

Formato PE (Portable Executable)

Arquivo Analisado

```
compilation_example.c
1  #include <stdio.h>
2
3  #define FORMAT_STRING "%s"
4  #define MESSAGE      "Hello, world!\n"
5
6  int
7  main(int argc, char *argv[]) {
8      printf(FORMAT_STRING, MESSAGE);
9      return 0;
10 }
11
```

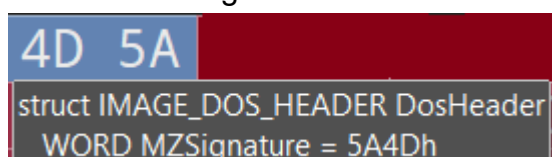
Ferramentas usadas

- 010 Editor
- Detect It Easy

Forma de um PE

Cabeçalho MZ do DOS
Fragmento(stub) do DOS
Cabeçalho do arquivo
=====
Diretório de Dados
Cabeçalhos das seções
Seção 1
Seção 2
...
Seção N

- 4D 5A -> MZ signature



```
4D 5A
struct IMAGE_DOS_HEADER DosHeader
WORD MZSignature = 5A4Dh
```

- 0x40000000 -> deslocamento para DOS-stub



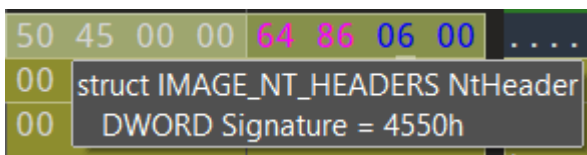
```
0000:0040  0e 1f ba 0e
```

- | | | | | | | | | | | | | | | | | | |
|--|------|------|----------|-----|----|----|----|----|----|----|----|----|----|----|----|----|-------------|
| e_ifanew | 003c | LONG | 000000f8 | Hex | | | | | | | | | | | | | |
| <div> <div>Hex</div> <div>Strings</div> </div> | | | | | | | | | | | | | | | | | |
| Endereço | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0a | 0b | 0c | 0d | 0e | 0f | Símbolos |
| 0000:0000 | 4d | 5a | 90 | 00 | 03 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | ff | ff | 00 | 00 | MZ..... |
| 0000:0010 | b8 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |@..... |
| 0000:0020 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | |
| 0000:0030 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | f8 | 00 | 00 | 00 | |

Hex		Strings	
Endereço	00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f	Símbolos	
0000:00f8	50 45 00 00	PE..	

- | | | | | | |
|------|-------------|-------------|-------------|-------------|------------------|
| 0040 | 0E 1F BA 0E | 00 B4 09 CD | 21 B8 01 4C | CD 21 54 68 | ..°...!.,.Lí!Th |
| 0050 | 69 73 20 70 | 72 6F 67 72 | 61 6D 20 63 | 61 6E 6E 6F | is program canno |
| 0060 | 74 20 62 65 | 20 72 75 6E | 20 69 6E 20 | 44 4F 53 20 | t be run in DOS |
| 0070 | 6D 6F 64 65 | 2E 0D 0D 0A | 24 00 00 00 | 00 00 00 00 | mode....\$..... |
| 0080 | 67 58 E0 03 | 23 39 8E 50 | 23 39 8E 50 | 23 39 8E 50 | gXà.#9ŽP#9ŽP#9ŽP |
| 0090 | 2A 41 1D 50 | 29 39 8E 50 | B0 59 8F 51 | 20 39 8E 50 | *A.P)9ŽP°Y.Q 9ŽP |
| 00A0 | B0 59 8D 51 | 21 39 8E 50 | B0 59 8B 51 | 32 39 8E 50 | °Y.Q!9ŽP°Y,Q29ŽP |
| 00B0 | B0 59 8A 51 | 28 39 8E 50 | 4E 64 8F 51 | 21 39 8E 50 | °YŠQ(9ŽPNd.Q!9ŽP |
| 00C0 | 23 39 8F 50 | 0C 39 8E 50 | 9A 58 87 51 | 22 39 8E 50 | #9.P.9ŽPŠX#Q"9ŽP |
| 00D0 | 9A 58 71 50 | 22 39 8E 50 | 9A 58 8C 51 | 22 39 8E 50 | šXqP"9ŽPŠXŒQ"9ŽP |
| 00E0 | 52 69 63 68 | 23 39 8E 50 | 00 00 00 00 | 00 00 00 00 | Rich#9ŽP..... |

- ```
typedef struct _IMAGE_NT_HEADERS {
 DWORD Signature;
 IMAGE_FILE_HEADER FileHeader;
 IMAGE_OPTIONAL_HEADER OptionalHeader;
} IMAGE_NT_HEADERS, *PIMAGE_NT_HEADERS;
```



```

64 86 06 00PE...d1..
F0 struct IMAGE_NT_HEADERS NtHeader
00 struct IMAGE_FILE_HEADER FileHeader
01 enum IMAGE_MACHINE Machine = AMD64 (8664h)

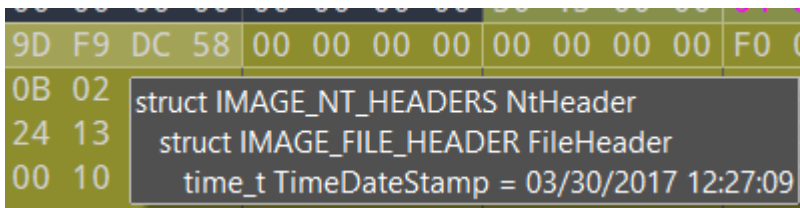
```

```
typedef struct _IMAGE_FILE_HEADER {
 WORD Machine;
 WORD NumberOfSections;
 DWORD TimeDateStamp;
 DWORD PointerToSymbolTable; // ponteiro para tabela de símbolos
 DWORD NumberOfSymbols; // ponteiro para número de símbolos
 WORD SizeOfOptionalHeader; //
 WORD Characteristics;
} IMAGE_FILE_HEADER, *PIMAGE_FILE_HEADER;
```

