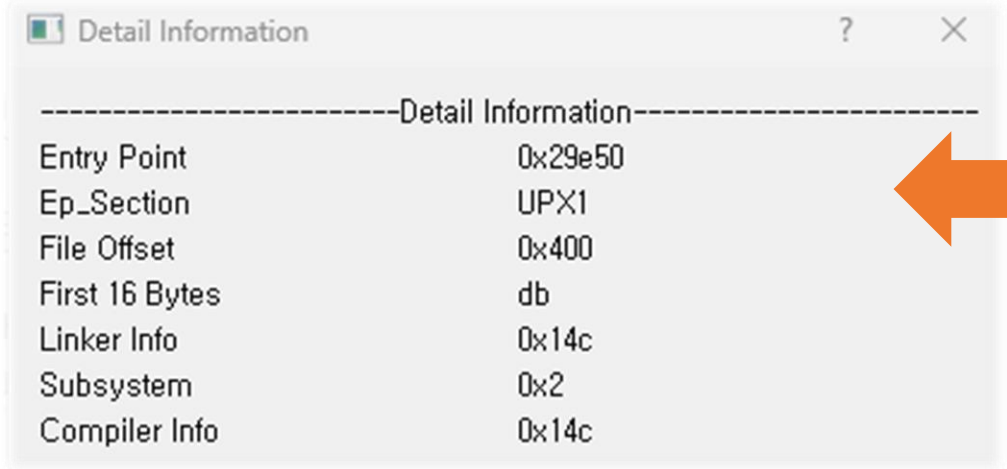


패킹정보는 ExinfoPE의 정보를 기반으로 제작

기능 소개 - [2] 패킹 정보 출력

- 만약 패킹 되어있는 파일을 분석한다면 **패킹여부에 '파일이 패킹 되어 있습니다.'** 문구가 출력됨.
- 자세한 정보에 어떻게 패킹 되어 있는지 확인 가능.

