02.Keyfiling the reverseme + assembler

2012년 1월 16일 월요일

오전 6:15

**Lena Reversing**



02.Keyfiling the reverseme + assembler

번역자 : re4lfl0w / re4lfl0w@gmail.com

Last updated: 2012.02.22

The creator of this movie or the ISP(s) hosting this movie or any content in this movie take no responsibility for the way you use the information provided in this movie.

Movie 만든 사람이나 ISP에 이 movie를 hosting하거나 이 movie 속의 어떤 content에서도 제공된 정보를 사용하는 방식에 대해 책임지지 않습니다.

These file and anything else on this site and in this movie are here for private purposes only and should not be downloaded or viewed whatsoever!

이 site에서 file들과 모든 것들은 오직 개인적인 목적으로 download 하거나 볼 수 없습니다.

If you are affiliated with any government, or Anti-Piracy group or any other related group or were formally a worker of one you cannot enter this web site nor see this movie,

당신이 어떤 정부 또는 불법 복제 방지 그룹 또는 기타 관련 단체와 제휴하거나 공식적으로 하나의 노동자로 일할 경우 이 web site에 접속할 수 없습니다. 또한 이 movie를 볼 수 없습니다.

cannot access any of its files and you cannot view any of the HTML files.

어떤 file에도 접속할 수 없습니다. 어떤 HTML file도 볼 수 없습니다.

All the objects on this site and in this movie are private property and are not meant for viewing or any other purposes other then bandwidth space.

이 movie 안의 Site에 있는 모든 object와 개인적인 재산이다. 다른 목적이나 대역폭을 의미하지 않는다.

Do not see this movie whatsoever!

이 movie를 참조하지 마십시오.

If you see this movie you are not agreeing to these terms and you are violating code 431.322.12 of the Internet Privacy Act singed by Bill Clinton in 1995 and that means that you can't threaten our ISP(s) or any person(s) or company storing these file or these movies, cannot prosecute any person(s) affiliated with this page and movie which includes family, friends or individuals who run or enter this site or see this movie.

이 movie를 볼 경우 이 약관에 동의하지 않습니다. 그리고 당신은 1995년 빌 클린턴의 서명이 인터넷 개인 정보 보호법의 코드 431.322.12을 위반하고 당신이 우리의 ISP들 또는 사람들을 위협할 수 없다는 것을 의미합니다. 또는 이러한 파일이나 이러한 movie를 실행하고 저장하는 회사, 가족, 친구거나 사이트에 들어오거나 영화를 볼 개인을 포함하거나 페이지 및 영화와 제휴한 다른사람들을 기소할 수 없습니다.

*번역 주)이 부분 번역이 제일 어려웠습니다. 이해도 잘 안되고.*

Hello everybody.

안녕 모두들.

Welcome to this Part2 in my series about reversing for newbies/beginners.

나의 reversing 초보자를 위한 series Part 2에 온 것을 환영해.

This "saga" is intended for complete starters in reversing, even for those without any programming experience at all.

이 saga의 대상은 reversing에서 Programming 경험조차 없는 완벽한 초보자다.

Set your screen resolution to 1152\*864 and press F11 to see the movie full screen !!!

Again, I have made this movie interactive.

You screen 해상도를 1152\*864로 설정해 그리고 full screen으로 movie를 보기 위해 F11를 눌러

So, if you are a fast reader and you want to continue to the next screen, just click here on this invisible hotspot. You don't see it, but it IS there on text screens.

그래서, 네가 이것을 빨리 읽고 다음 screen을 보고 싶다면, 보이는 hotspot 여기를 눌러. 보고 싶지 않을 때는 여기에 두지마.

Then the movie will skip the text and continue with the next screen.

Movie는 text와 다음 screen을 skip할 수 있다.

If something is not clear or goes too fast, you can always use the control buttons and the slider below on this screen.

무언가 명확하지 않거나 빨리 넘기고자 할 때, 항상 control button과 이 screen 밑에 있는 slider 바를 사용해.

He, try it out and click on the hotspot to skip this text and to go to the next screen now!!!

도전해봐. 그리고 이 text와 다음 screen을 보기 위해 hotspot을 click해.

During the whole movie you can click this spot to leave immediately

이 movie 어디에서나 즉시 떠나기 위해 이 spot을 click 할 수 있다.

Click here as soon as you finished reading(on each screen !)