Após a primeira estrutura, temos a estrutura FileHeader, primeiro campo da estrutura acima é Machine que é um `enum` que diz que tipo de arquitetura esse executável é previsto para rodar; No caso do nosso PE é arch AMD de 64 bits.

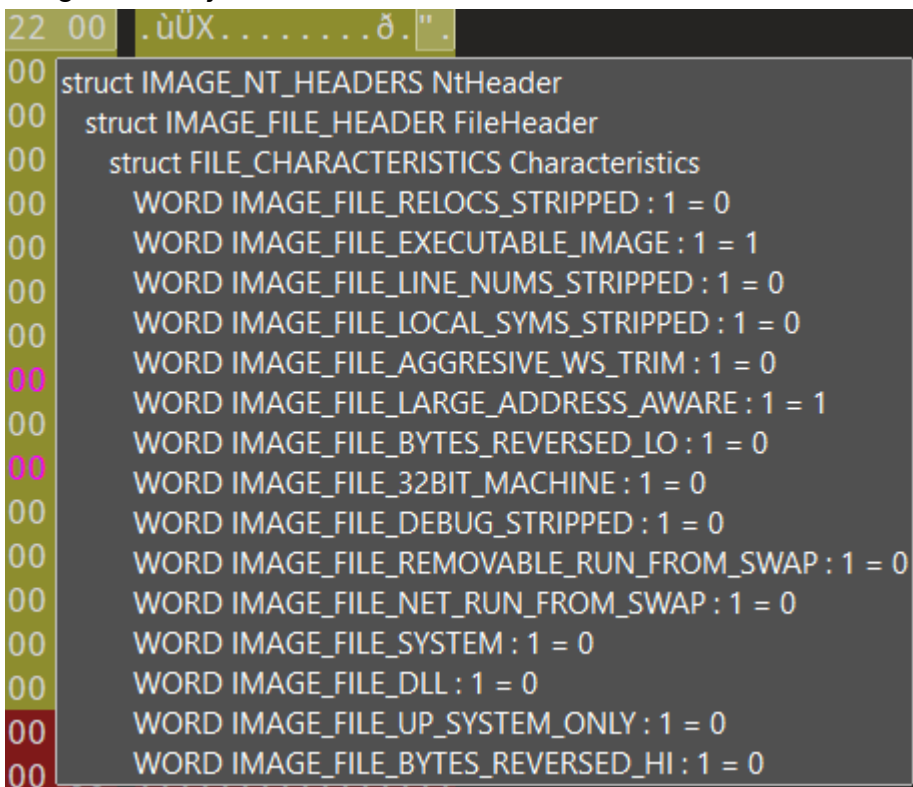
Em 0x0006 temos o número de seções;

TimeDataStamp corresponde ao número de segundos decorridos a partir de 01/01/1970. 0x58DCF99D



Characteristics é uma máscara de bits(16 bits). 0x0022

O campo Characteristics contém sinalizadores que indicam os atributos do arquivo de imagem ou objeto.



