



Memory Hacking



[with Cheat Engine]

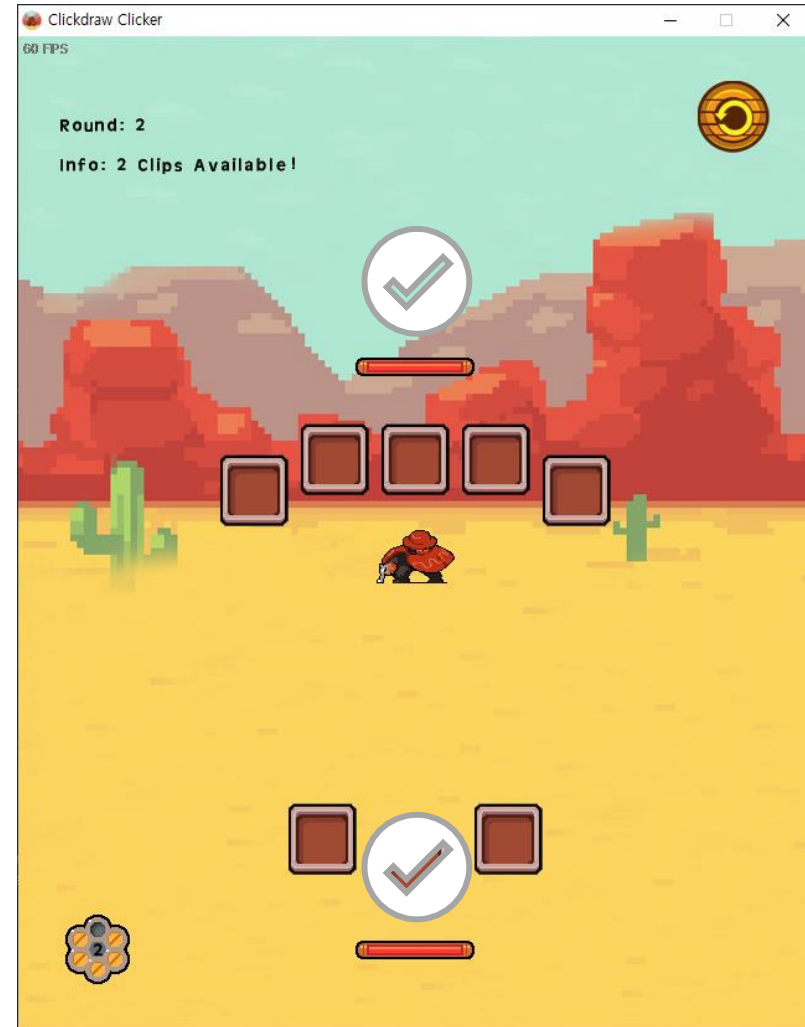
이다영



메모리 해킹?

: 메모리에 접근하여 저장된 값을 수정하거나 코드를 끼워넣는 메모리 변조

Memory Hacking





메모리 변조의 원리

- 01 프로세스의 메모리 불러오기
 - 02 원하는 값 스캔
 - 03 그 값을 자신이 원하는 값으로 변경
- ➡ 메모리 스캐너를 이용하여 값을 수정
- └──────────┬──────────> 치트엔진

[Level 1]