네가 읽기가 끝났을 때 이 곳을 클릭해. (이번 screen에서)

1. **Tools and Target**

Throughout most of these tutorials, you will use Ollydebug and… your brain. The first can be obtained for free at

대부분의 tutorial을 통해, 너는 Olly와 너의 두뇌를 사용할 것이다.

첫번째로 여기에서 무료로 얻을 수 있다.

<http://www.ollydbg.de>

...again, the second is your responsibility ;)

2번째는 너의 책임감이 필요하다.

For your convenience, I included the target (ReverseMe) in this package. In this package is also included a file "Keyfile.dat".

너의 편리를 위해, 이 package에 ReverseMe를 포함했다. 또한 "Keyfile.dat"도 포함되어 있다.

Do not extract it to the same directory as the ReverseMe if you want to try this tutorial yourself. (Practice is the best way to learn).

이 tutorial을 혼자서 시도할 때 같은 디렉토리에 풀 수 없다. (연습이 배울 때 가장 좋다)

1. **The use of plugins**

Study of the target is always extremely important. However, we did that for this target already in Part 1, so, I'll refer you to Part 1 for it. In fact, this tutorial will build some things on Part 1 and if you have not already done so, I strongly suggest to see that tutorial first ;))

공부 목표가 정말로 중요하다. 그러나, 우리는 이미 Part1에서 목표물을 정했다. 그래서 나는 Part 1을 참조하라고 얘기한다. 사실, 이 tutorial은 Part1에서 무엇을 만든다. 그리고 네가 먼저 보지 않았다면, 강력히 먼저 볼 것을 추천한다.

Keep your mouse pointer here and click whenever you are ready reading on each textscreen

네가 다음 textscreen을 읽고자 할 때 너의 마우스를 여기에 위치하고 클릭 해.

When loading a target, Olly will automatically analyze the code to comment the code for us. (Not in certain conditions, see later Parts)

If you doubt that Olly has effectively done it, you can always make Olly do that like this….

Target이 loading 될 때, Olly는 자동으로 code에 comment를 달아 해석한다.(이것은 명확한 조건은 아니다, 나중 Parts를 봐라)

Olly가 효과적으로 완료했는지 의심이 든다면, 이것처럼 항상 Olly가 할 수 있게 만든다.

시스템 생성 대체 텍스트: '''·｀'''L
bbbbbL
22222U
쵸J3쵸구쵸규3H
3B33B-U
2뱀ㅗ2즙‘3-D
騁ee觸脯畛베
ESCSSSDsFS騙
e뱀‘白U·욥白paUnpaO
CPpZSTDO
1100。R>
.RB],E얌《
∥죠琿dr
l'LOaalcon
l『R=r■－rl〕r.l'
- ．무·
臘LL
EIPoe4OIOOer.ver'eh.<h。du【．Ent펴汗
이FFFFFFFF)
e(FFFFFFFF)
0(FFFFFFFF〕
'Cur
.RFJ
IDC-nRRO비
》
0(FFFFFFFF〕
7FFDDOOO(FFF】
"ULL
DOHㄴV:HIDOEN:S무STE":RF
TIH6
EFL
LastErrERR0f큅no〔HOT一FO나江）(E
00000246(NO,HB,E,BE,NS,PE,6E,L
ehpty-UHOR"BBSOeiogelO4OOOE
ehpt,0.0
enpt닛0.0
e"ptye.8
enpt,e.8
enptyO.O
enptyl.000eee0000000000e0e
enot딪1.00000Oeoee800000000
e.'234567
TTTTTTTT
55555555
톤e홑헝．
Lr *
1010
.Lrr.｀■．
P·‘l
SaO
EaU
k
r5
ra
En
aUaU3
5
ㅓ‘ao '
R
20R
E
3뱁tN
괘UC
n샅
Or
CP
。a一
litP
sac知p
C。py
Binary
ASSefnble
Lab님
COmln온nt
Brea너ort
HittraCe
RUntraC횬
SPace
_SHRR
RERD
灌멥
:Flㄴ
IC-－닒
아뛰있
ㅜE
【e.dat"
_RPPL廉刃R【．
【eR．난er맬황＂e"
OnOer효出OUtO「dat..
FST4020
FC닒027F
ar-
Sear小慽 직
Fndre딛rencesto
騙ew
COPytoexecuta벼e
AnafySIS
Heiponsw枕枕name Ct겨＋F1
Plugi『ISin이ly....,
/rrl.''I''
GOto
F하IowhDump
mewCaltree
And『lowcliCktoanalyseCodㅌ｀
Hㅇwever,1wanttoexPlainyㅇutheuseof
》
Anjyseα》de
Ctrl수＾
J미E솥【32.7C816「
10738
一- r＇■■■J
rr 什 · ㅡ 노r、琵
괘니 · ■ 一 -IJ'
a 什 · -·r·>
AnalyzeTh비
AsmZC．肉。ard
6O瞭ma楸
(jnscrpt
Rem。v히In래y轍S什예mod며e
Scanobjectf걜es
Rem。veabjectscarlf누帆m沁ul.
『 Du呻debug醜d床oceSS
AsSLjmearguments
Removean히yS恪什예s비ect沁n
Durlingnextan님ySIs,treatsele山OnaS
HChain
r
7C816D드S
26a딩apaD677e·
6仕Oap白OapC92aU·
Ma眈dumpofProcess
UltraStringF柚싣rence
勿pear齡Ce
ㄱ·