| Sinalizador                    | Valor  | Descrição                                                                                                                                                                                                                                                                                                                                                       |
|--------------------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IMAGE_FILE_RELOCS_STRIPPED     | 0x0001 | Somente imagem, Windows CE e Microsoft Windows NT e posterior. Isso indica que o arquivo não contém realocações de base e, portanto, deve ser carregado no endereço base preferido. Se o endereço base não estiver disponível, o carregador relatará um erro. O comportamento padrão do vinculador é remover realocações de base de arquivos executáveis (EXE). |
| IMAGE_FILE_EXECUTABLE_IMAGE    | 0x0002 | Somente imagem. Isso indica que o arquivo de imagem é válido e pode ser executado. Se esse sinalizador não estiver definido, ele indicará um erro de vinculador.                                                                                                                                                                                                |
| IMAGE_FILE_LINE_NUMS_STRIPPED  | 0x0004 | Somente imagem. Isso indica que o arquivo de imagem é válido e pode ser executado. Se esse sinalizador não estiver definido, ele indicará um erro de vinculador.                                                                                                                                                                                                |
| IMAGE_FILE_LOCAL_SYMS_STRIPPED | 0x0008 | As entradas da tabela de símbolos COFF para símbolos locais foram removidas. Esse sinalizador está obsoleto e deve ser zero.                                                                                                                                                                                                                                    |
| IMAGE_FILE_AGGRESSIVE_WS_TRIM  | 0x0010 | Obsoleto. Conjunto de trabalho com corte agressivo. Esse sinalizador foi preterido no Windows 2000 e posterior e deve ser zero.                                                                                                                                                                                                                                 |
| IMAGEFILE_LARGE_ADDRESS_AWARE  | 0x0020 | O aplicativo pode processar > endereços de 2 GB.                                                                                                                                                                                                                                                                                                                |
| IMAGE_FILE_BYTES_REVERSED_LO   | 0x0080 | Little endian: o LSB (bit menos significativo) precede o MSB (bit mais significativo) na memória. Esse sinalizador está obsoleto e deve ser zero                                                                                                                                                                                                                |
| IMAGE_FILE_32BIT_MACHINE       | 0x0100 | A máquina se baseia em uma arquitetura de palavra de 32 bits.                                                                                                                                                                                                                                                                                                   |
| IMAGE_FILE_DEBUG_STRIPPED      | 0x0200 | As informações de depuração são removidas do arquivo de imagem.                                                                                                                                                                                                                                                                                                 |

| Sinalizador                       | Valor  | Descrição                                                                                                                                                                                   |
|-----------------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IMAGEFILE_REMOVABLE_RUN_FROM_SWAP | 0x0400 | Se a imagem estiver em mídia removível, carregue-a totalmente e copie-a para o arquivo de troca.                                                                                            |
| IMAGE_FILE_NET_RUN_FROM_SWAP      | 0x0800 | Se a imagem estiver em mídia de rede, carregue-a totalmente e copie-a para o arquivo de troca.                                                                                              |
| IMAGE_FILE_SYSTEM                 | 0x1000 | O arquivo de imagem é um arquivo de sistema, não um programa de usuário.                                                                                                                    |
| IMAGE_FILE_DLL                    | 0x2000 | O arquivo de imagem é uma DLL (biblioteca de vínculo dinâmico). Esses arquivos são considerados arquivos executáveis para quase todos os fins, embora não possam ser executados diretamente |
| IMAGE_FILE_UP_SYSTEM_ONLY         | 0x4000 | O arquivo deve ser executado apenas em uma máquina com um único processador.                                                                                                                |
| IMAGE_FILE_BYTES_REVERSED_HI      | 0x8000 | Big endian: o MSB precede o LSB na memória. Esse sinalizador está obsoleto e deve ser zero.                                                                                                 |
|                                   | 0x0040 | O sinalizador é reservado para uso futuro.                                                                                                                                                  |

## Agora falaremos da estrutura IMAGE\_OPTIONAL\_HEADER.

- Fornece infos ao loader;
- Arquivos de objeto não os contém, por isso o nome opcional;
- Arquivos de imagem(**executável**) é obrigatório;
- Abaixo a struct que representa o cabeçalho opcional

```
typedef struct _IMAGE_OPTIONAL_HEADER64 {
 WORD Magic;
 BYTE MajorLinkerVersion;
 BYTE MinorLinkerVersion;
 DWORD SizeOfCode;
 DWORD SizeOfInitializedData;
 DWORD SizeOfUninitializedData;
 DWORD AddressOfEntryPoint;
 DWORD BaseOfCode;
 ULONGLONG ImageBase;
```

```

DWORD SectionAlignment;
DWORD FileAlignment;
WORD MajorOperatingSystemVersion;
WORD MinorOperatingSystemVersion;
WORD MajorImageVersion;
WORD MinorImageVersion;
WORD MajorSubsystemVersion;
WORD MinorSubsystemVersion;
DWORD Win32VersionValue;
DWORD SizeOfImage;
DWORD SizeOfHeaders;
DWORD CheckSum;
WORD Subsystem;
WORD DllCharacteristics;
ULONGLONG SizeOfStackReserve;
ULONGLONG SizeOfStackCommit;
ULONGLONG SizeOfHeapReserve;
ULONGLONG SizeOfHeapCommit;
DWORD LoaderFlags;
DWORD NumberOfRvaAndSizes;
IMAGE_DATA_DIRECTORY DataDirectory[IMAGE_NUMBEROF_DIRECTORY_ENTRIES];
} IMAGE_OPTIONAL_HEADER64, *PIMAGE_OPTIONAL_HEADER64;

