

# Reverse engineering

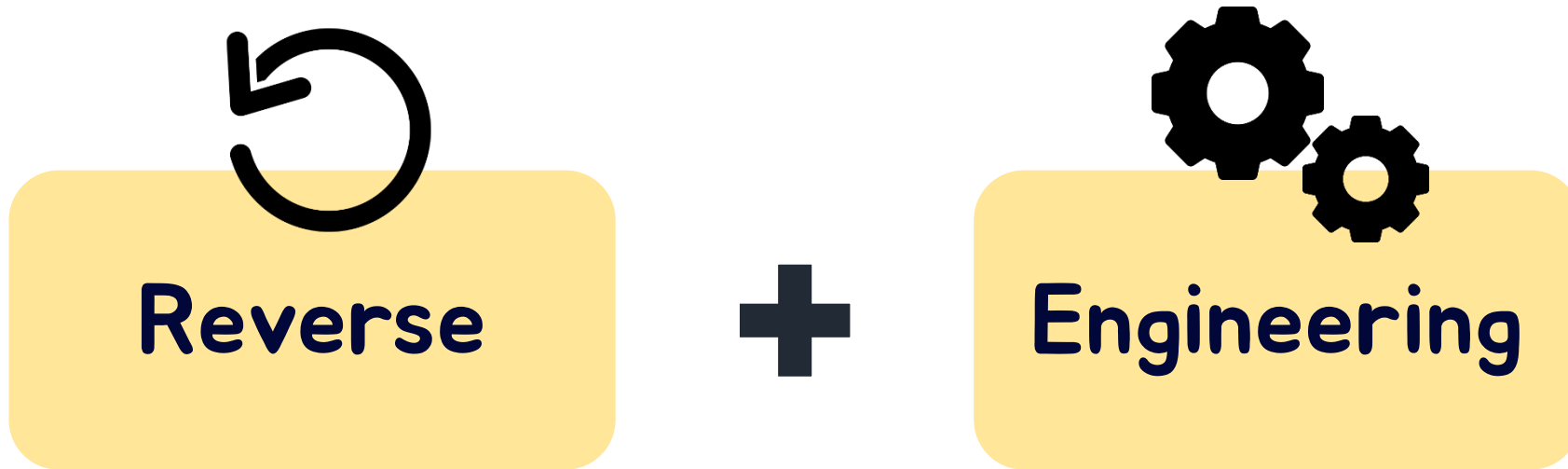
문승재

# Index

- 1. What is reversing ?**
- 2. How to reversing ?**
- 3. Prior knowledge**
- 4. Training**
- 5. QnA**

# What is reversing ?

# What is reversing ?



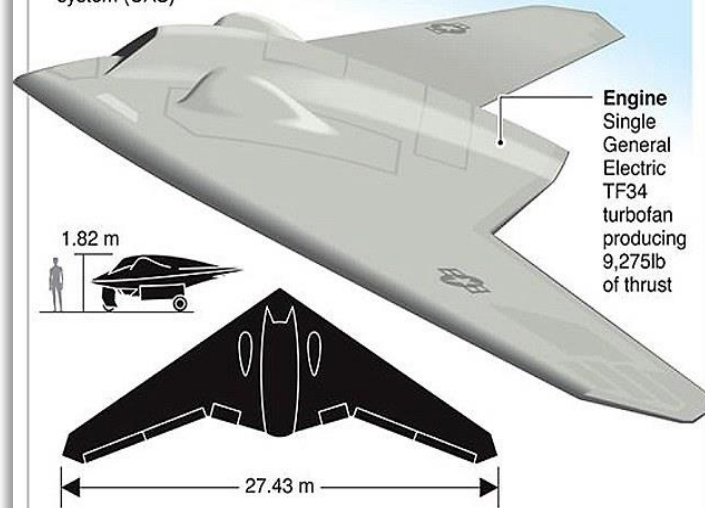
# What is reversing ?



## U.S. RQ-170 SENTINEL

Operated by the U.S. Air Force's Air Combat Command's 432nd Wing, the drone supports combatant needs for intelligence, reconnaissance and surveillance

|  |                      |            |                 |
|--|----------------------|------------|-----------------|
| ▶ Aircraft type                                | ▶ Operating altitude | ▶ Material | ▶ Manufacturer  |
| Low observable, unmanned aircraft system (UAS) | 15,240 m             | Composite  | Lockheed Martin |



Sources: af.mil, airforce-technology.com

REUTERS

## 이란 해군,美 드론 역설계 개발한 공격용 드론 '시모르그'공개

김익철 전문기자 | 승인 2019.12.08 18:26 | 댓글 0

- 작전 반경 1500km...최대 적재량 400KG



이란 해군이 7일 공개한 공격무인기 시모르그[사진=이란 국영방송 캡처=연합뉴스]

# What is reversing ?

```
import discord
import asyncio
import youtube_dl
import re, random

token = "NzAyMzUwMTgzZjMTUwMDA"
client = discord.Client()

que = {}
playlist = {}
playlist = List() #재생목록 리스트

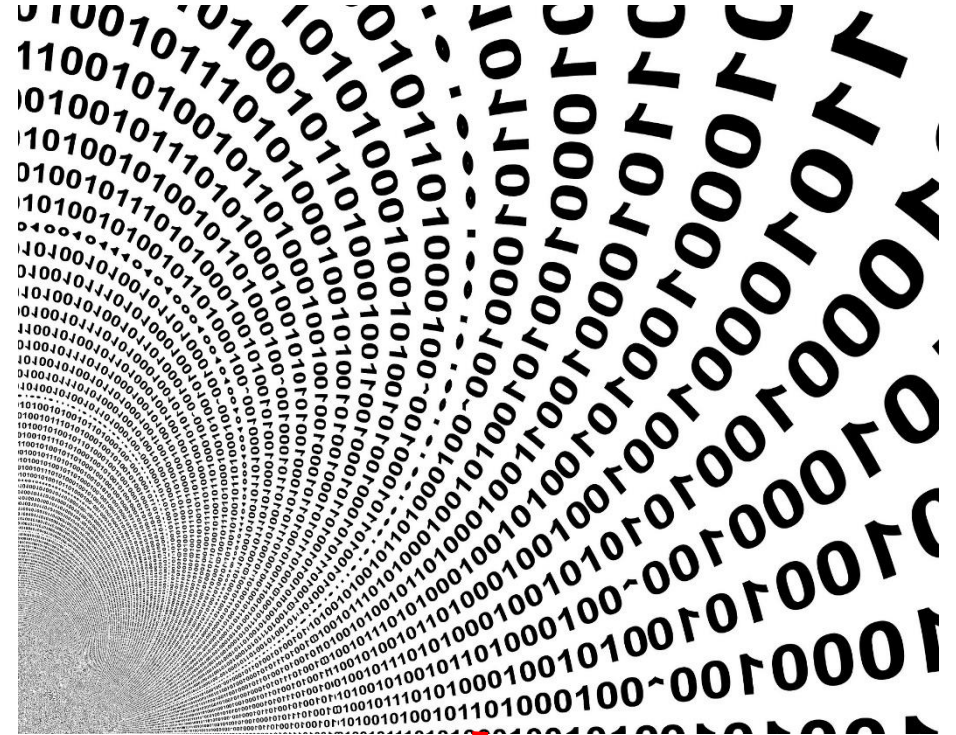
def queue(id): #음악 재생용 큐
    if que[id] != []:
        player = que[id].pop(0)
        playlist[id] = player
        del playlist[0]
        player.start()

@client.event
async def on_ready():
    await client.change_presence(status=discord.Status.online, activity=discord.Game("Hi Tony :D"))
    print("Hi Tony!")
    print(client.user.name)
    print(client.user.id)

@client.event
async def on_message(message):
    if message.author.bot:
        return None
    if message.content.startswith("!안녕"):
        embed = discord.Embed(title="안녕", description="안녕", url = "http://google.com", colour=0xDEADB8F)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="소재목", value="설명", inline=True)
        embed.add_field(name="안녕", value="[안녕](http://google.com)", inline=True)
        await message.channel.send(embed=embed)
        await message.channel.send("할 말", embed=embed)
```