And now click to analyze code.

However, I want to explain you the use of plugins in Olly….

해석 code를 click 하자.

그러나, Olly에서 사용하기 좋은 plugins을 소개하고자 한다.

시스템 생성 대체 텍스트: ■‘ r
■■■■■■■■■■■■■■■
『니O『쨍
驪룁L
"OU
PUSㅐ
PUSH
CRLL
NOU
PUSH
PUSH
PUSㅐ
PUSH
PUSH
PUSH
PUSH
CRL,L
C『IP
JIㅓZ
PUSH
PUSㅐ
PUSH
PUSH
CnLL
CRLL
JllP
PUSH
PUSH
PUSH
PUSH
PUSH
CF겸LL
TEST
r급”^
<J뿌．咎USER32.니）
脯【RDPTRDS:[4e
간：‘〕l:l
O
adIOOnR>
21RB〕,ER욤
'황I놀' - l'L겯一
LOadICOnR
R$rchan.=
<JP.各USER92.LOadCUr
며奐ORDPTRDS:[4e21RF]
sackup
IDC-RRROU
'
m 硼 棘
대 sHTE da
ST-L힌―--―ㅑ
....an（」seethis"AnalyseThi힛l'Pluginthatl
Wi∥usehere·(OnlytoshㅇwyㅇuPlugins).
INFO:AnalyzeThi힛15an이lyDbgPluginto
al―ㅇw이lyDbg'sanalysisfunctiontooperate
이』tSideofthemarkedCodesegment.
ItCanbedㅇWnlㅇadedat:
httF)：〃tuts4you.com/request.phP구22
Unzipin이ly'5Pluginsdirectoryandthenit
CanbeaCCe
ssedbyrightcl
iCkinginthecode
＿一一一一一一一一一一一一一一」』 _
An레yS'
喚b。ns畑鞠놈cnafy굳
An卵”〕諫쩌탭■■■■■
Ctd춈Fl
'- -F7Fn
-^蹄嫩僉낱
AsrnZC너沁ard
BO瞭ma楸
(msα盼t
山呻de恥g야ged床Ocess
Makedump아Process
UltraStrlngRef난ence
eo12FF
oe1ZFF
001ZFF
00!ZFF
oe1ZFF
ee!ZFF
0elZFF
001ZFF
001ZFF
0012FF
qnl츨「「
·예겨rr媤rT＂쨉‘딕＂뼙긋읽겨n니TJ
76勺召6n긴ao白O677aO·
296aon킵nonpC920·
7662enO06670·
朽걔6F觸凶觸卵69秘6F맷
7661泌柏ee騁的66邵馹옛
rDrJ41'eap09C0e·
6776enp0762e·
恪‘Cr구8e0仕디rDgEe·
r급676e00662e·
沈616F6S邵騁卵4B6664맷
朽76泌72松腑騁紛衿69앳
CrO4.pEapo白Ur댜CaO·
64676aonp066。O·
gdOFerpnonpdUBI쵸O■
…·…
……
勿pear齡ce

...and see this "analyze This!" plugin that I will use here. (Only to show you plugins).

그리고 "analyze This!" plugin을 여기에서 사용할 것이다.(너에게만 보여주는 plugin)

INFO : Analyze This! Is an OllyDbg plugin to allow OllyDbg's analysis function to operate outside of the marked code segment.