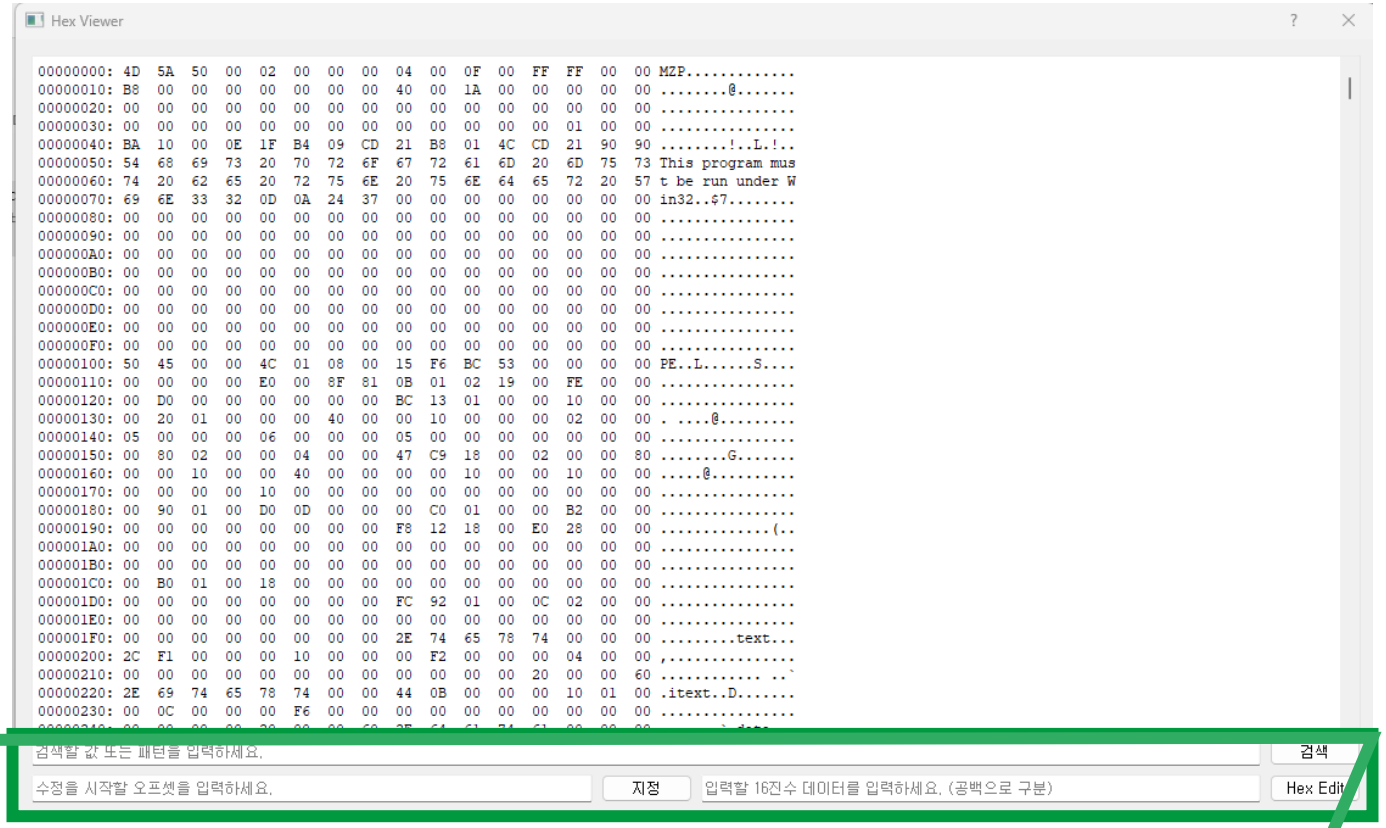




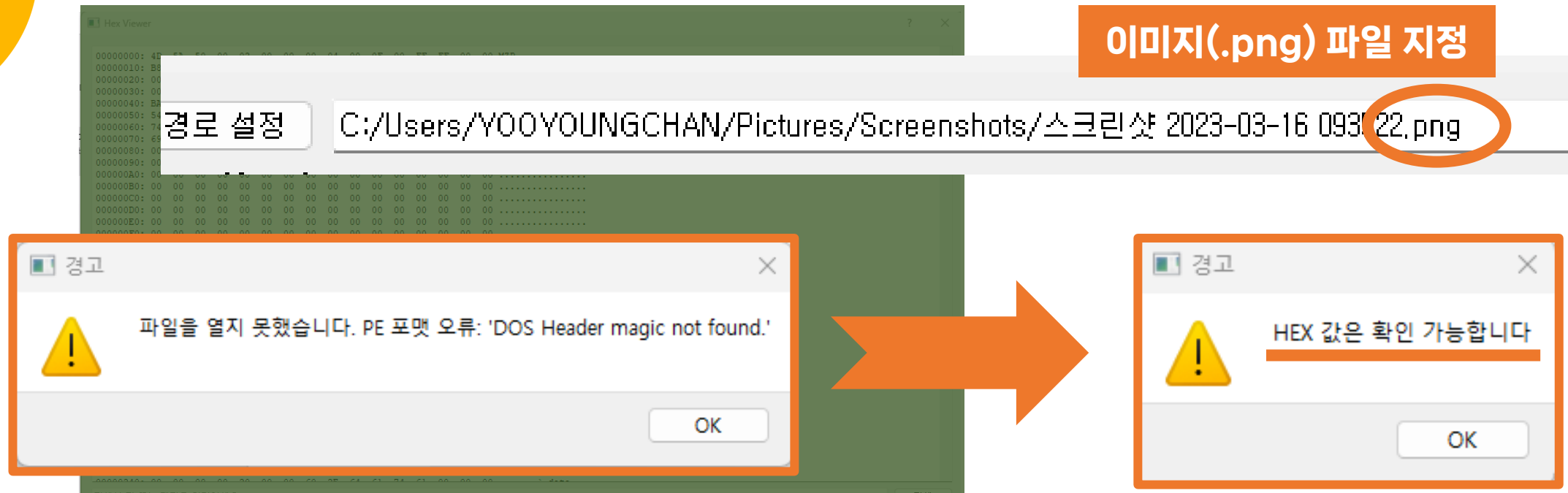
UPX로 패킹이 된 샘플파일을 분석했을 때 이런 정보가 출력되는 것을 확인 가능

기능 소개 - [3] 헥스 에디터

- "Hex" 버튼을 누르면 새 창에 **Hex 값들이 출력됨.**
- Hex 에디터 기능은 다른 파일도 분석 가능. (실행 파일이 아닌 이미지 파일 등..)
- **파일 오프셋이나 패턴을 검색** 1
- **파일의 오프셋을 지정하고 값들을 수정** 2



기능 소개 - [3] 헥스 에디터



사진파일을 지정한 뒤, 분석하면 경고창이 출력되나
헥스 값은 확인 가능하다는 문구가 출력됨.

