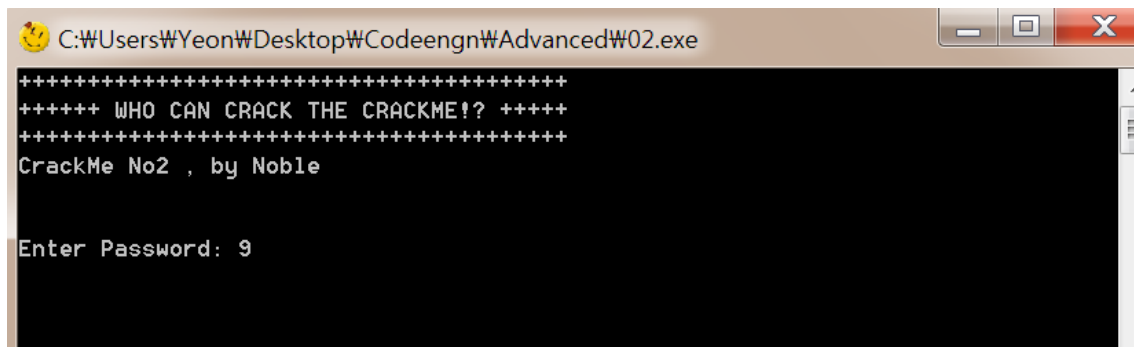


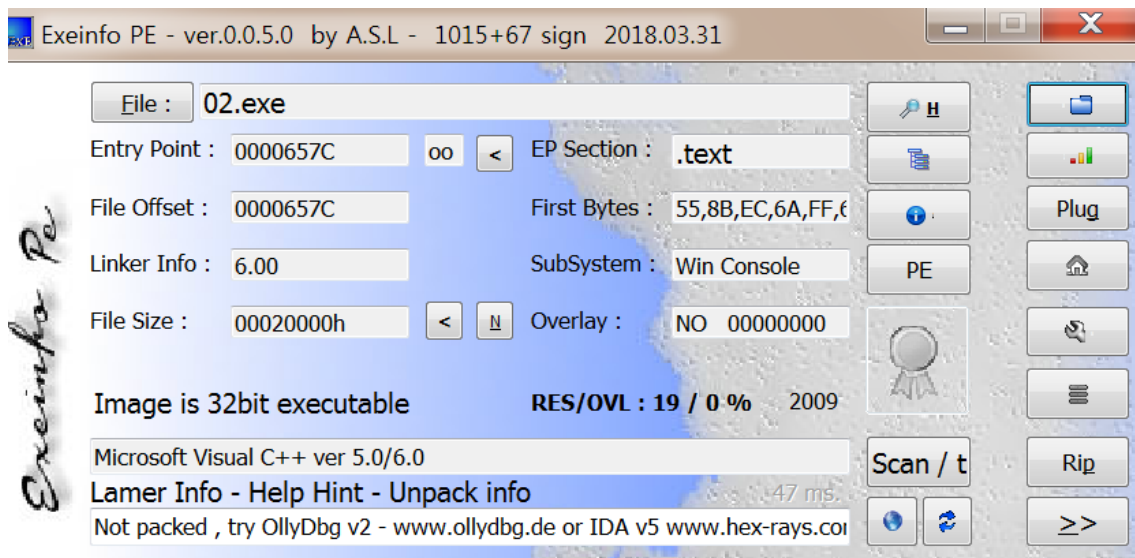
코드 엔진 Challenges: Advance 02

Author: Noble
Korean: 정답은 무엇인가

문제에는 따로 힌트가 없다. 파일을 실행 해 보고 어떻게 해볼지 생각해보자.



프로그램은 password를 입력하게 되어있고 패스워드에 9를 입력하고 엔터를 누르자 메시지 박스 없이 종료되었다. PEID를 통해 프로그램을 열어 패킹여부를 알아보자.