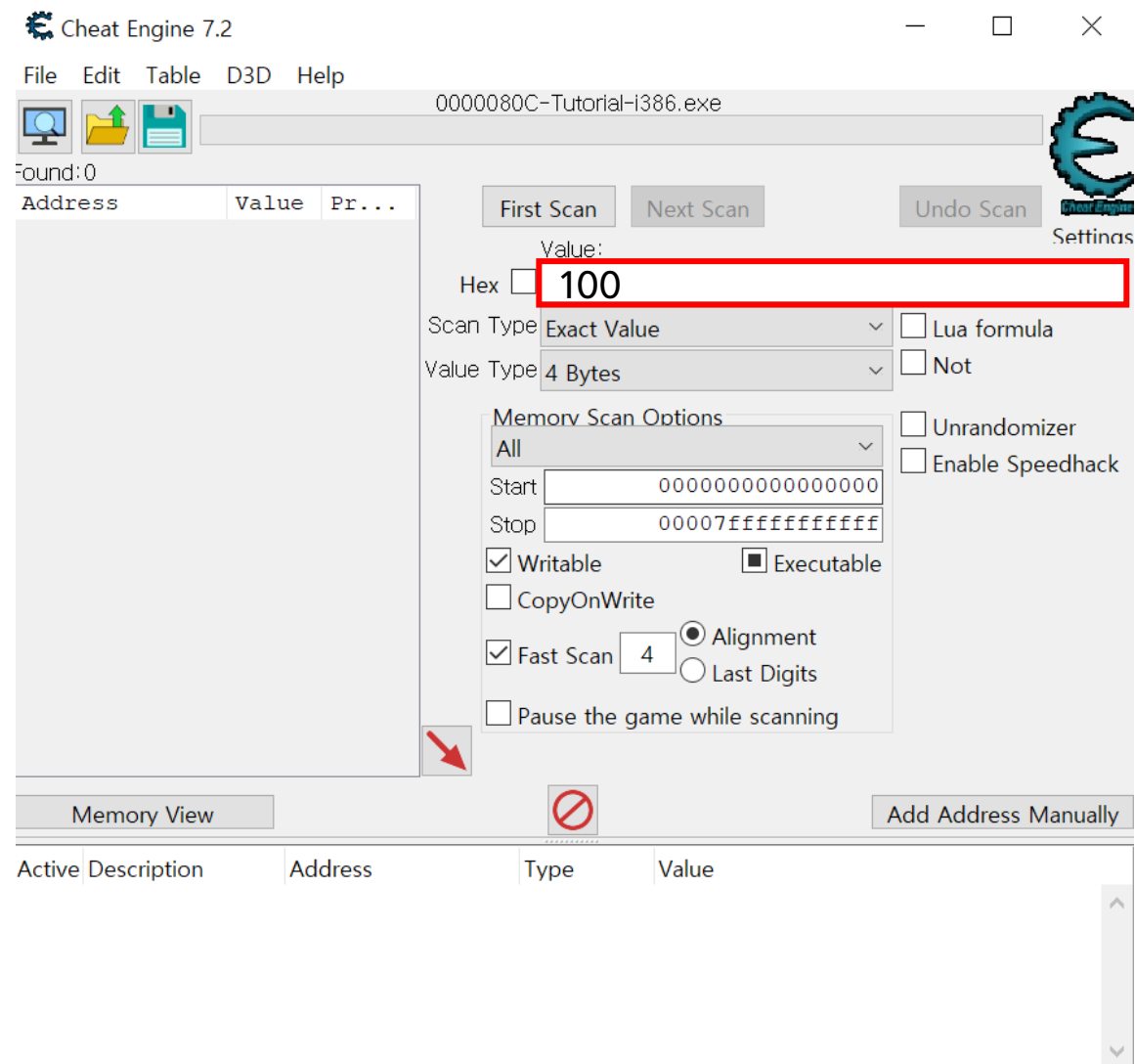
100으로 설정이 되어 있는 체력
(Health) 값을 1000으로 바꾸자

Health: 100

Hit me

Next

Memory Hacking



Memory Hacking

Found: 59

Address	Value	Pr...
"Tutorial-i...	100	100
"Tutorial-i...	100	100
"Tutorial-i...	100	100
0165B6DC	100	100
0165CCD4	100	100
0165ED9C	100	100
0165EE04	100	100
0165F390	100	100
0165F848	100	100
0165FA5C	10...	100
01830860	100	100
01834668	100	100
0183D3BC	100	100
01843EAC	100	100
01849784	100	100
018497EC	100	100
018499AC	100	100

New Scan Next Scan Undo Scan Settings

Value: 100

Hex ☐ 100

Scan Type Exact Value ☐ Lua formula

Value Type 4 Bytes ☐ Not

☐ Compare to first scan ☐ Unrandomizer

☐ Enable Speedhack

Memory Scan Options

All

Start 0000000000000000

Stop 00007fffffffffffffff

☒ Writable ☐ Executable

☐ CopyOnWrite

☒ Fast Scan 4 ☒ Alignment ☐ Last Digits

☐ Pause the game while scanning

Memory Hacking

Found: 59

Address	Value	Pr...	^
0184CE20	100	100	
01854BC8	100	100	
0185D288	100	100	
0185D6C8	100	100	
0185D6E8	100	100	
0185E96C	100	100	
0185E98C	100	100	
01889350	100	100	
0188CD44	99	100	
018B199C	100	100	
019035F0	100	100	
01903810	100	100	
0192EF10	100	100	
061336EC	100	100	
06133870	100	100	
0613396C	100	100	
061339CC	100	100	

