



# UPX Packer

이지훈

# 목차

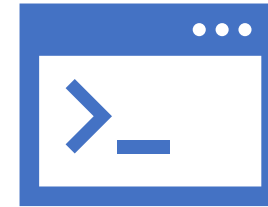


## Packer

압축이란?

Packer

PE Protector



## Notepad\_upx.exe

UPX Packer

Notepad.exe

Notepad\_upx.exe



# 압축이란?

파일, 데이터의 크기를 줄여서 보관 및 이동에 용이하도록 만드는 행위.

어떤 형태의 파일이라도 바이너리 형식으로 이루어져 있다면 적절한 압축 알고리즘을 사용하여 압축할 수 있다.





# 압축이란?

**비손실 압축** : 우리가 흔히 알고 있는 모습의 압축. 압축을 해제하는데 있어서 데이터의 무결성이 보장되어야 한다. (Ex : ZIP, RAR ...)

**손실 압축** : 파일에 의도적인 손상을 주어서 그 대가로 압축률을 높이는 목적으로 사용한다. 주로 멀티미디어 파일에 사용한다. (Ex : JPG, MP3, MP4)

## • 무손실 압축



## • 손실 압축





# 압축이란?

**실행 압축** : 실행파일(PE파일)을 대상으로 파일 내부에 압축해제 코드를 포함하여  
파일이 실행되는 순간에 메모리에서 압축을 해제 후 실행하는 기술.

항목	일반 압축	실행 압축
대상파일	모든 파일	PE 파일(exe, dll, sys)
압축 결과물	압축(zip, rar) 파일	PE 파일(exe, dll, sys)
압축해제 방식	전용 압축해제 프로그램 사용	내부의 decoding 루틴
파일 실행 여부	자체 실행 불가	자체 실행 가능
장점	모든 파일에 대해 높은 압축율로 압축 가능	별도의 해제 프로그램 없이 바로 실행 가능
단점	전용 압축해제 프로그램이 없으면 해당 압축 파일을 사용할 수 없다.	실행할 때마다 decoding 루틴이 호출되기 때문에 실행시간이 아주 미세하게 느려진다.

표 14.1 일반 압축과 실행 압축의 비교





# Packer

PE 파일을 압축하는 유틸리티. PE Packer, Run-Time Packer 라고도 부른다.

순수한 의도의 패커 : UPX, ASPack 등

불순한 의도의 패커 : Upack, PESpin, NSAnti 등

-> Virus, Trojan, Worm 등에서 사용됨





# PE Protector

PE 파일을 리버싱으로부터 보호하기 위한 유틸리티.

Anti Debugging, Anti Emulating, Garbage Code 등의 여러 기법이 사용된다.

Packer 와는 다르게 원본 PE 파일보다 파일의 크기가 커진다.

악성 코드에서도 많이 사용된다. (Anti Virus 제품의 진단을 막거나 늦추기 위함)



# UPX Packer

```
명령 프롬프트
C:\Jihoon\2024\SCP\0402\upx-4.2.3-win32\upx-4.2.3-win32>upx.exe
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2024
UPX 4.2.3      Markus Oberhumer, Laszlo Molnar & John Reiser   Mar 27th 2024

Usage: upx [-123456789dlthVL] [-qvfk] [-o file] file..

Commands:
  -1      compress faster              -9      compress better
  -d      decompress                  -l      list compressed file
  -t      test compressed file        -V      display version number
  -h      give more help              -L      display software license

Options:
  -q      be quiet                    -v      be verbose
  -oFILE  write output to 'FILE'
  -f      force compression of suspicious files
  -k      keep backup files
file..   executables to (de)compress

Type 'upx --help' for more detailed help.

UPX comes with ABSOLUTELY NO WARRANTY; for details visit https://upx.github.io

C:\Jihoon\2024\SCP\0402\upx-4.2.3-win32\upx-4.2.3-win32>
```







# UPX Packer

실행 파일의 크기가 줄어든 것을 확인할 수 있다.  
(1286656 -> 504832)

Notepad.exe	2024-03-16 오전 ...	응용 프로그램	1,257KB
notepad_upx.exe	2024-03-16 오전 ...	응용 프로그램	493KB