```

## Campos padrão de cabeçalho opcionais (somente imagem)

Contém infos importantes para o carregamento do arquivo executável;

Cada retângulo referencia uma campo na tabela abaixo em ordem da esquerda para direita.

| Deslocamento | Size | Campo            | Descrição                                                                                                                                                                                                                     |
|--------------|------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0            | 2    | Magic            | O inteiro sem sinal que identifica o estado do arquivo de imagem. O número mais comum é 0x10B, que o identifica como um arquivo executável normal. 0x107 o identifica como uma imagem ROM, e 0x20B, como um executável PE32+. |
| 2            | 1    | MajorLinkVersion | O número da versão principal do vinculador.                                                                                                                                                                                   |
| 3            | 1    | MinorLinkVersion | O número da versão secundária do vinculador.                                                                                                                                                                                  |

| Deslocamento | Size | Campo                   | Descrição                                                                                                                                                                                                                                                                                                                                                       |
|--------------|------|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4            | 4    | SizeOfCode              | O tamanho da seção de código (texto) ou a soma de todas as seções de código quando há várias seções                                                                                                                                                                                                                                                             |
| 8            | 4    | SizeOfInitializedData   | O tamanho da seção de dados inicializados ou a soma de todas essas seções quando há várias seções.                                                                                                                                                                                                                                                              |
| 12           | 4    | SizeOfUninitializedData | O tamanho da seção de dados não inicializados (BSS) ou a soma de todas essas seções quando há várias seções de BSS.                                                                                                                                                                                                                                             |
| 16           | 4    | AddressOfEntryPoint     | O endereço do ponto de entrada relativo à base de imagem quando o arquivo executável é carregado na memória. Para imagens de programa, este é o endereço inicial. Para drivers de dispositivo, esse é o endereço da função de inicialização. Um ponto de entrada é opcional para DLLs. Quando não houver ponto de entrada presente, esse campo deverá ser zero. |
| 20           | 4    | BaseOfCode              | O endereço que é relativo à base de imagem da seção de início de código quando a imagem é carregada na memória                                                                                                                                                                                                                                                  |

**A próxima tabela fala dos campos específicos do windows para o cabeçalho opcional**

|             |             |             |             |             |             |
|-------------|-------------|-------------|-------------|-------------|-------------|
|             |             |             |             | 00 00 00 40 | 01 00 00 00 |
| 00 10 00 00 | 00 02 00 00 | 06 00       | 00 00       | 00 00 00 00 | 00 00 00 00 |
| 06 00       | 00 00       | 00 00 00 00 | 00 70 00 00 | 00 04 00 00 |             |
| 00 00 00 00 | 03 00       | 60 81       | 00 00 10 00 | 00 00 00 00 |             |
| 00 10 00 00 | 00 00 00 00 |             | 00 00 10 00 | 00 00 00 00 |             |
| 00 10 00 00 | 00 00 00 00 |             | 00 00 00 00 | 10 00 00 00 |             |

Da mesma forma que a imagem anterior, cada retângulo da imagem representa um campo na tela, começando pelo ImageBase em preto e assim por diante.

| <b>Deslocamento<br/>(PE32/PE32+)</b> | <b>Tamanho<br/>(PE32/PE32+)</b> | <b>Campo</b>     | <b>Descrição</b>                                                                                                                                                                                                                                                                                                    |
|--------------------------------------|---------------------------------|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 28/24                                | 4/8                             | ImageBase        | O endereço preferido do primeiro byte de imagem quando carregado na memória; deve ser um múltiplo de 64 K. O padrão para DLLs é 0x10000000. O padrão para EXEs do Windows CE é 0x00010000. O padrão para Windows NT, Windows 2000, Windows XP, Windows 95, Windows 98 e Windows ME é 0x00400000.                    |
| 32/32                                | 4                               | SectionAlignment | O alinhamento (em bytes) das seções quando elas são carregadas na memória. Ele deve ser maior ou igual a FileAlignment. O padrão é o tamanho da página para a arquitetura.                                                                                                                                          |
| 36/36                                | 4                               | FileAlignment    | O fator de alinhamento (em bytes) usado para alinhar os dados brutos das seções no arquivo de imagem. O valor deve ser uma potência de 2 entre 512 e 64 K, inclusive. O padrão é 512. Se SectionAlignment for menor que o tamanho da página da arquitetura, o FileAlignment deverá corresponder a SectionAlignment. |