Health: 100

Hit me

Next

Health: 99

Hit me

Next

Memory Hacking

Found: 59

Address	Value	Pr...	^
0184CE20	100	100	
01854BC8	100	100	
0185D288	100	100	
0185D6C8	100	100	
0185D6E8	100	100	
0185E96C	100	100	
0185E98C	100	100	
01889350	100	100	
0188CD44	99	100	
018B199C	100	100	
019035F0	100	100	
01903810	100	100	
0192EF10	100	100	
061336EC	100	100	
06133870	100	100	
0613396C	100	100	
061339CC	100	100	

Found: 59

Address	Value	Pr...	^
0184CE20	100	100	
01854BC8	100	100	
0185D288	100	100	
0185D6C8	100	100	
0185D6E8	100	100	
0185E96C	100	100	
0185E98C	100	100	
01889350	100	100	
0188CD44	1000	100	
018B199C	100	100	
019035F0	100	100	
01903810	100	100	
0192EF10	100	100	
061336EC	100	100	
06133870	100	100	
0613396C	100	100	
061339CC	100	100	

Health: 996

Hit me

Next

[Level 2]

값을 변경하는 기능을 무효화하여
Change value를 눌러도 값이 변
경되지 않도록 하자

100

Change value

Next

Address	Type
01894FE8	4 Bytes

Address	Type
00145328	4 Bytes

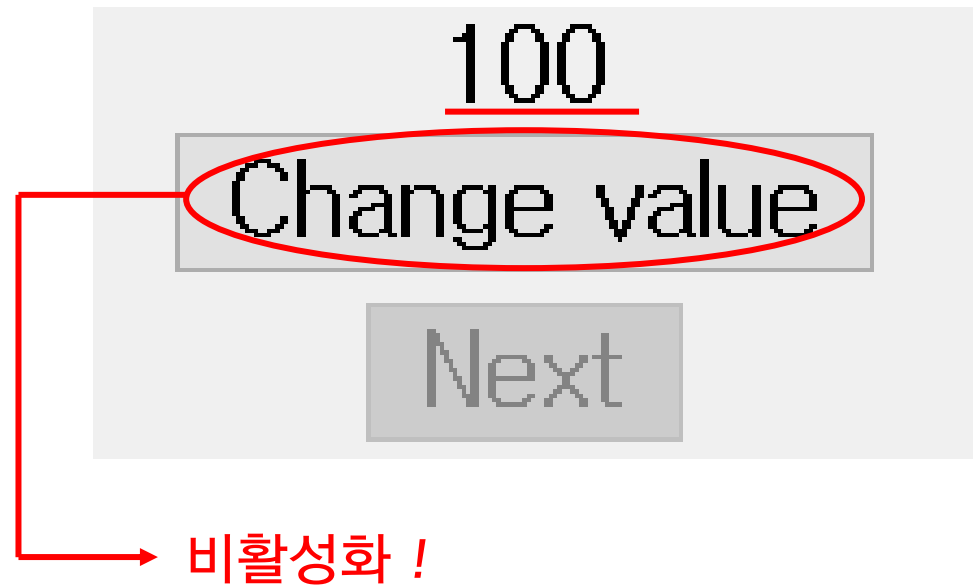
★ ASLR(Address Space Location Randomization)
: 프로그램을 실행시킬 때마다 각 주소들이 바뀌는
메모리 보호 기법



“소지금의 주소가 **동적**이 아니라 **고정**이라면?”



소지금의 주소만 알아낸다면 메모리를
변조하여 소지금을 쉽게 변경할 수 있음



Find out what writes to this address

: 해당 변수에 접근하여 값을 변경하는 기능을
수행하는 어셈블리어를 찾는다

Change value

⚙ The following opcodes write to 01775660 ✕

C...	Instruction
1	00426932 - 89 10 - mov [eax],edx

Replace

Show disassembler

Add to the codelist

More information

Select an item from the list for a small description

Stop

Code: X

What name do you want to give this code?