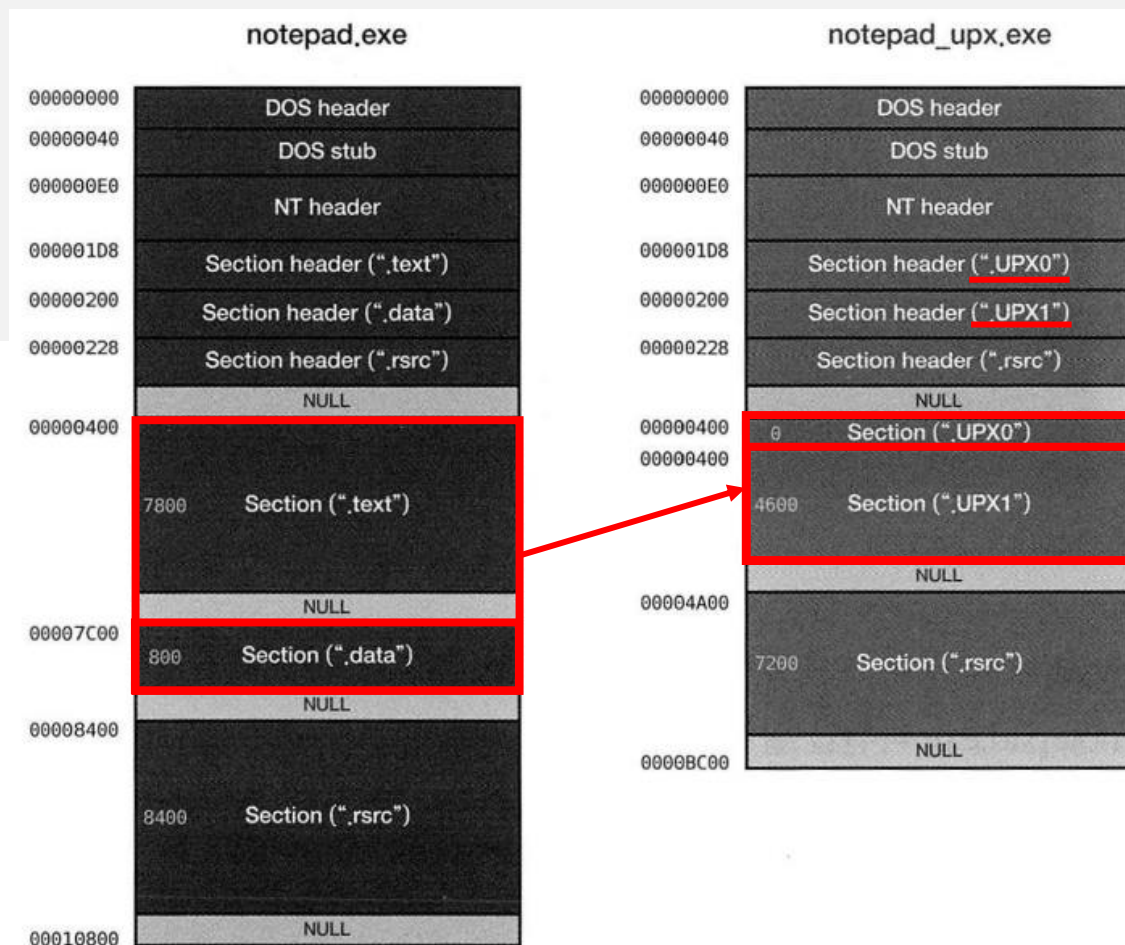
```
명령 프롬프트
C:\Jihoon\2024\SCP\0402\upx-4.2.3-win32\upx-4.2.3-win32>upx.exe -f -o notepad_upx.exe Notepad.exe
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2024
UPX 4.2.3      Markus Oberhumer, Laszlo Molnar & John Reiser   Mar 27th 2024

  File size      Ratio      Format      Name
  -----      -
1286656 -> 504832  39.24%    win64/pe    notepad_upx.exe

Packed 1 file.
```