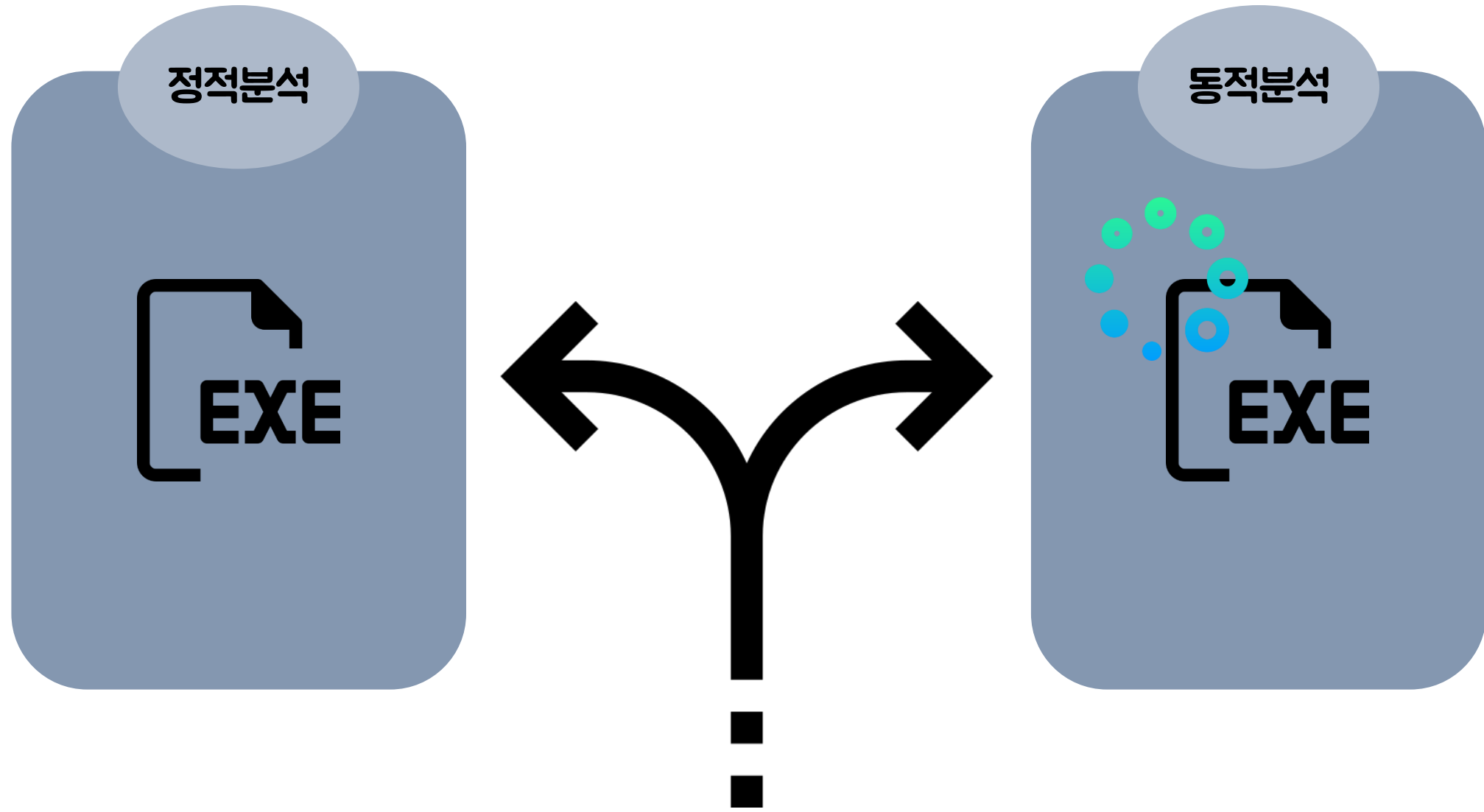


리버싱



# How to reversing ?

# How to reversing ?





# How to reversing ?

## 정적분석

파일의 종류, 크기 등  
내용 확인

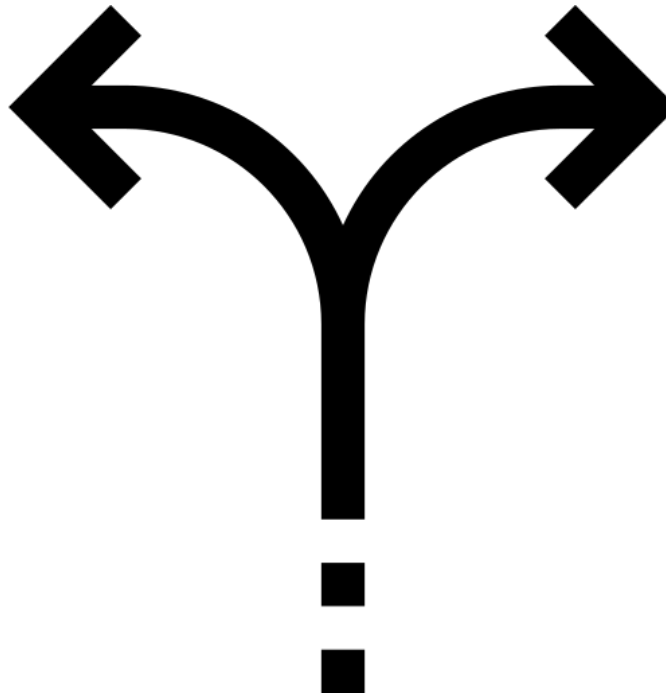
내부 코드와 구조 확인



## 동적분석

코드 흐름과 메모리  
상태 확인

프로그램의 동작원리  
분석



# Prior knowledge

# | Prior knowledge

CPU

Register

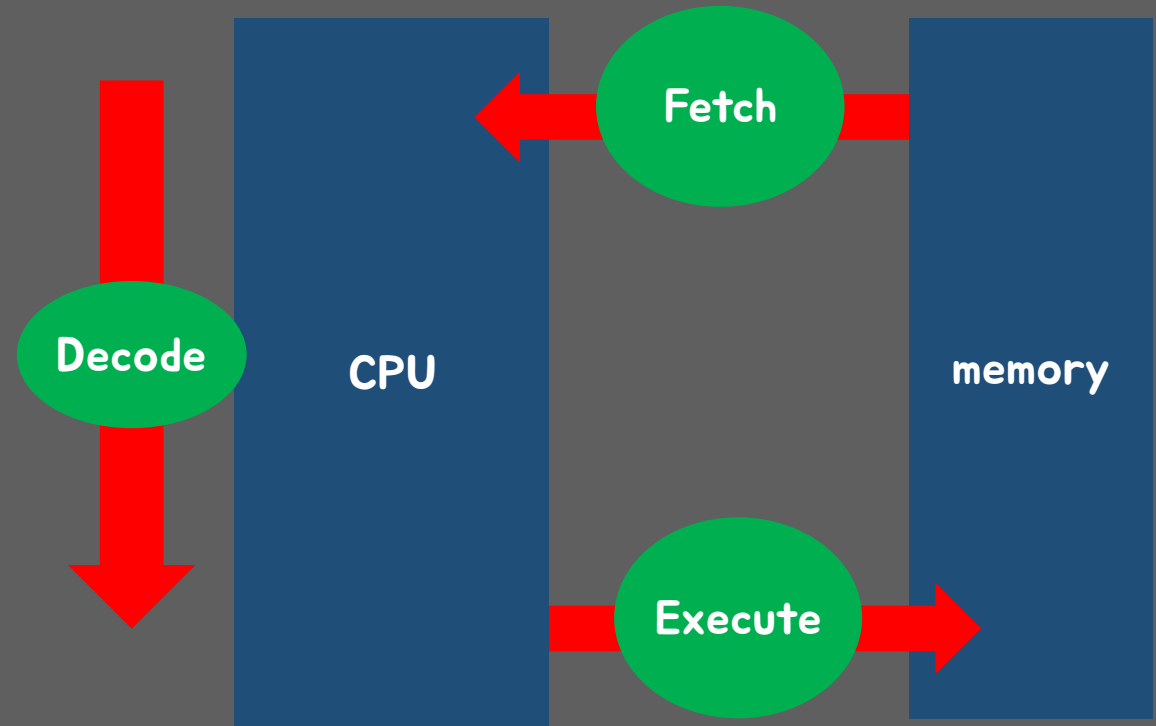
Assembly language

# | Prior knowledge

CPU

Register

Assembly language



# | Prior knowledge

CPU

Register

Assembly language

rax rbp  
rbx rip  
rcx  
rdx  
rsp

# | Prior knowledge

CPU

Register

Assembly language

|             |            |             |            |            |
|-------------|------------|-------------|------------|------------|
| <b>PUSH</b> | <b>POP</b> | <b>CALL</b> | <b>MOV</b> | <b>LEA</b> |
| <b>ADD</b>  | <b>SUB</b> | <b>TEST</b> | <b>CMP</b> | <b>JMP</b> |
| <b>JNE</b>  | <b>JE</b>  | <b>JLE</b>  |            |            |

# Training

# Training #1





# Training #1

00401238 68 141E4000 push crackme2.401E14 EntryPoint  
0040123D E8 F0FFFFFF call <JMP.&ThunRTMain>  
00401242 0000 add byte ptr ds:[eax],al  
00401244 0000 add byte ptr ds:[eax],al  
00401246 0000 add byte ptr ds:[eax],al  
00401248 3000 xor byte ptr ds:[eax],al  
0040124A 0000 add byte ptr ds:[eax],al  