Change of nop

OK Cancel

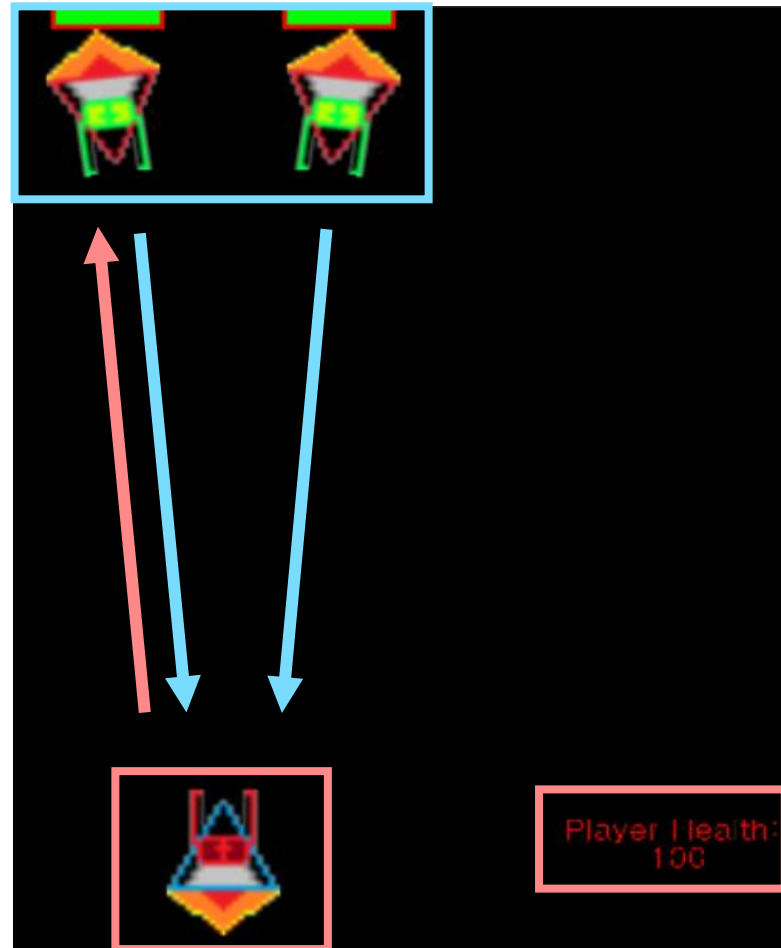
→ 아무 것도 수행하지 않는 명령어

596

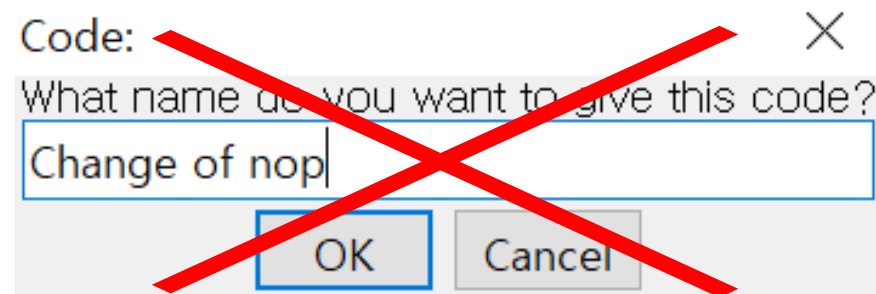
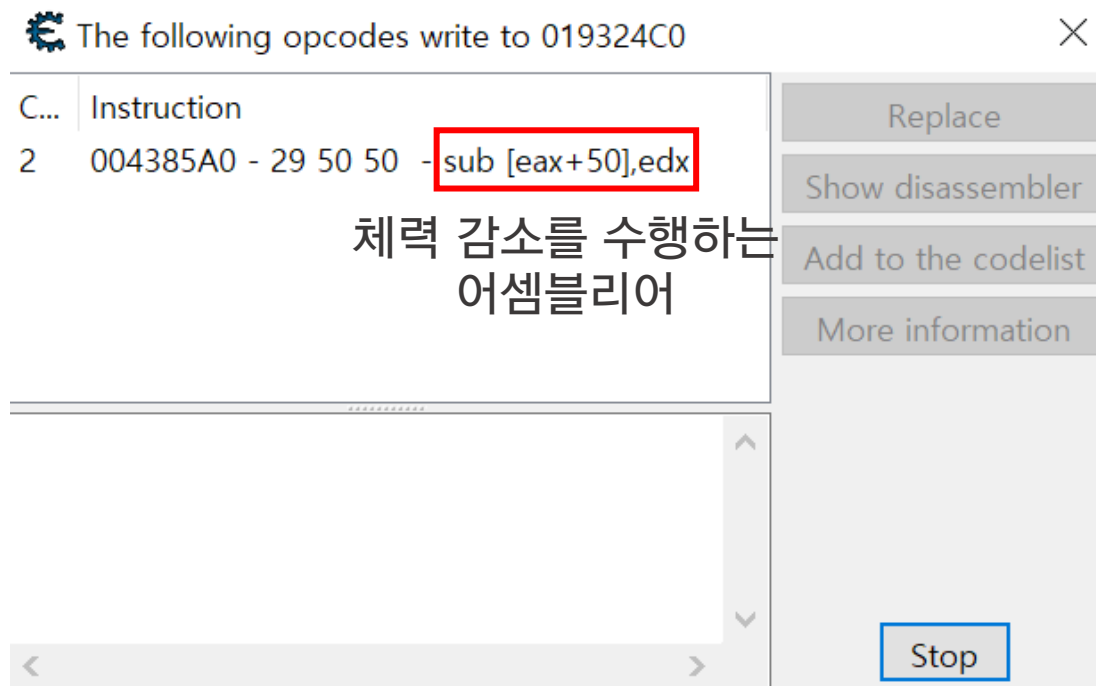
Change value