# UPX Packer



섹션 헤더와 섹션의 이름이 바뀜

그림 14.3 notepad.exe와 notepad\_upx.exe의 비교



# UPX Packer

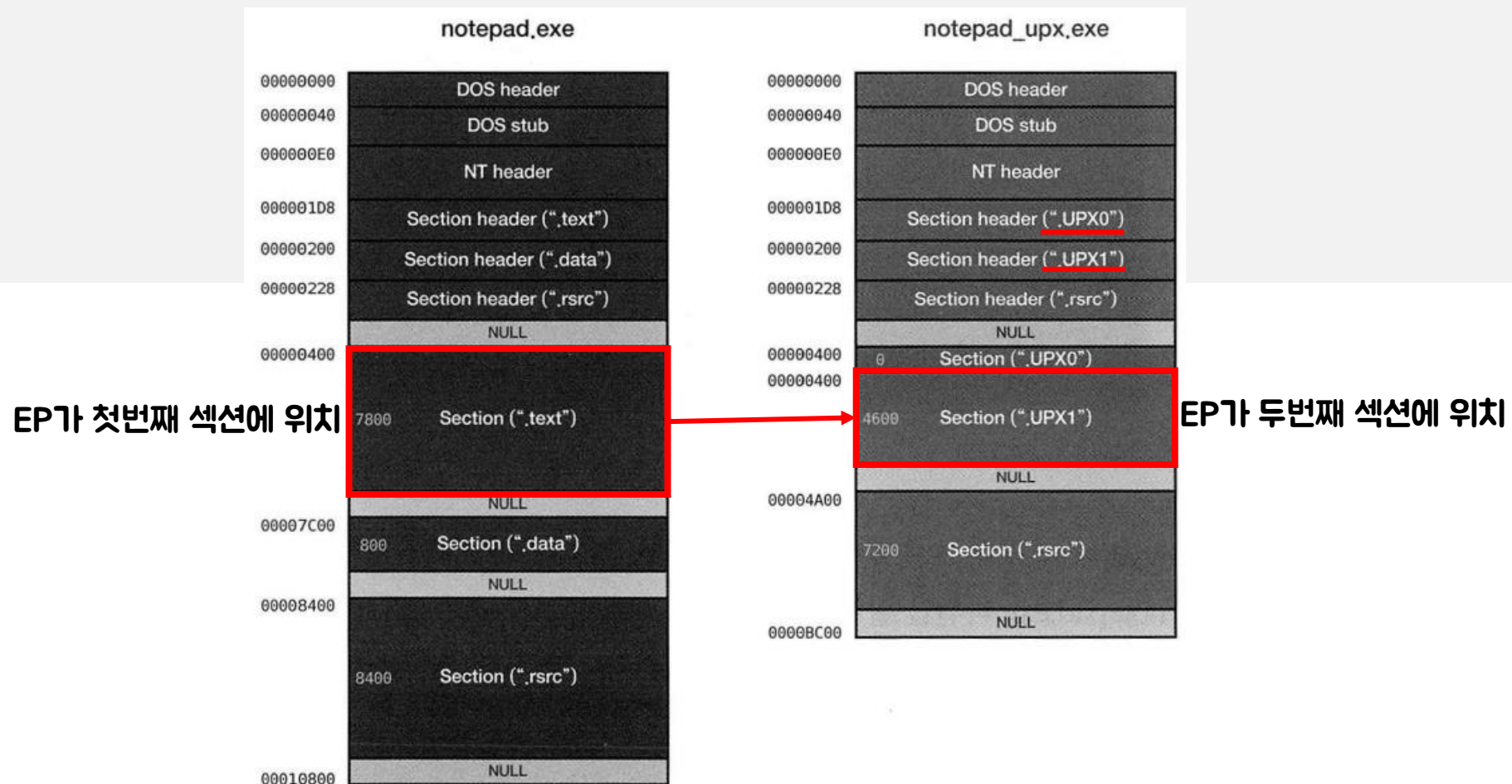


그림 14.3 notepad.exe와 notepad\_upx.exe의 비교



# Notepad\_upx.exe

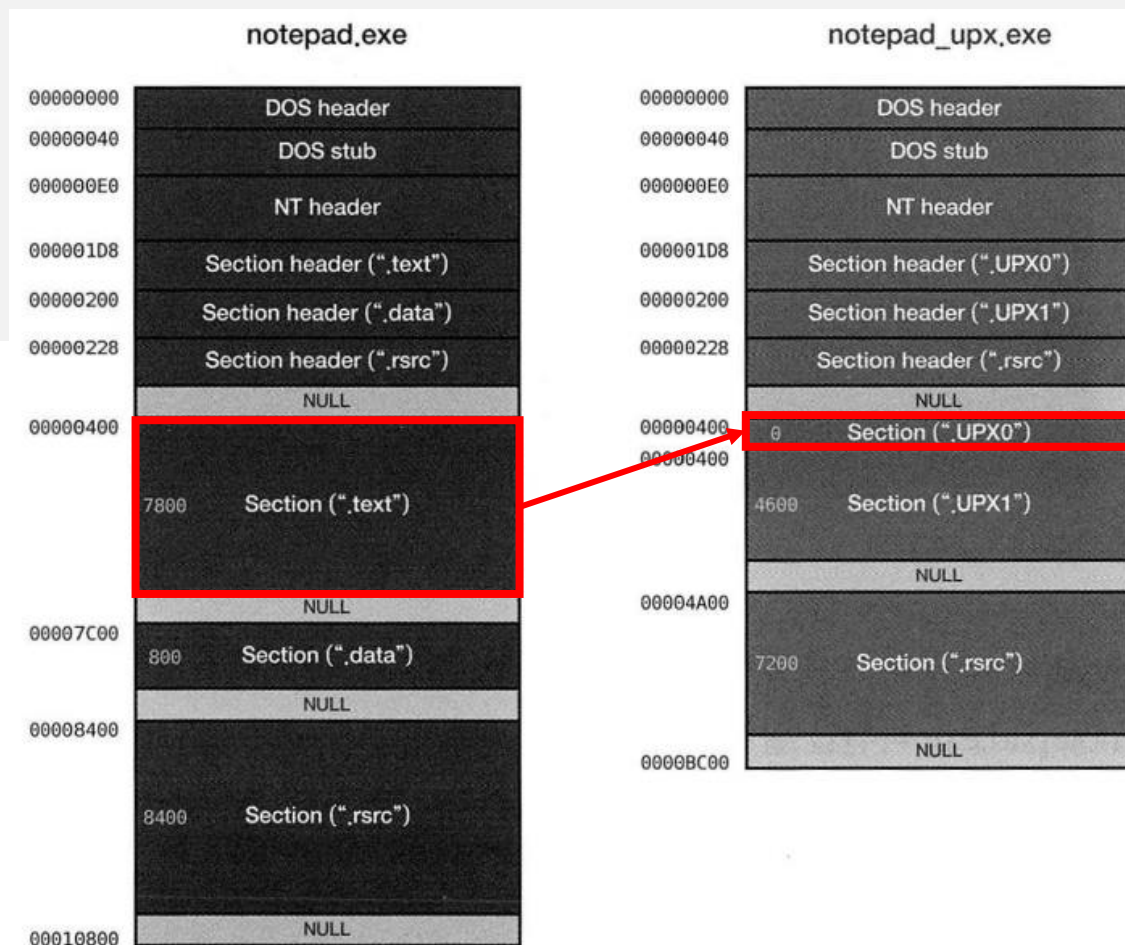
32upx_notepad.exe
IMAGE_DOS_HEADER
MS-DOS Stub Program
IMAGE_NT_HEADERS
Signature
IMAGE_FILE_HEADER
<b>IMAGE_OPTIONAL_HEADER</b>
IMAGE_SECTION_HEADER UPX0
IMAGE_SECTION_HEADER UPX1
IMAGE_SECTION_HEADER .rsrc
SECTION UPX0
+ SECTION UPX1
+ SECTION .rsrc

pFile	Data	Description
000000F8	010B	Magic
000000FA	07	Major Linker Version
000000FB	0A	Minor Linker Version
000000FC	00005000	Size of Code
00000100	00008000	Size of Initialized Data
00000104	00010000	Size of Uninitialized Data
00000108	<u>00015330</u>	Address of Entry Point
0000010C	<u>00011000</u>	Base of Code
00000110	00016000	Base of Data
00000114	01000000	Image Base
00000118	00001000	Section Alignment
0000011C	00000200	File Alignment