0040124C 40 inc eax  
0040124D 0000 add byte ptr ds:[eax],al  
0040124F 0000 add byte ptr ds:[eax],al  
00401251 0000 add byte ptr ds:[eax],al  
00401253 0027 add byte ptr ds:[edi],ah  
00401255 8CE2 mov edx,edi edi:EntryPoint  
00401257 94 xchg esp,eax edx:EntryPoint  
00401258 3C 70 cmp al,70  
0040125A D311 rcl dword ptr ds:[ecx],cl  
0040125C B3 95 mov bl,95  
0040125E 0000 add byte ptr ds:[eax],al  
00401260 B4 39 mov ah,39  
00401262 824A 00 00 or byte ptr ds:[edx],0  
00401266 0000 add byte ptr ds:[eax],al  
00401268 0000 add byte ptr ds:[eax],al  
0040126A 0100 add dword ptr ds:[eax],eax  
0040126C 0000 add byte ptr ds:[eax],al  
0040126E 0000 add byte ptr ds:[eax],al  
00401270 FC cld  
00401271 3A75 01 cmp dh,byte ptr ss:[ebp+1]  
00401274 50 push eax  
00401275 72 6F push crackme2.4012E6  
00401277 6A 65 push 65  
00401279 687431 00 C1 imul esi,dword ptr ds:[ecx+esi],FFFFFFC esi:EntryPoint  
0040127E 40 inc eax  
0040127F 0008 add byte ptr ds:[eax],cl  
00401281 C140 00 00 rol dword ptr ds:[eax],0  
00401285 0000 add byte ptr ds:[eax],al  
00401287 00FF add bh,bh  
00401289 CC int3  
0040128A 3100 xor dword ptr ds:[eax],eax  
0040128C 0A23 or ah,byte ptr ds:[ebx]  
0040128E 8CE2 mov edx,edi edx:EntryPoint  
00401290 94 xchg esp,eax