Next

[Level 3]



Memory Hacking



Find out what addresses this instruction accesses

: 어셈블리어가 접근하는 데이터를 찾는다

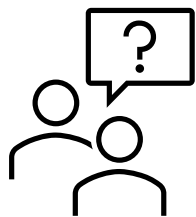
⚙️ Accessed addresses by 4385A0 — □ ×

Code Address 4385A0

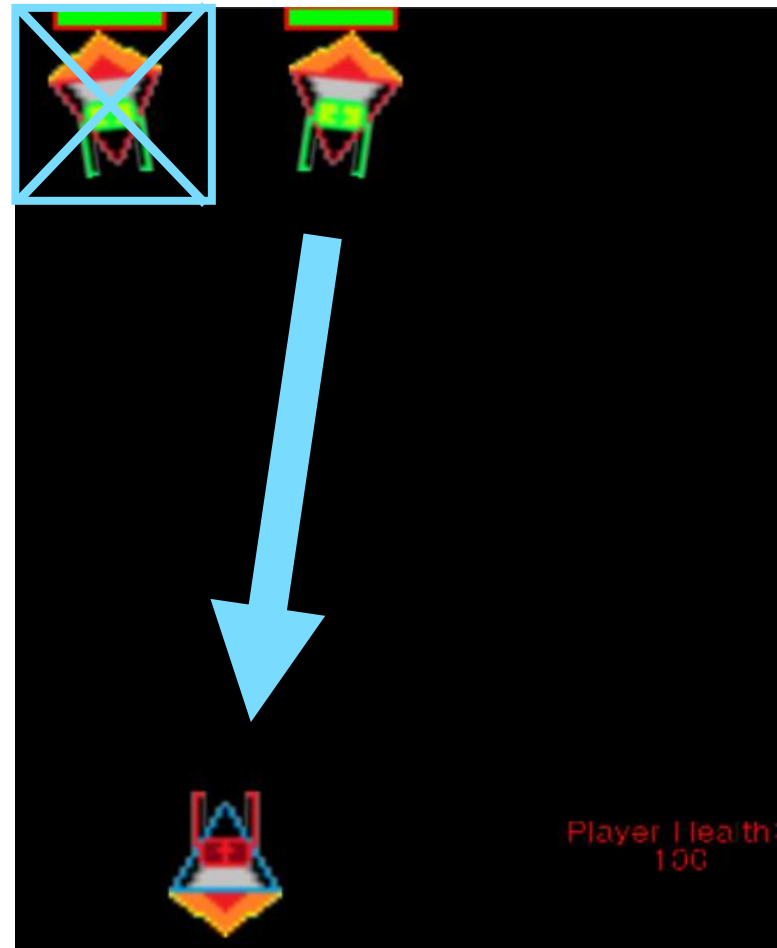
The following 3 addresses have been accessed by the code you selected