00000120	0005	Major O/S Version
00000122	0001	Minor O/S Version
00000124	0005	Major Image Version
00000126	0001	Minor Image Version





# UPX Packer



첫번째 섹션의 크기가 0

그림 14.3 notepad.exe와 notepad\_upx.exe의 비교



# Notepad\_upx.exe

32upx_notepad.exe	pFile	Data	Description
IMAGE_DOS_HEADER	000001D8	55 50 58 30	Name
MS-DOS Stub Program	000001DC	00 00 00 00	
IMAGE_NT_HEADERS	000001E0	<u>00010000</u>	Virtual Size
Signature	000001E4	00001000	RVA
IMAGE_FILE_HEADER	000001E8	<u>00000000</u>	Size of Raw Data
IMAGE_OPTIONAL_HEADER	000001EC	00000400	Pointer to Raw Data
<b>IMAGE_SECTION_HEADER UPX0</b>	000001F0	00000000	Pointer to Relocations
IMAGE_SECTION_HEADER UPX1	000001F4	00000000	Pointer to Line Numbers
IMAGE_SECTION_HEADER .rsrc	000001F8	0000	Number of Relocations
SECTION UPX0	000001FA	0000	Number of Line Numbers
+ SECTION UPX1	000001FC	E0000080	Characteristics
+ SECTION .rsrc			00000080 20000000 40000000 80000000