crackme2.00401E14

.text:00401238 crackme2.exe:\$1238 #1238 <EntryPoint>

주소 Hex ASCII

|          |             |             |             |             |                      |
|----------|-------------|-------------|-------------|-------------|----------------------|
| 772B1000 | 16 00 18 00 | C0 8B 2B 77 | 14 00 16 00 | 38 84 2B 77 | .....A..w....8..+w   |
| 772B1010 | 00 00 02 00 | 80 5B 2B 77 | 0E 00 10 00 | E0 8D 2B 77 | .....[w....a..+w     |
| 772B1020 | 0C 00 0E 00 | D0 8D 2B 77 | 06 00 08 00 | 80 8D 2B 77 | .....D..w....'..+w   |
| 772B1030 | 06 00 08 00 | C0 8D 2B 77 | 06 00 08 00 | B8 8D 2B 77 | .....A..w....'..+w   |
| 772B1040 | 06 00 08 00 | C8 8D 2B 77 | 08 00 0A 00 | 70 83 2B 77 | .....E..w....p..+w   |
| 772B1050 | 1C 00 1E 00 | 6C 84 2B 77 | 2A 00 2C 00 | C4 8C 2B 77 | .....l..+w*...A..+w  |
| 772B1060 | 08 00 0A 00 | D8 8B 2B 77 | 02 00 04 00 | 98 8D 2B 77 | .....O..w....'..+w   |
| 772B1070 | 08 00 0A 00 | A4 D7 2B 77 | 18 00 1A 00 | 50 84 2B 77 | .....x..w....'..+w   |
| 772B1080 | 1C 00 1E 00 | 70 D9 2B 77 | 28 00 2A 00 | 44 D9 2B 77 | .....pU+w(. *..pU+w  |
| 772B1090 | 34 00 36 00 | 0C D9 2B 77 | 1E 00 20 00 | EC D8 2B 77 | 4..6...U..w....'..+w |
| 772B10A0 | 1A 00 1C 00 | D0 D8 2B 77 | 18 00 1A 00 | B4 D8 2B 77 | .....D0+w....'..+w   |
| 772B10B0 | 20 00 22 00 | 90 D8 2B 77 | 30 00 32 00 | 3C D8 2B 77 | .....O..w....'..+w   |
| 772B10C0 | 2C 00 2E 00 | 2C D8 2B 77 | 20 00 22 00 | 08 D8 2B 77 | .....O..w....'..+w   |
| 772B10D0 | 18 00 1A 00 | EC D7 2B 77 | 10 00 12 00 | D8 D7 2B 77 | .....l..x+w....'..+w |
| 772B10E0 | 36 00 38 00 | A4 D9 2B 77 | 08 00 0A 00 | A4 8D 2B 77 | 6..8..xU+w....'..+w  |
| 772B10F0 | 06 00 08 00 | 9C 8D 2B 77 | 41 63 4D 67 | FF FF FF 7F | .....+wACmgyyy..     |
| 772B1100 | 02 00 00 00 | 24 55 2B 77 | 00 00 00 00 | 00 00 00 00 | .....\$U..w....'..+w |

명령:

일지 중지됨 INT3 중단점 "진입점 중단점" 적용! <crackme2.EntryPoint> (00401238)

FPU 숨기기

EAX 0019FFCC  
EBX 002E7000  
ECX 00401238 <crackme2.EntryPoint>  
EDX 00401238 <crackme2.EntryPoint>  
EBP 0019FF80  
ESP 0019FF74  
ESI 00401238 <crackme2.EntryPoint>  
EDI 00401238 <crackme2.EntryPoint>  
EIP 00401238 <crackme2.EntryPoint>

EFlags 00000244  
ZF 1 PF 1 AF 0  
OF 0 SF 0 DF 0  
CF 0 TF 0 IF 1

LastError 00000000 (ERROR\_SUCCESS)  
LastStatus C0000008 (STATUS\_INVALID\_HANDLE)

GS 002B FS 0053  
ES 002B DS 002B  
CS 0023 SS 002B

ST(0) 000000000000000000000000 x87r0 비어 있음 0.000000000000000000000000  
ST(1) 000000000000000000000000 x87r1 비어 있음 0.000000000000000000000000  
ST(2) 000000000000000000000000 x87r2 비어 있음 0.000000000000000000000000  
ST(3) 000000000000000000000000 x87r3 비어 있음 0.000000000000000000000000  
ST(4) 000000000000000000000000 x87r4 비어 있음 0.000000000000000000000000  
ST(5) 000000000000000000000000 x87r5 비어 있음 0.000000000000000000000000  
ST(6) 000000000000000000000000 x87r6 비어 있음 0.000000000000000000000000  
ST(7) 000000000000000000000000 x87r7 비어 있음 0.000000000000000000000000

x87Tagword FFFF  
x87TW\_0 3 (비어 있음) x87TW\_1 3 (비어 있음)  
x87TW\_2 3 (비어 있음) x87TW\_3 3 (비어 있음)