| <b>Deslocamento<br/>(PE32/PE32+)</b> | <b>Tamanho<br/>(PE32/PE32+)</b> | <b>Campo</b>                | <b>Descrição</b>                                                                                                                               |
|--------------------------------------|---------------------------------|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| 40/40                                | 2                               | MajorOperatingSystemVersion | O número de versão principal do sistema operacional necessário.                                                                                |
| 42/42                                | 2                               | MinorOperatingSystemVersion | O número de versão secundária do sistema operacional necessário.                                                                               |
| 44/44                                | 2                               | MajorImageVersion           | O número de versão principal da imagem.                                                                                                        |
| 46/46                                | 2                               | MinorImageVersion           | O número de versão secundária da imagem.                                                                                                       |
| 48/48                                | 2                               | MajorSubsystemVersion       | O número de versão principal do subsistema.                                                                                                    |
| 50/50                                | 2                               | MinorSubsystemVersion       | O número de secundária principal do subsistema.                                                                                                |
| 52/52                                | 4                               | Win32VersionValue           | Reservado; deve ser zero                                                                                                                       |
| 56/56                                | 4                               | SizeOfImage                 | O tamanho (em bytes) da imagem, incluindo todos os cabeçalhos, pois a imagem é carregada na memória. Deve ser um múltiplo de SectionAlignment. |
| 60/60                                | 4                               | SizeOfHeaders               | O tamanho combinado de um stub do MS-DOS, o cabeçalho do PE e os cabeçalhos de seção arredondados para um múltiplo de FileAlignment.           |
| 64/64                                | 4                               | Checksum                    | A soma de verificação do arquivo de imagem. O algoritmo para calcular a soma de verificação é incorporado a                                    |

| Deslocamento<br>(PE32/PE32+) | Tamanho<br>(PE32/PE32+) | Campo              | Descrição                                                                                                                                                                                                                           |
|------------------------------|-------------------------|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                              |                         |                    | IMAGHELP.DLL.<br>Estes são os elementos verificados quanto à validação no momento do carregamento: todos os drivers, qualquer DLL carregada no momento da inicialização e qualquer DLL carregada em um processo crítico do Windows. |
| 68/68                        | 2                       | Subsistema         | O subsistema necessário para executar esta imagem. Para obter mais informações, confira <a href="#">Subsistema do Windows</a> .                                                                                                     |
| 70/70                        | 2                       | DllCharacteristics | Para obter mais informações, confira <a href="#">Características de DLL</a> posteriormente nesta especificação.                                                                                                                     |
| 72/72                        | 4/8                     | SizeOfStackReserve | O tamanho da pilha a ser reservada. Somente SizeOfStackCommit é confirmado; o restante é disponibilizado uma página por vez até que o tamanho da reserva seja atingido.                                                             |
| 76/80                        | 4/8                     | SizeOfStackCommit  | O tamanho da pilha a ser confirmada.                                                                                                                                                                                                |
| 80/88                        | 4/8                     | SizeOfHeapReserve  | O tamanho do espaço de heap local a ser reservado. Somente SizeOfHeapCommit é confirmado; o restante é                                                                                                                              |

| <b>Deslocamento<br/>(PE32/PE32+)</b> | <b>Tamanho<br/>(PE32/PE32+)</b> | <b>Campo</b>        | <b>Descrição</b>                                                                                                              |
|--------------------------------------|---------------------------------|---------------------|-------------------------------------------------------------------------------------------------------------------------------|
|                                      |                                 |                     | disponibilizado uma página por vez até que o tamanho da reserva seja atingido.                                                |
| 84/96                                | 4/8                             | SizeOfHeapCommit    | O tamanho do espaço de heap local a ser confirmado.                                                                           |
| 88/104                               | 4                               | LoaderFlags         | Reservado; deve ser zero                                                                                                      |
| 92/108                               | 4                               | NumberOfRvaAndSizes | O número de entradas de diretório de dados no restante do cabeçalho opcional. Cada uma descreve uma localização e um tamanho. |