따로 패킹되어있거나 특이 사항은 보이지 않는다. 올리디버거로 파일을 열어 분석해보자.
사용된 함수목록을 봤을때도 특별한 함수는 보이지 않아 문자열 검색을 통해 실행 창에서 본
“Enter Passwod” 문자열을 찾아 이동해보았다.

004012B5	. 68 30424100	PUSH 02.00414230	ASCII "Enter Password: "
004012BA	. 68 40844100	PUSH 02.00418440	
004012BF	. AA	STOS BYTE PTR ES:[EDI]	
004012C0	. E8 BB0A0000	CALL 02.00401D80	
004012C5	. 8D8C24 F8030000	LEA ECX,DWORD PTR SS:[ESP+3F8]	
004012CC	. 51	PUSH ECX	
004012CD	. 68 D0844100	PUSH 02.004184D0	
004012D2	. E8 390D0000	CALL 02.00402D10	
004012D7	. 83C4 10	ADD ESP,10	
004012DA	. 8D9424 88070000	LEA EDX,DWORD PTR SS:[ESP+788]	
004012E1	. 68 D0144000	PUSH 02.00401400	
004012E6	. 68 40144000	PUSH 02.00401440	
004012EB	. 6A 64	PUSH 64	
004012ED	. 6A 10	PUSH 10	
004012EF	. 52	PUSH EDX	
004012F0	. E8 B0460000	CALL 02.004059A5	
004012F5	. C78424 D00D0000	MOV DWORD PTR SS:[ESP+DD0],0	
00401300	. 8D9C24 8C070000	LEA EBX,DWORD PTR SS:[ESP+78C]	
00401307	. C74424 10 64000000	MOV DWORD PTR SS:[ESP+10],64	
0040130F	> 8DBC24 F0030000	LEA EDI,DWORD PTR SS:[ESP+3F0]	
00401316	. 83C9 FF	OR ECX,FFFFFFFF	
00401319	. 33C0	XOR EAX,EAX	
0040131B	. 6A 01	PUSH 1	

004012B5로 이동해서 한줄씩 실행하다보니 004012D2에서 사용자에게서 값을 입력받는 다는 것을 알았다. 계속 코드를 실행해보자. 코드에 따라 64번 분기를 한 후 EAX의 값이 0이 되어서 분기문을 빠져나온 후 계속 코드를 실행하다가 004013C5 다음 구문에서 RETN되는 것을 확인했다. 이 함수의 내부를 확인해보자.

0012F75C	55	PUSH EBP	
0012F75D	8BEC	MOV EBP,ESP	
0012F75F	81EC E4000000	SUB ESP,0E4	
0012F765	59	PUSH EBX	
0012F766	56	PUSH ESI	
0012F767	57	PUSH EDI	
0012F768	8DBD 1CFFFFFF	LEA EDI,DWORD PTR SS:[EBP-E4]	
0012F76E	B9 39000000	MOV ECX,39	
0012F773	B8 CCCCCCCC	MOV EAX,CCCCCCCC	
0012F778	F3:AB	REP STOS DWORD PTR ES:[EDI]	
0012F77A	A1 08604000	MOV EAX,DWORD PTR DS:[406008]	
0012F77F	33C5	XOR EAX,EBP	
0012F781	8945 FC	MOV DWORD PTR SS:[EBP-4],EAX	
0012F784	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F787	0FBE08	MOVSB ECX,BYTE PTR DS:[EAX]	
0012F78A	83F9 43	CMP ECX,43	
0012F78D	0F85 F7000000	JNZ 0012F88A	
0012F793	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F796	0FBE48 01	MOVSB ECX,BYTE PTR DS:[EAX+1]	
0012F79A	83F9 52	CMP ECX,52	
0012F79D	0F85 E7000000	JNZ 0012F88A	
0012F7A3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7A6	0FBE48 02	MOVSB ECX,BYTE PTR DS:[EAX+2]	
0012F7AA	83F9 41	CMP ECX,41	
0012F7AD	0F85 D7000000	JNZ 0012F88A	
0012F7B3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7B6	0FBE48 03	MOVSB ECX,BYTE PTR DS:[EAX+3]	
0012F7BA	83F9 41	CMP ECX,41	
0012F7BD	0F85 C7000000	JNZ 0012F88A	
0012F7C3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7C6	0FBE48 04	MOVSB ECX,BYTE PTR DS:[EAX+4]	
0012F7CA	83F9 41	CMP ECX,41	
0012F7CD	0F85 B7000000	JNZ 0012F88A	
0012F7D3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7D6	0FBE48 05	MOVSB ECX,BYTE PTR DS:[EAX+5]	
0012F7DA	83F9 43	CMP ECX,43	
0012F7DD	0F85 A7000000	JNZ 0012F88A	
0012F7E3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	