기본값 (stdcall) 5 잠금 해제됨

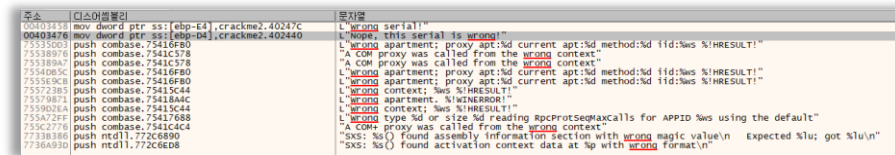
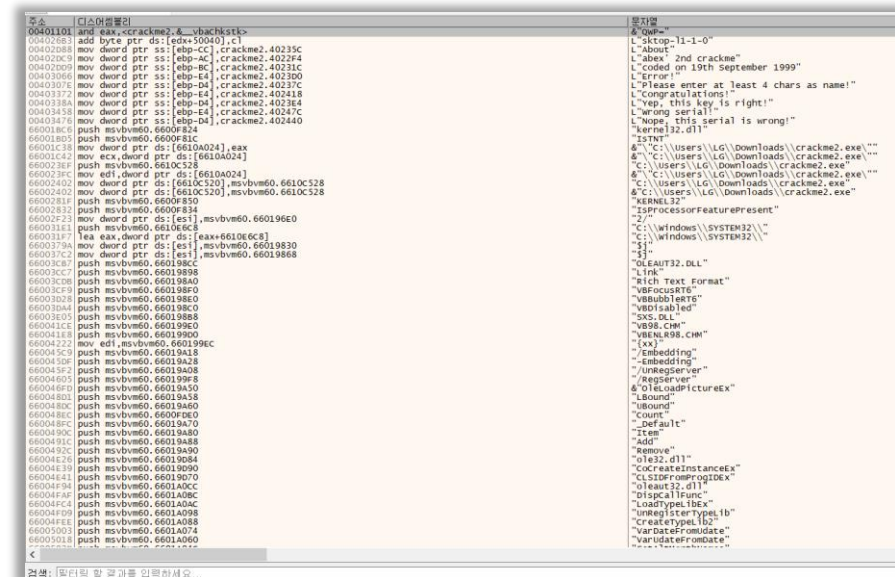
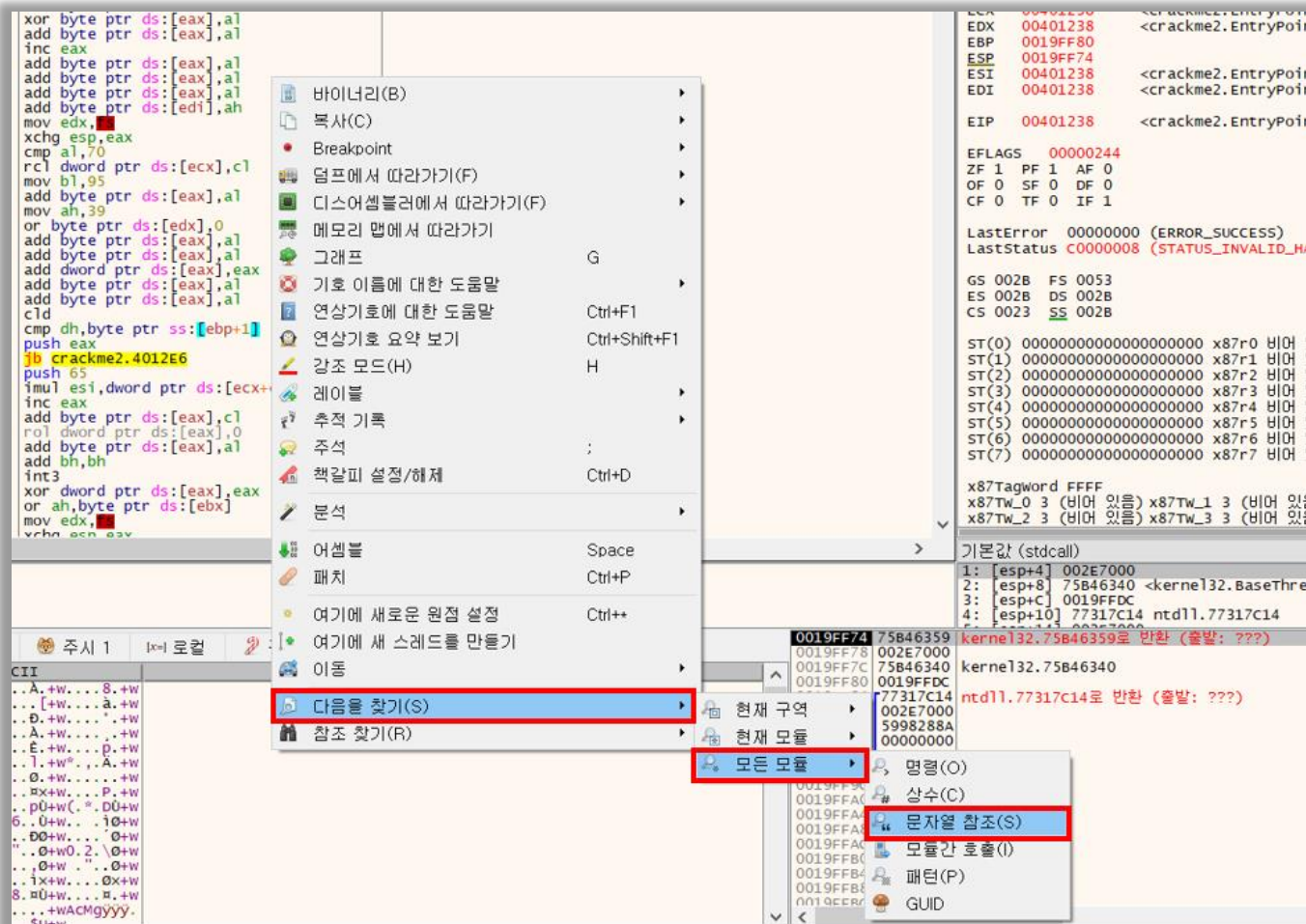
1: [esp+4] 002E7000  
2: [esp+8] 75B46340 <kernel32.BaseThreadInitThunk>  
3: [esp+C] 0019FFDC  
4: [esp+10] 77317C14 ntdll.77317C14

0019FF74 75B46359 kernel32.75B46359로 반환 (출발: ???)  
0019FF78 002E7000  
0019FF7C 75B46340  
0019FF80 0019FFDC  
0019FF84 77317C14  
0019FF88 002E7000  
0019FF8C 5998288A  
0019FF90 00000000  
0019FF94 00000000  
0019FF98 002E7000  
0019FF9C 00000000  
0019FFA0 00000000  
0019FFA4 00000000  
0019FFA8 00000000  
0019FFAC 00000000  
0019FFB0 00000000  
0019FFB4 00000000  
0019FFB8 00000000  
0019FFBC 00000000