# UPX Packer



.rsrc 섹션은 크게 변하지 않았음

그림 14.3 notepad.exe와 notepad\_upx.exe의 비교



# Notepad.exe

## GetModuleHandleA() API 호출

## MZ, PE 시그니처 비교

```
01007390 6A 70          PUSH 70
0100739F 68 98180001   PUSH 01001898
010073A4 E8 BF010000   CALL 01007568
010073A9 33DB          XOR EBX,EBX
010073AB 53            PUSH EBX
010073AC 8B3D CC100000 MOV EDI,DWORD PTR DS:[<&KERNEL32.GetModu
010073B2 FFD7          CALL EDI
010073B4 66:8138 4D5A  CMP WORD PTR DS:[EAX],5A4D
010073B9 75 1F          JNE SHORT 010073DA
010073BB 8B48 3C        MOV ECX,DWORD PTR DS:[EAX+3C]
010073BE 03C8          ADD ECX,EAX
010073C0 8139 50450000 CMP DWORD PTR DS:[ECX],4550
010073C6 75 12          JNE SHORT 010073DA
010073C8 0FB741 18      MOVZX EAX,WORD PTR DS:[ECX+18]
010073CC 3D 0B010000    CMP EAX,10B
010073D1 74 1F          JE SHORT 010073F2
010073D3 3D 0B020000    CMP EAX,20B
010073D8 74 05          JE SHORT 010073DF
010073DA 895D E4        MOV DWORD PTR SS:[EBP-1C],EBX
010073DD EB 27          JMP SHORT 01007406
```

Module Name => NULL  
KERNEL32.GetModuleHandleA

OllyDbg - Entry Point

32notepad.exe		pFile	Data	Description
IMAGE_DOS_HEADER		00000000	5A4D	Signature
MS-DOS Stub Program		00000002	0000	Bytes on Last Page of File
IMAGE_NT_HEADERS		00000004	0003	Pages in File
Signature		00000006	0000	Relocations
IMAGE_FILE_HEADER		00000008	0004	Size of Header in Paragraphs
IMAGE_OPTIONAL_HEADER		0000000A	0000	Minimum Extra Paragraphs
IMAGE_SECTION_HEADER .text		0000000C	FFFF	Maximum Extra Paragraphs

32notepad.exe		pFile	Data	Description	Value
IMAGE_DOS_HEADER		000000E0	00004550	Signature	IMAGE_NT_SIGNATURE PE

PE Viewer - Signature







# Notepad\_upx.exe(EP)

PUSHAD 로 스택에 레지스터 값 저장

ESI에 두번째 섹션 시작주소(1011000), EDI 에 첫번째 섹션 시작주소(1001000) 세팅

01015330	60	pushad
01015331	BE 00100101	mov esi,32upx_notepad.1011000
01015336	8DBE 0000FFFF	lea edi,dword ptr ds:[esi-10000]
0101533C	57	push edi
0101533D	83CD FF	or ebp,FFFFFFFF
01015340	EB 10	jmp 32upx_notepad.1015352

x32dbg - Entry Point





# Notepad\_upx.exe(Loop1)

1번째 Loop 구간. edx의 값을 edi에 쓰고 있다.

크게 변화되는 부분이 없으니 Loop 다음 주소에 BP 후 Loop 탈출.

010153D8	83FD FC	cmp ebp,FFFFFFFF
010153DB	76 0F	jbe 32upx_notepad.010153EC
010153DD	8A02	mov al,byte ptr ds:[edx]
010153DF	42	inc edx
010153E0	8807	mov byte ptr ds:[edi],al
010153E2	47	inc edi
010153E3	49	dec ecx
010153E4	75 F7	jne 32upx_notepad.010153DD
010153E6	E9 63FFFFFF	jmp 32upx_notepad.0101534E

EAX	FFFFFFF0	
EBX	DB800000	
ECX	00000364	Le
EDX	01001007	32upx_notepad.01001007
EBP	FFFFFFFF	
ESP	000DFF54	
ESI	01011005	32upx_notepad.01011005
EDI	01001008	32upx_notepad.01001008

x32dbg - Loop1





# Notepad\_upx.exe(Loop2)

2번째 Loop 구간. Loop1 을 포함한 더 큰 구간에서 반복하고 있다.

특정 연산들을 통해 압축된 코드를 해제하여 edi(첫번째 섹션을 가리키고 있음)에 값을 넣고 있다.

```
010153D5 8D142F lea edx,0
010153D8 83FD FC cmp ebp,0
010153DB 76 0F jbe 32upx_notepad.10153DD
010153DD 8A02 mov al,byte ptr ds:[esi]
010153DF 42 inc edx
010153E0 8807 mov byte ptr ds:[edi],al
010153E2 47 inc edi
010153E3 49 dec ecx
010153E4 75 F7 jne 32upx_notepad.10153E6
010153E6 E9 63FFFFFF jmp 32upx_notepad.1015390
010153EB 90 nop
010153EC 8B02 mov eax,dword ptr ds:[edx]
010153EE 83C2 04 add edx,4
010153F1 8907 mov dword ptr ds:[edi],eax
010153F3 83C7 04 add edi,4
010153F6 83E9 04 sub ecx,4
010153F9 77 F1 ja 32upx_notepad.10153EC
010153FB 01CF add edi,ecx
010153FD E9 4CFFFFFF jmp 32upx_notepad.101534E
01015402 7E pop esi
01015381 72 0D jb 32upx_notepad.1015390
01015383 C1E0 08 shl eax,8
01015386 8A06 mov al,byte ptr ds:[esi]
01015388 46 inc esi
01015389 83F0 FF xor eax,FFFFFFFF
0101538C 74 74 je 32upx_notepad.1015402
0101538E 89C5 mov ebp,eax
01015390 01DB add ebx,ebx
01015392 75 07 jne 32upx_notepad.101539B
01015394 8B1E mov ebx,dword ptr ds:[esi]
01015396 83EE FC sub esi,FFFFFFFC
01015399 11DB adc ebx,ebx
0101539B 11CB adc esi,esi
```