0012F78A	83F9 43	CMP ECX,43	
0012F78D	0F85 F7000000	JNZ 0012F88A	
0012F793	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F796	0FBE48 01	MOVSB ECX,BYTE PTR DS:[EAX+1]	
0012F79A	83F9 52	CMP ECX,52	
0012F79D	0F85 E7000000	JNZ 0012F88A	
0012F7A3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7A6	0FBE48 02	MOVSB ECX,BYTE PTR DS:[EAX+2]	
0012F7AA	83F9 41	CMP ECX,41	
0012F7AD	0F85 D7000000	JNZ 0012F88A	
0012F7B3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7B6	0FBE48 03	MOVSB ECX,BYTE PTR DS:[EAX+3]	
0012F7BA	83F9 41	CMP ECX,41	
0012F7BD	0F85 C7000000	JNZ 0012F88A	
0012F7C3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7C6	0FBE48 04	MOVSB ECX,BYTE PTR DS:[EAX+4]	
0012F7CA	83F9 41	CMP ECX,41	
0012F7CD	0F85 B7000000	JNZ 0012F88A	
0012F7D3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7D6	0FBE48 05	MOVSB ECX,BYTE PTR DS:[EAX+5]	
0012F7DA	83F9 43	CMP ECX,43	
0012F7DD	0F85 A7000000	JNZ 0012F88A	
0012F7E3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	

코드를 보면 알 수 있듯이 CMP를 이용해 ECX와 특 정 값을 비교하고 있다 .CMP 부분으로 가고 ECX에 어떤 값이 들어가는지 확인해보자.

※일단 0012F75C의 EAX에 우리가 입력한 값인 10이 아스키 값으로 들어가 있다.

```
Registers (FPU)
EAX 00362729 ASCII "10"
ECX 0012F19C
EDX 0012F75C
EBX 0012FF44
ESP 0012F16C
EBP 00363783
ESI 0012FB90
EDI 00000000
EIP 0012F75C
```