기본값

디버깅에 사용한 시간: 0:02:13:21

# Training #1





# Training #1

|          |                       |   |                                       |
|----------|-----------------------|---|---------------------------------------|
| 00403413 | 8D45 BC               | lea eax,dword ptr ss:[ebp-44]             |                                       |
| 00403416 | 8D4D CC               | lea ecx,dword ptr ss:[ebp-34]             | ecx:EntryPoint                        |
| 00403419 | 50                    | push eax                                  |                                       |
| 0040341A | 51                    | push ecx                                  | ecx:EntryPoint                        |
| 0040341B | FF15 A4104000         | call dword ptr ds:[<__vbaVarTstNe>]       |                                       |
| 00403421 | 66:85C0               | test ax,ax                                |                                       |
| 00403424 | 0F84 C7000000         | je crackme2.4034F1                        |                                       |
| 0040342A | 899D 34FFFFFF         | mov dword ptr ss:[ebp-CC],ebx             |                                       |
| 00403430 | 899D 44FFFFFF         | mov dword ptr ss:[ebp-BC],ebx             |                                       |
| 00403436 | B8 04000280           | mov eax,80020004                          |                                       |
| 0040343B | BB 08000000           | mov ebx,8                                 |                                       |
| 00403440 | 8D95 14FFFFFF         | lea edx,dword ptr ss:[ebp-EC]             | edx:EntryPoint                        |
| 00403446 | 8D8D 54FFFFFF         | lea ecx,dword ptr ss:[ebp-AC]             | ecx:EntryPoint                        |
| 0040344C | 8985 3CFFFFFF         | mov dword ptr ss:[ebp-C4],eax             |                                       |
| 00403452 | 8985 4CFFFFFF         | mov dword ptr ss:[ebp-B4],eax             |                                       |
| 00403458 | C785 1CFFFFFF 7C24400 | mov dword ptr ss:[ebp-E4],crackme2.40247C | 40247C:L"wrong serial!"               |
| 00403462 | 899D 14FFFFFF         | mov dword ptr ss:[ebp-EC],ebx             |                                       |
| 00403468 | FFD7                  | call edi                                  | edi:EntryPoint                        |
| 0040346A | 8D95 24FFFFFF         | lea edx,dword ptr ss:[ebp-DC]             | edx:EntryPoint                        |
| 00403470 | 8D8D 64FFFFFF         | lea ecx,dword ptr ss:[ebp-9C]             | ecx:EntryPoint                        |
| 00403476 | C785 2CFFFFFF 4024400 | mov dword ptr ss:[ebp-D4],crackme2.402440 | 402440:L"Nope, this serial is wrong!" |

|          |                       |   |                                   |
|----------|-----------------------|---|-----------------------------------|
| 00403321 | 8D55 BC               | lea edx,dword ptr ss:[ebp-44]             | edx:EntryPoint                    |
| 00403324 | 8D45 CC               | lea eax,dword ptr ss:[ebp-34]             |                                   |
| 00403327 | 52                    | push edx                                  | edx:EntryPoint                    |
| 00403328 | 50                    | push eax                                  |                                   |
| 00403329 | FF15 58104000         | call dword ptr ds:[<__vbaVarTstEq>]       |                                   |
| 0040332F | 66:85C0               | test ax,ax                                |                                   |
| 00403332 | 0F84 D0000000         | je crackme2.403408                        |                                   |
| 00403338 | B8 04000280           | mov eax,80020004                          |                                   |
| 0040333D | BB 0A000000           | mov ebx,A                                 | A:'\n'                            |
| 00403342 | 89BD 14FFFFFF         | mov dword ptr ss:[ebp-EC],edi             | edi:EntryPoint                    |
| 00403348 | 8B3D B0104000         | mov edi,dword ptr ds:[<__vbaVarDup>]      | edi:EntryPoint                    |
| 0040334E | 8D95 14FFFFFF         | lea edx,dword ptr ss:[ebp-EC]             | edx:EntryPoint                    |
| 00403354 | 8D8D 54FFFFFF         | lea ecx,dword ptr ss:[ebp-AC]             | ecx:EntryPoint                    |
| 0040335A | 8985 3CFFFFFF         | mov dword ptr ss:[ebp-C4],eax             |                                   |
| 00403360 | 899D 34FFFFFF         | mov dword ptr ss:[ebp-CC],ebx             |                                   |
| 00403366 | 8985 4CFFFFFF         | mov dword ptr ss:[ebp-B4],eax             |                                   |
| 0040336C | 899D 44FFFFFF         | mov dword ptr ss:[ebp-BC],ebx             |                                   |
| 00403372 | C785 1CFFFFFF 1824400 | mov dword ptr ss:[ebp-E4],crackme2.402418 | 402418:L"Congratulations!"        |
| 0040337C | FFD7                  | call edi                                  | edi:EntryPoint                    |
| 0040337E | 8D95 24FFFFFF         | lea edx,dword ptr ss:[ebp-DC]             | edx:EntryPoint                    |
| 00403384 | 8D8D 64FFFFFF         | lea ecx,dword ptr ss:[ebp-9C]             | ecx:EntryPoint                    |
| 0040338A | C785 2CFFFFFF E423400 | mov dword ptr ss:[ebp-D4],crackme2.4023E4 | 4023E4:L"Yep, this key is right!" |

# Training #1

```
00403307 8985 6CFFFFFF mov dword ptr ss:[ebp-94],eax [ebp-94]:L"da1t0ry"
0040330D 89BD 64FFFFFF mov dword ptr ss:[ebp-9C],edi
00403313 FFD6 call esi
00403315 8D8D 74FFFFFF lea ecx,dword ptr ss:[ebp-8C]
0040331B FF15 C8104000 call dword ptr ds:[<&_vbaFreeObj>]
00403321 8D55 BC lea edx,dword ptr ss:[ebp-44]
00403324 8D45 CC lea eax,dword ptr ss:[ebp-34]
00403327 52 push edx
00403328 50 push eax
00403329 FF15 58104000 call dword ptr ds:[<&_vbaVarTstEq>]
0040332F 66:85C0 test ax,ax
00403332 0F84 D0000000 je crackme2.403408
00403338 B8 04000280 mov eax,80020004
0040333D BB 0A000000 mov ebx,A A: '\n'
00403342 89BD 14FFFFFF mov dword ptr ss:[ebp-EC],edi
00403348 8B3D B0104000 mov edi,dword ptr ds:[<&_vbaVarDup>]
0040334E 8D95 14FFFFFF lea edx,dword ptr ss:[ebp-EC]
00403354 8D8D 54FFFFFF lea ecx,dword ptr ss:[ebp-AC]
0040335A 8985 3CFFFFFF mov dword ptr ss:[ebp-C4],eax
00403360 899D 34FFFFFF mov dword ptr ss:[ebp-CC],ebx
00403366 8985 4CFFFFFF mov dword ptr ss:[ebp-B4],eax
0040336C 899D 44FFFFFF mov dword ptr ss:[ebp-BC],ebx
00403372 C785 1CFFFFFF 1824400 mov dword ptr ss:[ebp-E4],crackme2.402418:L"Congratulations!"
0040337C FFD7 call edi
0040337E 8D95 24FFFFFF lea edx,dword ptr ss:[ebp-DC]
00403384 8D8D 64FFFFFF lea ecx,dword ptr ss:[ebp-9C]
0040338A C785 2CEEEEE E423400 mov dword ptr ss:[ebp-D4],crackme2.4023E4:L"Yep, this key is right!"
```

|            |          |
|------------|----------|
| EAX        | 0019F2A0 |
| EBX        | 660E702F |
| ECX        | 022BC9AC |
| <u>EDX</u> | 0019F290 |
| <u>EBP</u> | 0019F2D4 |
| ESP        | 0019F170 |
| ESI        | 66106AEE |
| EDI        | 00000008 |

|          |          |                   |
|----------|----------|-------------------|
| 0019F280 | 66100008 | msvbvm60.66100008 |
| 0019F284 | 0019F280 |                   |
| 0019F288 | 0062455C | L"D8"             |
| 0019F28C | 0019F238 |                   |
| 0019F290 | 00000008 |                   |
| 0019F294 | 0019F280 |                   |
| 0019F298 | 0062469C | L"C8C595D8"       |
| 0019F29C | 0019F238 |                   |
| 0019F2A0 | 00000008 |                   |
| 0019F2A4 | 0019F280 |                   |
| 0019F2A8 | 006247B4 | L"da1t0ry"        |
| 0019F2AC | 0019F238 |                   |
| 0019F2B0 | 00000002 |                   |
| 0019F2B4 | 006238EC | L"da1t0ry"        |