x32dbg - Loop2





# Notepad\_upx.exe(Loop2)

2번째 Loop 구간. Loop1 을 포함한 더 큰 구간에서 반복하고 있다.

특정 연산들을 통해 압축된 코드를 해제하여 edi(첫번째 섹션을 가리키고 있음)에 값을 넣고 있다.

주소	Hex	ASCII
01001420	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001430	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001440	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001450	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001460	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001470	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001480	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001490	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001500	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001510	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001520	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001530	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001540	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001550	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....

주소	Hex	ASCII
01004220	56 E8 01 00 2F 95 57 B8 80 00 00 00 53 6A 03 57	Vè../.W»....Sj.W
01004230	8B 3D 04 11 00 01 6A 03 68 00 00 00 80 56 FF D7	.=....j.h....Vyx
01004240	83 F8 FF A3 80 A4 00 01 0F 85 84 00 00 00 FF 15	.øÿf.»......ÿ.
01004250	38 11 00 01 48 48 74 39 83 E8 03 6A 31 56 74 29	8...Hht9.e.j1vt)
01004260	83 E8 76 74 1C FF 35 34 90 00 01 FF 35 54 90 00	.èvt.ÿ54...ÿST..
01004270	01 FF 35 30 98 00 01 E8 01 00 0F 6C 89 45 FC EB	.ÿ50...è...l.Eüè
01004280	48 FF 35 64 90 00 01 EB E2 FF 35 9C 90 00 01 EB	Hÿ5d...ëäÿ5....ë
01004290	DA 6A 33 56 FF 35 38 90 00 01 FF 35 54 90 00 01	Új3vÿ58...ÿST...
010042A0	FF 35 30 98 00 01 E8 01 00 0F 6C 83 F8 06 89 45	ÿ50...è...l.ø..E
010042B0	FC 75 16 6A 00 53 6A 04 6A 00 6A 03 68 00 00 00	üu.j.Sj.j.j.h...
010042C0	C0 56 FF D7 A3 80 A4 00 01 83 3D 80 A4 00 01 FF	ÄVÿxf.».==.».ÿ
010042D0	74 19 FF 35 80 9A 00 01 56 E8 01 00 41 75 6A 02	t.ÿ5....Vè..Au.j.
010042E0	58 39 45 FC 74 07 6A 06 58 EB 02 33 C0 5F 5E 58	x9Eüt.j.Xè.3A.^[
010042F0	C9 C2 04 00 CC CC CC CC CC 8B FF 55 8B EC 51 56	ÉÄ..iiii.ÿu.ïqv
01004300	8B 75 08 57 8B C6 B9 74 17 00 01 C7 45 FC 01 00	.u.w.Æ't...ÇEü..
01004310	00 00 E8 01 00 2F 48 33 FF 85 C0 75 0E 83 C6 06	..è../H3ÿ.Au..Æ.
01004320	56 E8 01 00 31 76 89 7D FC EB 1D 8B C6 B9 6C 17	Vè..1v.ÿüè..Æ'l.
01004330	00 01 E8 01 00 2F 48 85 C0 0F 85 0C 01 00 00 83	..è../H.A.....
01004340	C6 04 56 E8 01 00 31 76 8B F0 66 39 3E 0F 84 F8	Æ.Vè..1v.ðf9>...ø
01004350	00 00 00 FF 75 0C FF 35 30 98 00 01 FF 15 B0 11	...ÿu.ÿ50...ÿ.°

x32dbg - Hex Dump (.upx0)





# Notepad\_upx.exe(Loop2)

2번째 Loop 구간. Loop1 을 포함한 더 큰 구간에서 반복하고 있다.

특정 연산들을 통해 압축된 코드를 해제하여 edi(첫번째 섹션을 가리키고 있음)에 값을 넣고 있다.

주소	Hex	ASCII
01001420	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001430	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001440	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001450	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001460	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001470	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001480	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001490	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
010014F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001500	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001510	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001520	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001530	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001540	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
01001550	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....