0012F75C	55	PUSH EBP	0012F75D	8BEC	MOV EBP, ESP	Registers (FPU)	EAX 00362729	ASCII "10"
0012F75F	81EC E4000000	SUB ESP, 0E4	0012F765	53	PUSH EBX	ECX 00000000	EDX 0012F75C	
0012F766	56	PUSH ESI	0012F767	57	PUSH EDI	EBX 0012F744	ESP 0012F748	
0012F768	0B0D 1CFFFFFF	LEA EDI, DWORD PTR SS:[EBP-E4]	0012F76E	B9 39000000	MOV ECX, 39	EBP 0012F168	ESI 0012FB90	
0012F770	08 CCCCCCCC	MOV EAX, CCCCCCCC	0012F773	08 CCCCCCCC	MOV EAX, CCCCCCCC	EDI 0012F168	EIP 0012F787	
0012F778	F3:0B	REP STOS DWORD PTR ES:[EDI]	0012F77A	A1 0B604000	MOV EAX, DWORD PTR DS:[4060001]			
0012F77F	33C5	XOR EAX, EBP	0012F781	8945 FC	MOV DWORD PTR SS:[EBP-4], EAX			
0012F781	8945 FC	MOV DWORD PTR SS:[EBP-4], EAX	0012F784	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
0012F787	0FBEC0	MOVSK ECX, BYTE PTR DS:[EAX]	0012F788	83F9 43	CMP ECX, 43			
0012F78B	0FB5 F7000000	JNZ 0012F800	0012F793	8B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
0012F796	0FBEC48 01	MOVSK ECX, BYTE PTR DS:[EAX-1]	0012F79A	83F9 52	CMP ECX, 52			
0012F79D	0FB5 F7000000	JNZ 0012F800	0012F7A3	8B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
0012F7A6	0FBEC48 02	MOVSK ECX, BYTE PTR DS:[EAX-2]	0012F7A9	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
DS:1003627291-31 ('1')			0012F7AC	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
ECX:00000000			0012F7AF	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7B2	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7B5	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7B8	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7BB	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7BE	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7C1	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7C4	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7C7	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7CA	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7CD	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7D0	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7D3	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7D6	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7D9	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7DC	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7DF	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7E2	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7E5	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7E8	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7EB	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7EE	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7F1	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7F4	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7F7	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7FA	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F7FD	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F800	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F803	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F806	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F809	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F80C	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F80F	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F812	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F815	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F818	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F81B	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F81E	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F821	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			
			0012F824	0B45 0C	MOV EAX, DWORD PTR SS:[EBP-C]			

ECX 에 어떤 값이 들어 가는지 확인하기 위해서 CMP위의 부분은 분석해보니 EAX의 1바이트 값에는 31 즉 우리가 입력한 값의 첫째자리인 '1'이 저장되어 있는 것을 알 수 있다. cmp를 보면 31과 43 이랑 비교해서 ZF가 0이면 JNZ에 의해 18FF8A로 이동하게 된다. ZF 0이라 0012F88A로 이동하게 된다. 18FF8A로 이동하게되면 ECX를 호출하게 돼서 프로그램이 종료된다.

```

0012F834 E8 83030000 CALL 0012FBBC
0012F839 C745 F4 01000000 MOV DWORD PTR SS:[EBP-C1],1
0012F840 C645 E0 57 MOV BYTE PTR SS:[EBP-20],57
0012F844 C645 E1 45 MOV BYTE PTR SS:[EBP-1F],45
0012F848 C645 E2 4C MOV BYTE PTR SS:[EBP-1E],4C
0012F84C C645 E3 4C MOV BYTE PTR SS:[EBP-1D],4C
0012F850 C645 E4 20 MOV BYTE PTR SS:[EBP-1C],20
0012F854 C645 E5 44 MOV BYTE PTR SS:[EBP-1B],44
0012F858 C645 E6 4F MOV BYTE PTR SS:[EBP-1A],4F
0012F85C C645 E7 4E MOV BYTE PTR SS:[EBP-19],4E
0012F860 C645 E8 45 MOV BYTE PTR SS:[EBP-18],45
0012F864 C645 E9 21 MOV BYTE PTR SS:[EBP-17],21
0012F868 33C0 XOR EAX,EAX
0012F86A 8845 EA MOV BYTE PTR SS:[EBP-16],AL
0012F86D 8BF4 MOV ESI,ESP
0012F86F 60 00 PUSH 0

```

코드를 위로 올려보면 위와 같은 코드가 보이는데 57 45 4c 4c 20 44 4f 4e 45 21을 차례대로 넣는데 아스키 코드로 확인해보면 WELL DONE! 이라는 것을 알 수 있다. 이 코드를 실행해 볼 필요가 있다. 이 코드는 0018F839 로 시작하는데 이 주소는