Address	Value	Count
01A01180	84	4
01A04DE0	197	1
01A04D70	199	1

Stop 4 Bytes ▾



“ 적들의 체력을 1로 바꾸자! ”





Accessed addresses by 4385A0



Code Address

4385A0

The following 3 addresses have been accessed by the code you selected

	Address	Value	Count
그룹 2	01A01180	84	4
	01A04DE0	197	1
그룹 1	01A04D70	199	1

Stop

4 Bytes



Memory Hacking

Offse...	Offse...	Offse...	G1:01A04D90	G1:01A04D20	G2:01A01130
1C			01A04DAC : 3193039749	01A04D3C : 1045556101	01A0114C : 0
20			01A04DB0 : 3209481421	01A04D40 : 3209481421	01A01150 : 1061997773
2C			01A04DBC : 1127002850	01A04D4C : 1127959838	01A0115C : 1083388722
50			01A04DE0 : 197	01A04D70 : 199	01A01180 : 84
54			01A04DE4 : 200	01A04D74 : 200	01A01184 : 100
58			01A04DE8 : 26939448	01A04D78 : 26939528	01A01188 : 0
5C			01A04DEC : 1	01A04D7C : 1	01A0118C : 0

^
체력
체력 최대치
적군/아군
구분

⚙ The following opcodes write to 019324C0 ✕

C...	Instruction
2	004385A0 - 29 50 50 - sub [eax+50],edx

Replace

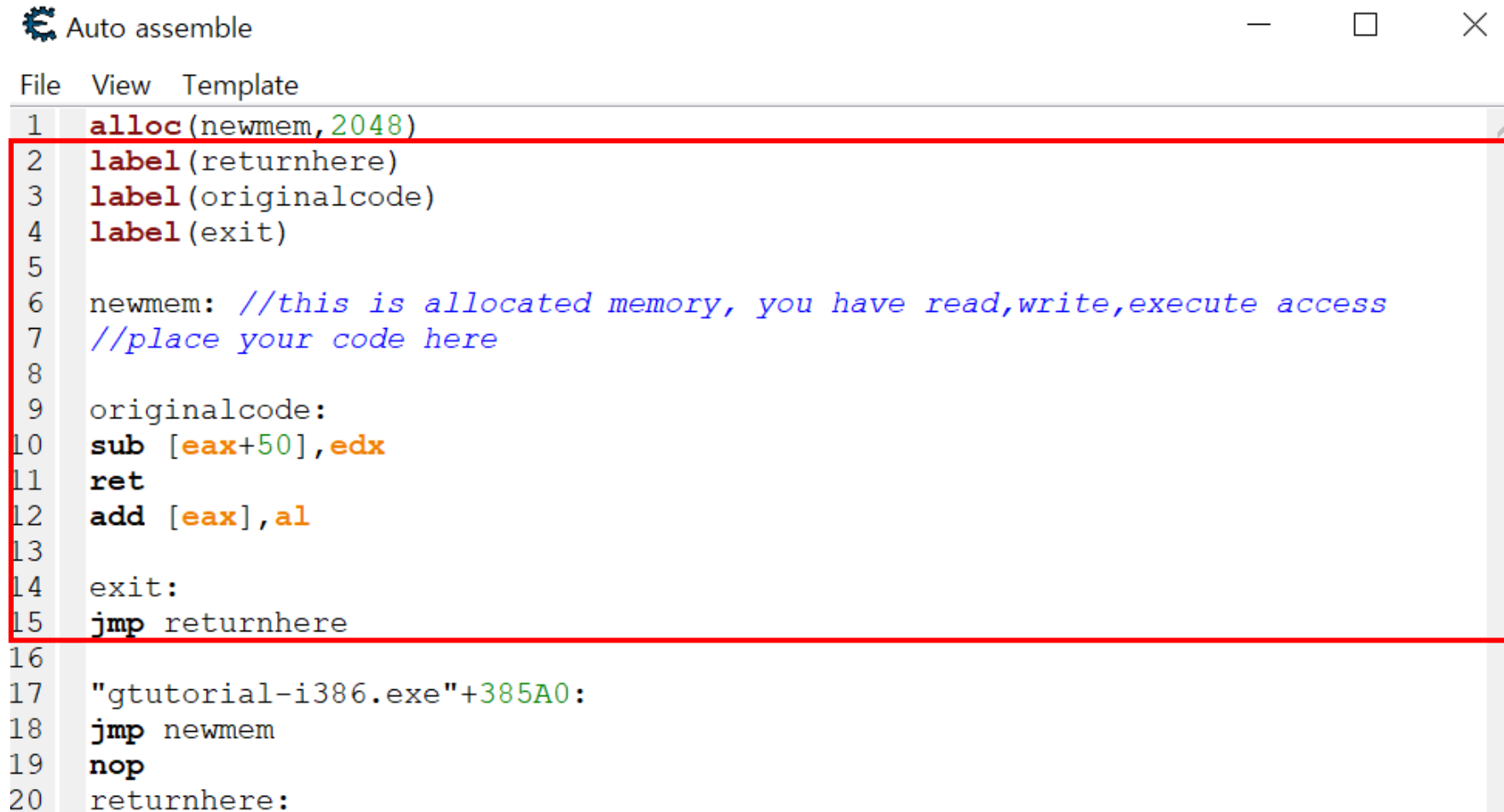