주소	Hex	ASCII
01004214	0F84 CC000000	je 32upx_notepad.10042E6
0100421A	50	push eax
0100421B	BE 00A90001	mov esi,32upx_notepad.100A900
01004220	56	push esi
01004221	E8 01002F95	call 962F4227
01004226	57	push edi
01004227	BB 80000000	mov ebx,80
0100422C	53	push ebx
0100422D	6A 03	push 3
0100422F	57	push edi
01004230	8B3D 04110001	mov edi,dword ptr ds:[1001104]
01004236	6A 03	push 3
01004238	68 00000080	push 80000000
0100423D	56	push esi
0100423E	FFD7	call edi
01004240	83F8 FF	cmp eax,FFFFFFFF
01004243	A3 80A40001	mov dword ptr ds:[100A480],eax

x32dbg - Hex Dump (.upx0)





# Notepad\_upx.exe(Loop3)

3번째 Loop 구간.

원본 코드의 Call/Jmp 명령어의 도착지 주소를 복원시켜주는 코드. 루프가 끝난 뒤의 주소에 BP 설치 후 탈출.

0101540A	8A07	mov al,byte ptr ds:[edi]
0101540C	47	inc edi
0101540D	2C E8	sub al,E8
0101540F	3C 01	cmp al,1
01015411	77 F7	ja 32upx_notepad.101540A
01015413	803F 01	cmp byte ptr ds:[edi],1
01015416	75 F2	jne 32upx_notepad.101540A
01015418	8B07	mov eax,dword ptr ds:[edi]
0101541A	8A5F 04	mov bl,byte ptr ds:[edi+4]
0101541D	66:C1E8 08	shr ax,8
01015421	C1C0 10	rol eax,10
01015424	86C4	xchg ah,al
01015426	29F8	sub eax,edi
01015428	80EB E8	sub bl,E8
0101542B	01F0	add eax,esi
0101542D	8907	mov dword ptr ds:[edi],eax
0101542F	83C7 05	add edi,5
01015432	88D8	mov al,bl
01015434	E2 D9	loop 32upx_notepad.101540F
01015436	8DBE 00300100	lea edi,dword ptr ds:[esi+13000]

x32dbg - Loop3



# Notepad\_upx.exe(Loop3)

3번째 Loop 구간.

원본 코드의 Call/Jmp 명령어의 도착지 주소를 복원시켜주는 코드. 루프가 끝난 뒤의 주소에 BP 설치 후 탈출.

01004214	✓ 0F84 CC000000	je 32upx_notepad.10042E6
0100421A	50	push eax
0100421B	BE 00A90001	mov esi,32upx_notepad.100A900
01004220	56	push esi
01004221	E8 01002F95	<u>call 962F4227</u>
01004226	57	push edi
01004227	BB 80000000	mov ebx,80
0100422C	53	push ebx
0100422D	6A 03	push 3
0100422F	57	push edi
01004230	8B3D 04110001	mov edi,dword ptr ds:[1001104]
01004236	6A 03	push 3
01004238	68 00000080	push 80000000
0100423D	56	push esi
0100423E	FFD7	call edi
01004240	83F8 FF	cmp eax,FFFFFFFF
01004243	A3 80A40001	mov dword ptr ds:[100A480],eax

01004214	✓ 0F84 CC000000	je 32upx_notepad.10042E6
0100421A	50	push eax
0100421B	BE 00A90001	mov esi,32upx_notepad.100A900
01004220	56	push esi
01004221	E8 73FDFFFF	<u>call 32upx_notepad.1003F99</u>
01004226	57	push edi
01004227	BB 80000000	mov ebx,80
0100422C	53	push ebx
0100422D	6A 03	push 3
0100422F	57	push edi
01004230	8B3D 04110001	mov edi,dword ptr ds:[1001104]
01004236	6A 03	push 3
01004238	68 00000080	push 80000000
0100423D	56	push esi
0100423E	FFD7	call edi
01004240	83F8 FF	cmp eax,FFFFFFFF
01004243	A3 80A40001	mov dword ptr ds:[100A480],eax

x32dbg - Hex Dump (.upx0)





# Notepad\_upx.exe(Loop4)

4번째 Loop 구간.

압축할 때 해당 프로그램에 사용되는 API를 분석해 두번째 섹션에 저장해두고,  
다시 원본 파일의 IAT(Import Address Table)를 세팅하는 Loop.