## abex' 2nd crackme

Name:

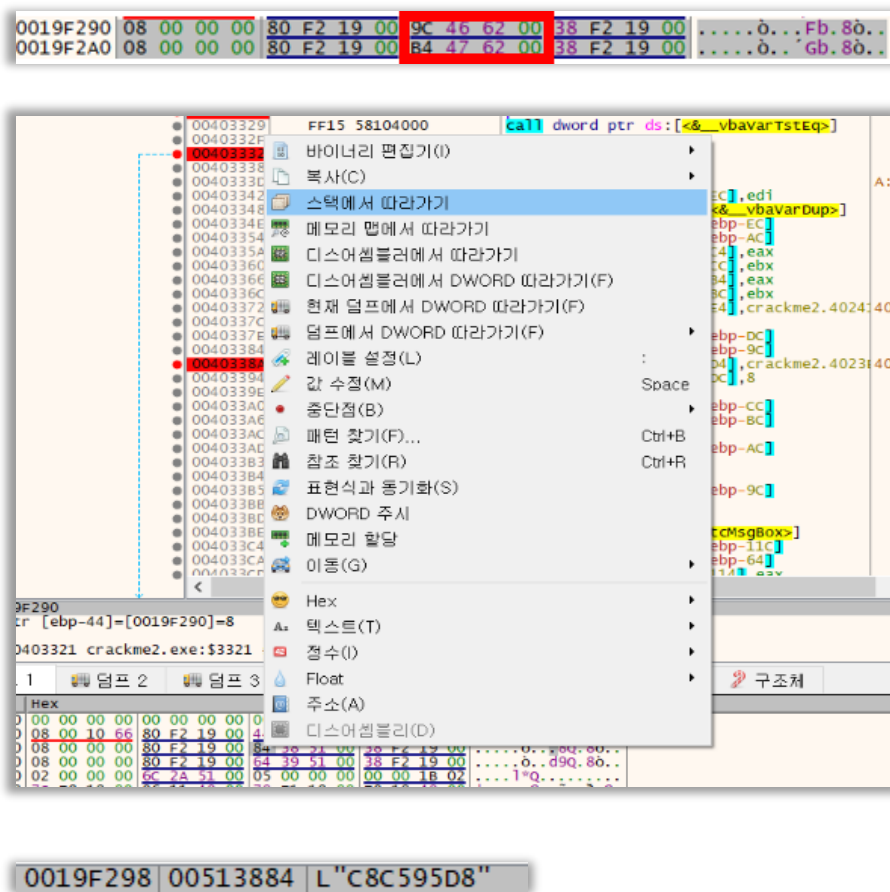
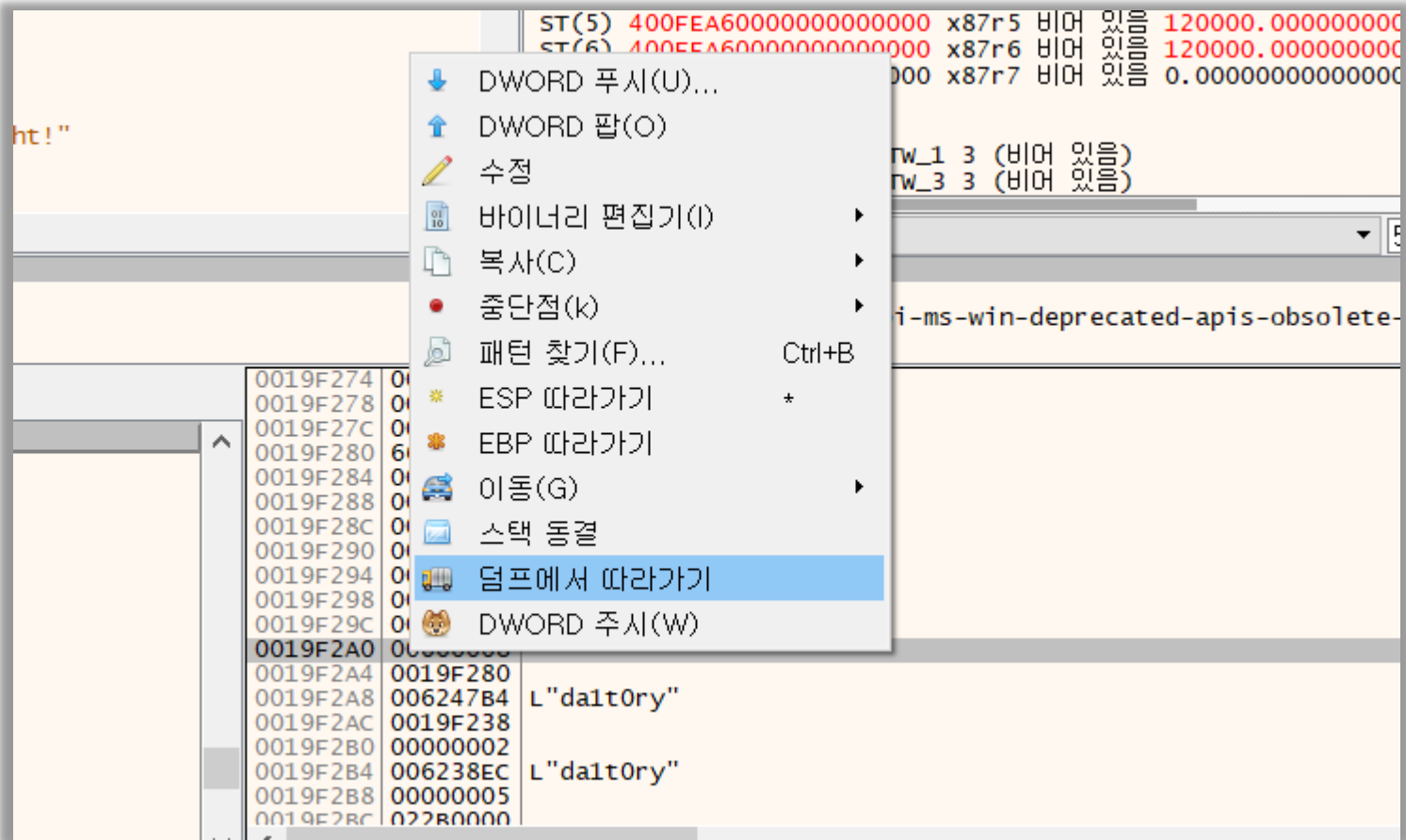
Check