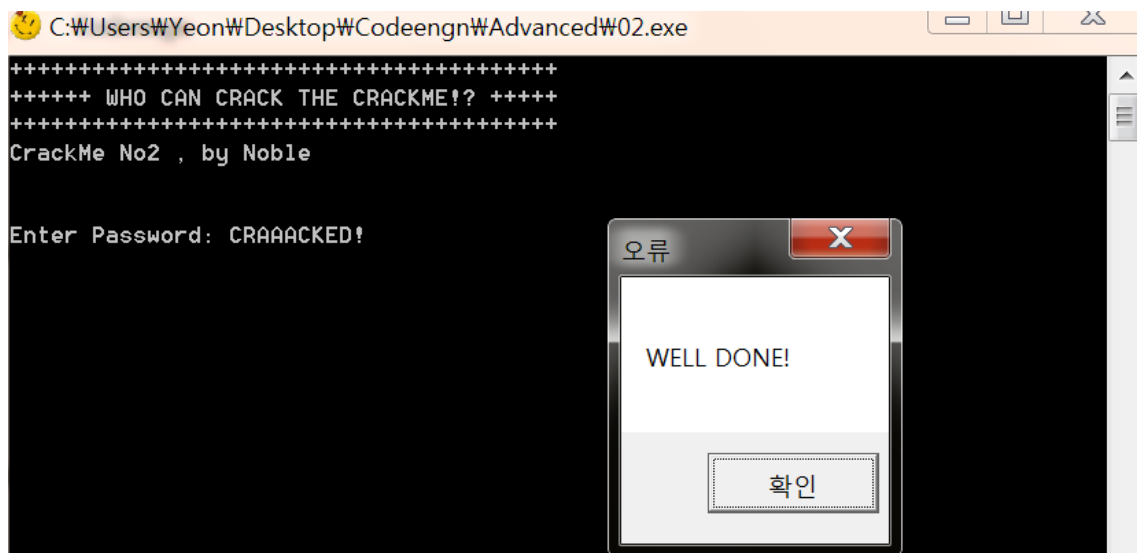
0012F7EA	83F9 4B	CMP ECX,4B	
0012F7ED	0F85 97000000	JNZ 0012F88A	
0012F7F3	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F7F6	0FBE48 07	MOVSK ECX,BYTE PTR DS:[EAX+7]	
0012F7FA	83F9 45	CMP ECX,45	
0012F7FD	0F85 87000000	JNZ 0012F88A	
0012F803	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F806	0FBE48 08	MOVSK ECX,BYTE PTR DS:[EAX+8]	
0012F80A	83F9 44	CMP ECX,44	
0012F80D	75 7B	JNZ SHORT 0012F88A	
0012F80F	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F812	0FBE48 09	MOVSK ECX,BYTE PTR DS:[EAX+9]	
0012F816	83F9 21	CMP ECX,21	
0012F819	75 6F	JNZ SHORT 0012F88A	
0012F81B	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	
0012F81E	0FBE48 0A	MOVSK ECX,BYTE PTR DS:[EAX+A]	
0012F822	85C9	TEST ECX,ECX	
0012F824	74 13	JE SHORT 0012F839	
0012F826	8B45 0C	MOV EAX,DWORD PTR SS:[EBP+C]	

0012F824에서 TEST ECX,ECX가 0일 경우 0012F839로 이동하는 것을 알 수 있다. 이렇게 이동하기 위해서는 위에서 점프하지 않도록 CMP문을 해결해야만한다.

첫 번째 CMP는 ECX와 43을 비교한다. 이때 JNZ에서 분기하지 않기위해서 두 값이 같아야한다. 그럼 ECX의 값을 CMP문마다 다르게 바꿔준다.

즉 43 52 41 41 41 45 4B 45 44 21을 아스키 값으로 바꿔주는 것이다.

아스키 값으로 변경하면 CRAAAACKED!가 된다 .이를 입력해보자.



CRACK이 되는 것을 알 수 있다.