Show disassembler

Add to the codelist

More information

Stop

Code Injection(코드 인젝션) : 원하는 코드 삽입



```
Auto assemble
File View Template
1  alloc(newmem,2048)
2  label(returnhere)
3  label(originalcode)
4  label(exit)
5
6  newmem: //this is allocated memory, you have read,write,execute access
7  //place your code here
8
9  originalcode:
10 sub [eax+50],edx
11 ret
12 add [eax],al
13
14 exit:
15 jmp returnhere
16
17 "gtutorial-i386.exe"+385A0:
18 jmp newmem
19 nop
20 returnhere:
```

Auto assemble

File View Template

```
1  alloc (newmem, 2048)
```

```
2
```

```
3  newmem:
```

```
4  cmp [EAX+5C], 0
```

```
5  je friend
```

```
6  
```

→ 두 값이 같으면 점프

```
7  cmp [EAX+5C], 1
```

```
8  je enemy
```

```
9
```

```
10 friend:
```

```
11 mov [EAX+50], #9999
```

```
12 ret
```

```
13
```

```
14 enemy:
```

```
15 mov [EAX+50], 0
```

```
16 ret
```

```
17
```

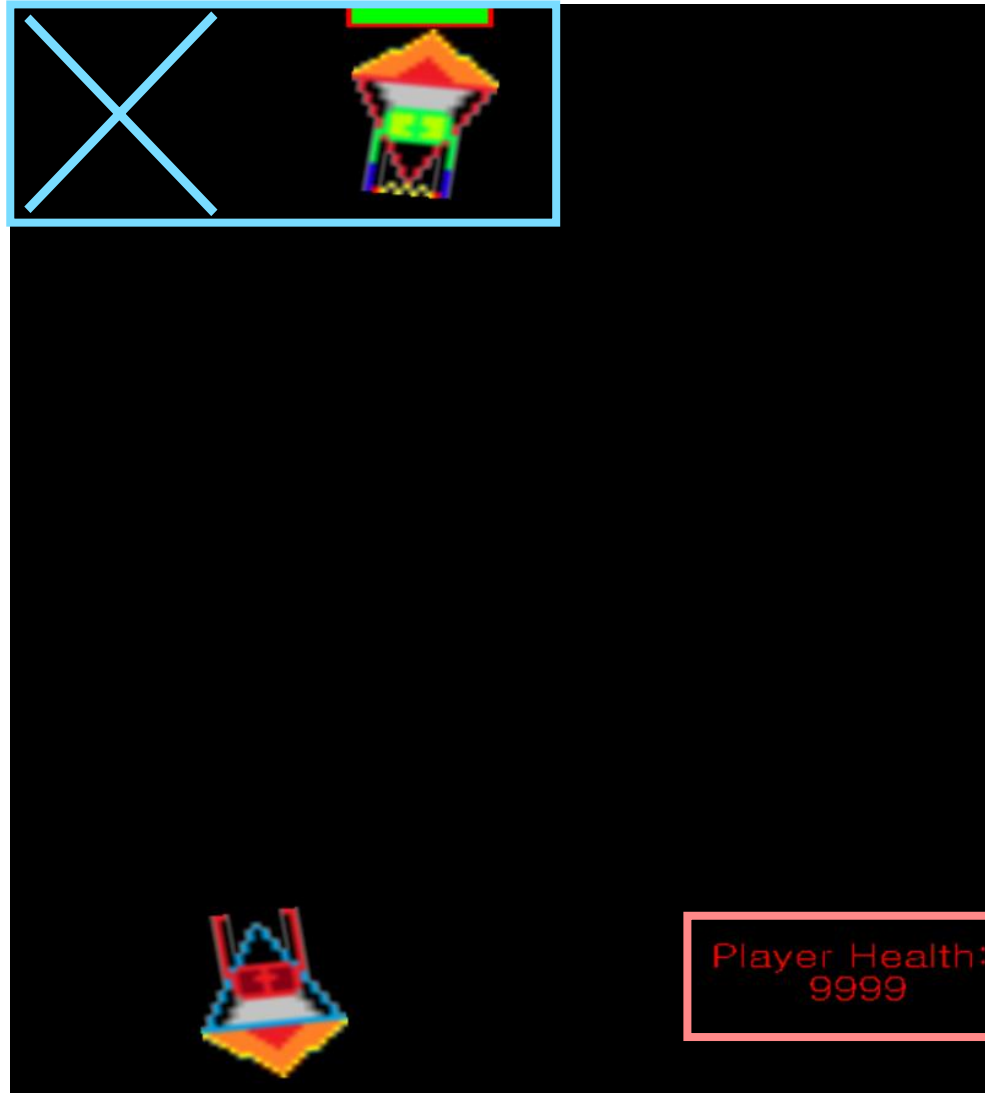
```
18 "gtutorial-i386.exe"+385A0:
```

```
19 jmp newmem
```