Serial:

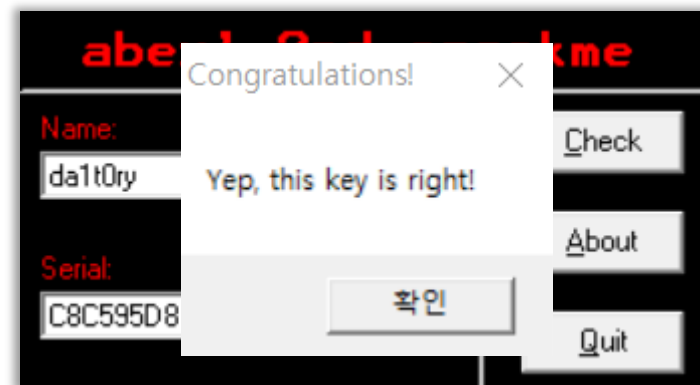
About

Quit

# Training #1

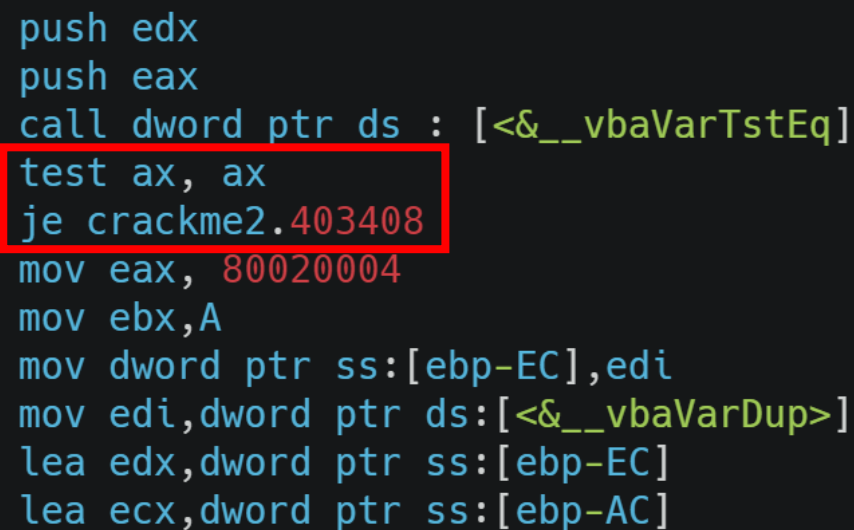


# Training #1





# Training #2



```
push edx
push eax
call dword ptr ds : [<_vbaVarTstEq]
test ax, ax
je crackme2.403408
mov eax, 80020004
mov ebx, A
mov dword ptr ss:[ebp-EC], edi
mov edi, dword ptr ds:[<_vbaVarDup>]
lea edx, dword ptr ss:[ebp-EC]
lea ecx, dword ptr ss:[ebp-AC]
```

# Training #2

```
00403321 8D55 BC lea edx,dword ptr ss:[ebp-44] edx:EntryPoint
00403324 8D45 CC lea eax,dword ptr ss:[ebp-34] edx:EntryPoint
00403327 52 push edx
00403328 50 push eax
00403329 FF15 58104000 call dword ptr ds:[<&__vbaVarTstEq>]
0040332F 66:85C0 test ax,ax
00403332 0F84 D0000000 je crackme2.403408
00403338 B8 04000280 mov eax,80020004
0040333D BB 0A000000 mov ebx,A
00403342 89BD 14FFFFFF mov dword ptr ss:[ebp-EC],edi
00403348 8B3D B0104000 mov edi,dword ptr ds:[<&__vbaVarDup>]
0040334E 8D95 14FFFFFF lea edx,dword ptr ss:[ebp-EC]
00403354 8D8D 54FFFFFF lea ecx,dword ptr ss:[ebp-AC]
0040335A 8985 3CFFFFFF mov dword ptr ss:[ebp-C4],eax
00403360 899D 34FFFFFF mov dword ptr ss:[ebp-CC],ebx
00403366 8985 4CFFFFFF mov dword ptr ss:[ebp-B4],eax
0040336C 899D 44FFFFFF mov dword ptr ss:[ebp-BC],ebx
00403372 C785 1CFFFFFF 1824400 mov dword ptr ss:[ebp-E4],crackme2.402418
0040337C FFD7 call edi
0040337E 8D95 24FFFFFF lea edx,dword ptr ss:[ebp-DC]
00403384 8D8D 64FFFFFF lea ecx,dword ptr ss:[ebp-9C]
0040338A C785 2CFFFFFF E423400 mov dword ptr ss:[ebp-D4],crackme2.4023E4
```

00403332 어셈블

jne 0x00403408

☐ 크기 유지(S) ☐ 잔존 바이트를 NOP로 채우기(F) ☐ XEDParse ☒ asmiit

명령어가 성공적으로 인코딩되었습니다!

현재 상황

어떤 값을 넣든 ax의 값이 같음

JE

값이 같을 경우  
에러 출력 주소로 이동한다.

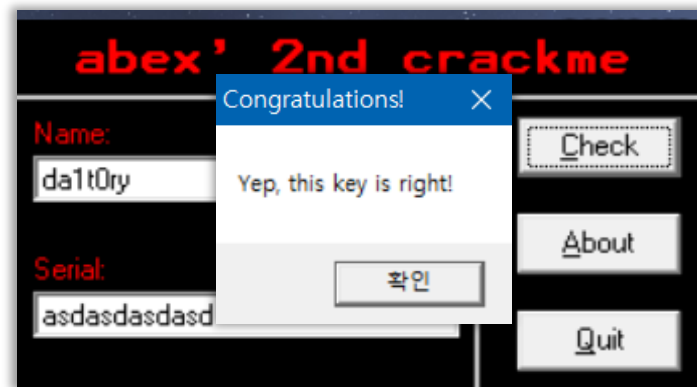
변경

JNE

값이 같지 않을 경우  
에러 출력 주소로 이동한다.



# Training #2



# QnA