0101543C	8B07	mov eax,dword ptr ds:[edi]
0101543E	09C0	or eax,eax
01015440	74 3C	je 32upx_notepad.101547E
01015442	8B5F 04	mov ebx,dword ptr ds:[edi+4]
01015445	8D8430 04BE0100	lea eax,dword ptr ds:[eax+esi+1BE04]
0101544C	01F3	add ebx,esi
0101544E	50	push eax
0101544F	83C7 08	add edi,8
01015452	FF96 CCBE0100	call dword ptr ds:[esi+1BECC]
01015458	95	xchg ebp,eax
01015459	8A07	mov al,byte ptr ds:[edi]
0101545B	47	inc edi
0101545C	08C0	or al,al
0101545E	74 DC	je 32upx_notepad.101543C
01015460	89F9	mov ecx,edi
01015462	57	push edi
01015463	48	dec eax
01015464	F2:AE	repne scasb
01015466	55	push ebp
01015467	FF96 D0BE0100	call dword ptr ds:[esi+1BED0]
0101546D	09C0	or eax,eax
0101546F	74 07	je 32upx_notepad.1015478
01015471	8903	mov dword ptr ds:[ebx],eax
01015473	83C3 04	add ebx,4
01015476	EB E1	jmp 32upx_notepad.1015459
01015478	FF96 E0BE0100	call dword ptr ds:[esi+1BEE0]
0101547E	8BAE D4BE0100	mov ebp,dword ptr ds:[esi+1BED4]

x32dbg - Loop4





# Notepad\_upx.exe(Loop4)

4번째 Loop 구간.

압축할 때 해당 프로그램에 사용되는 API를 분석해 두번째 섹션에 저장해두고,  
다시 원본 파일의 IAT(Import Adress Table)를 세팅하는 Loop.

01004214	0F84 CC000000	je 32upx_notepad.10042E6
0100421A	50	push eax
0100421B	BE 00A90001	mov esi,32upx_notepad.100A900
01004220	56	push esi
01004221	E8 73FDFFFF	call 32upx_notepad.1003F99
01004226	57	push edi
01004227	BB 80000000	mov ebx,80
0100422C	53	push ebx
0100422D	6A 03	push 3
0100422F	57	push edi
01004230	8B3D 04110001	mov edi,dword ptr ds:[1001104]
01004236	6A 03	push 3
01004238	68 00000080	push 80000000
0100423D	56	push esi
0100423E	FFD7	call edi
01004240	83F8 FF	cmp eax,FFFFFFFF
01004243	A3 80A40001	mov dword ptr ds:[100A480],eax

01004214	0F84 CC000000	je 32upx_notepad.10042E6
0100421A	50	push eax
0100421B	BE 00A90001	mov esi,32upx_notepad.100A900
01004220	56	push esi
01004221	E8 73FDFFFF	call 32upx_notepad.1003F99
01004226	57	push edi
01004227	BB 80000000	mov ebx,80
0100422C	53	push ebx
0100422D	6A 03	push 3
0100422F	57	push edi
01004230	8B3D 04110001	mov edi,dword ptr ds:[<&CreateFileW>]
01004236	6A 03	push 3
01004238	68 00000080	push 80000000
0100423D	56	push esi
0100423E	FFD7	call edi
01004240	83F8 FF	cmp eax,FFFFFFFF
01004243	A3 80A40001	mov dword ptr ds:[100A480],eax

x32dbg - Hex Dump (.upx0)





# Notepad\_upx.exe(OEP)

EP : pushad <---> popad .... OEP!!!

압축 해제 후 본래 갖고 있던 레지스터 값을 다시 가져온 뒤 OEP(Original Entry Point)로 JMP

01015330	<	60	pushad
01015331		BE 00100101	mov esi,32upx_notepad.1011000
01015336		8DBE 0000FFFF	lea edi,dword ptr ds:[esi-10000]
0101533C		57	push edi
0101533D		83CD FF	or ebp,FFFFFFFF
01015340	✓	EB 10	jmp 32upx_notepad.1015352

Loop1..2...3..4...

010154AD		61	popad
010154AE		8D4424 80	lea eax,dword ptr ss:[esp-80]
010154B2		6A 00	push 0
010154B4		39C4	cmp esp,eax
010154B6	^	75 FA	jne 32upx_notepad.10154B2
010154B8		83EC 80	sub esp,FFFFFF80
010154BB	-	E9 DD1EFFFF	jmp 32upx_notepad.100739D





# OEP에 빠르게 접근하는 법

## 1. POPAD 명령어 이후의 JMP 명령어에 BP 설치

- UPX 패커의 경우 압축 해제 과정이 PUSHAD로 시작해 POPAD로 끝난다.

## 2. 스택에 하드웨어 BP 설치

- 하드웨어 BP : CPU 에서 지원하는 BP로, BP가 설치된 주소 스택에 액세스 하게 되면 멈추게 된다. 이 또한 PUSHAD/POPAD 의 특성을 이용한 방법이다.





# Q & A