기능 소개 - [4] 분석 리포트 출력 기능

- 분석한 파일들을 "PDF" 버튼을 누르면 PE_Report.pdf로 파일이 출력됨.
- 파일에는 **분석한 날짜, 시간, 분석한 헤더들의 정보, 패킹 파일에 관련된 정보가 출력** 1 2

1

[DOS Header]

실제 변수명	값	의미
e_magic	MZ	DOS Signature
e_lfanew	0x100	NT header offset

[NT Header]

실제 변수명	값	의미
Signature	PE	NF Signature
Machine	0x14c	CPU 별 고유값 (x
NumberOfSections	8	Section의 총 개수
SizeOfOptionalHeader	0xe0	OptionalHeader의
Characteristics	0x818f	이 파일의 속성
Magic	0x10b	Optional header를
SizeOfCode	0xfe00	IMAGE_SCN_CNT_
AddressOfEntryPoint	0x113bc	PE 파일이 메모리
ImageBase	0x400000	PE파일이 매핑되는
SectionAlignment	4096	메모리 상에서의 최소 섹션 단
FileAlignment	512	파일 상에서의 최소 섹션 단위

[Sections Header]

Name	Section 이름
VirtualAddress	섹션의 RAV(ImageBase + VA)를 위한 VA 값
SizeOfRawData	파일 상에서 섹션이 차지하는 크기

[Details about Packed File]

Entry Point	0x113bc
Ep_Section	.itext
File Offset	0xf600
First 16 Bytes	b'₩x83'
Linker Info	0x14c
Subsystem	0x2
Compiler Info	0x14c

컴파일러 정보: intel 368/x86 = 0x14c , intel 64 = 0x0200 , AMD64 = 0x8664

2

PE 분석 보고서

2023 December 02 Saturday . 14시 33분 31초

[DOS Header]

실제 변수명	값	의미
e_magic	MZ	DOS Signature
e_lfanew	0x100	NT header offset

[NT Header]

실제 변수명	값	의미
Signature	PE	NF Signature
Machine	0x14c	CPU 별 고유값 (x86 = 0x14c / x64 = 0x8664)
NumberOfSections	8	Section의 총 개수
SizeOfOptionalHeader	0xe0	OptionalHeader의 크기
Characteristics	0x818f	이 파일의 속성
Magic	0x10b	Optional header를 구분하는 Signature (32bit=10b / 64bit=20b)
SizeOfCode	0xfe00	IMAGE_SCN_CNT_CODE 속성을 갖는 섹션들의 총 사이즈 크기
AddressOfEntryPoint	0x113bc	PE 파일이 메모리 로드 후 처음 실행되어야 하는 코드 주소
ImageBase	0x400000	PE파일이 매핑되는 시작주소
SectionAlignment	4096	메모리 상에서의 최소 섹션 단위
FileAlignment	512	파일 상에서의 최소 섹션 단위

[Sections Header]

Name	Section 이름
VirtualAddress	섹션의 RAV(ImageBase + VA)를 위한 VA 값
SizeOfRawData	파일 상에서 섹션이 차지하는 크기
PointerToRawData	파일 상에서 섹션이 시작하는 위치
Characteristics	섹션의 특징을 나타냄

Name	Virtual Address	SizeOfRawData	PointerToRawData	Characteristics
.text	0x1000	0xf200	0x400	0x60000020
.itext	0x11000	0xc00	0xf600	0x60000020
.data	0x12000	0xe00	0x10200	0xc0000040
.bss	0x13000	0x0	0x11000	0xc0000000
.idata	0x19000	0xe00	0x11000	0xc0000040
.tls	0x1a000	0x0	0x11e00	0xc0000000
.rdata	0x1b000	0x200	0x11e00	0x40000040
.rsrc	0x1c000	0xb200	0x12000	0x40000040

[Details about Packed File]

Entry Point	0x113bc
Ep_Section	.itext
File Offset	0xf600
First 16 Bytes	b'₩x83'
Linker Info	0x14c
Subsystem	0x2
Compiler Info	0x14c

컴파일러 정보: intel 368/x86 = 0x14c , intel 64 = 0x0200 , AMD6

PE_Report.pdf

PDF

사용한 기술 및 도구

파이썬

파이참

PyQt

디스코드


프로그램 시현
https://youtu.be/npsuiVakZ1c

감사합니다.