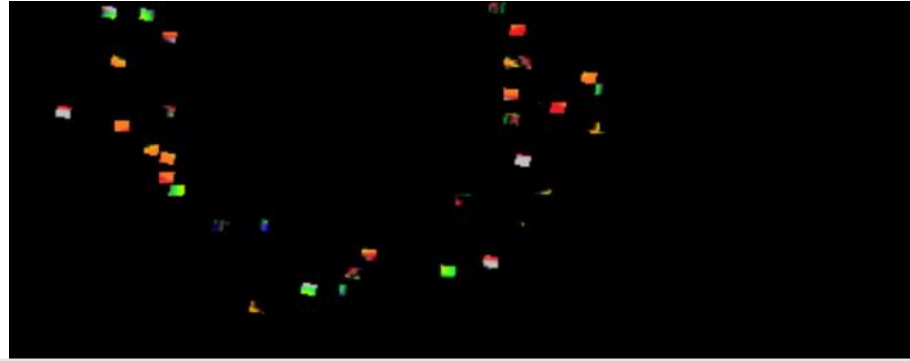
```
20 nop
```

```
21 returnhere:
```

Memory Hacking

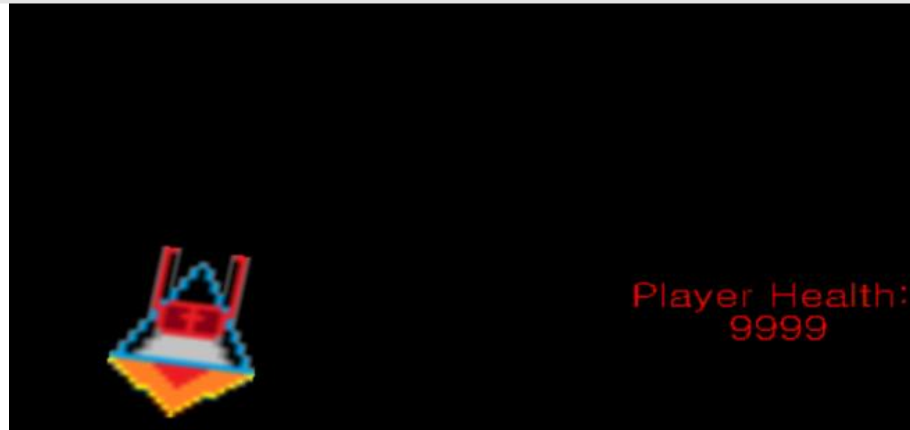


Memory Hacking



well done

OK



[Cheat Engine tutorial]



THANK YOU



[감사합니다.]