It can be downloaded at :

<http://tuts4you.com/request.php?22>

Unzip in Olly's plugins directory and then it can be accessed by rightclicking in the code

정보 : Analyze This는 Ollydbg의 해석 함수가 표시된 code segment를 외부에서 조작할 수 있는 Ollydbg plugin 입니다.

여기에서 다운로드 가능하다.

<http://tuts4you.com/request.php?22>

Olly plugin directory에 풀고 code에서 rightclicking으로 접근할 수 있다.

Ok. In the blink of an eye, Olly has analyzed the code.

Ok. 눈을 깜빡이자, Olly는 code를 해석했다.

1. **Solving the ReverseMe**

See also Part 1 in this tutorial series but to understand everything fine, we will need to broaden your overall knowledge a little concerning bits - bytes - numbers - hex.

I'll try to keep it to the minimum that is required to understand this series though.

이 tutorial의 Part1을 봐라. 그러나 네가 이해가 됐다면 우리는 지식이 넓히는데 bits - bytes - numbers - hex를 집중할 수 있다.

나는 작게 유지하기 위해 이 series를 이해하는 것이 요구된다.

If you have an outstanding knowledge in this matter, it is clear that you can skip this piece. Let's go!

만약에 뛰어난 지식을 가진다면, 이것은 명확하다. 너는 이 조각을 넘길 수 있다.

INFO:

* The decimal number system.

The easiest way to understand bits is to compare them to something known: digits. A digit is a single place that can hold numerical values between 0 and 9 in our numeric decimal system.

가장 쉽게 bit를 이해할 수 있는 방법은 숫자들을 비교하는 것이다. Digit는 한 자리다. 그것은 0과 9 사이의 10진수 system이다.

Digits are normally combined together in groups to create larger numbers.

For example, 6357 is a decimal number and has four digits.

숫자들은 보통 그룹 안에서 큰 숫자들을 만들 때 조합된다.

예를 들어, 6357은 10진수 system이고 4개의 숫자로 되어 있다.

That should all feel pretty comfortable -- we work with decimal digits every day. The neat thing about number systems is that there is nothing that forces you to have 10 different values in a digit.

매우 편안하다. -- 우리가 매일 10진수를 사용한다. 단정된 숫자 system은 아무것도 아니다. 숫자에서 10가지의 다른 값들을 가진다.

Out base-10 number system likely grew up because we have 10 fingers, but if we happened to evolve to have eight fingers instead, we would probably have a base-8 number system.

Base-10 진수는 성장했다. 왜냐하면 우리는 10개의 손가락을 가지기 때문이다. 그러나 만약에 우리가 8개의 손가락에서 진화했다면, 우리는 아마 8진수 system을 사용했을 것이다.

You can have base-anything number systems. In fact, there are lots of good reasons to use different bases in different situations.

너는 다른 number system을 가질 수 있다. 사실, 그것은 많은 다른 상황 속에서 다르게 사용하는 이유가 있다.

So, all this seems pretty simple, but how do we calculate in the decimal number system, I mean how do we assign values for numbers?

그래서, 매우 간단하다, 그러나 우리가 어떻게 10진수 system에서 계산을 해야되나? 어떻게 숫자 value 값을 할당하나?

INFO:

It is understood that in the number 6,357, the 7 is filling the 1s place, while the 5 is filling the 10s place, the 3 is filling the 100s place and the 6 is filling the 1,000s place.

6357은 7은 1의 자리, 5는 10의 자리, 3은 100의 자리, 6은 1000의 자리.

So you could express things this way if you wanted to be explicit:

네가 원하는 명쾌한 것이 있다면 표현할 수 있다.

(6 \* 1000) + (3 \* 100) + (5 \* 10) + (7 \* 1)

= 6000 + 300 + 50 + 7 = 6357

Another way to express it would be to use powers of 10. Assuming that we are going to represent the concept of "raised to the power of" with the "^" symbol (so "10 squared" is written as "10^2"), then another way to express it is like this:

다른 방법으로 표현할 수 있다. 그것은 10진법으로 사용할 수 있다. 생각하건데 우리는 "raised to the power of" 개념을 설명하기 위한 상징으로 "^"(그래서 "10 squared"는 "10^2"로 쓰여졌다), 다른 방법으로 이것처럼 표현할 수 있다.

(6 \* 10^3) + (3 \* 10^2) + (5 \* 10^1) + (7 \* 10^0) = 6,357

6000 + 300 + 50 + 7 = 6357

What you can see from this expression is that each digit is a placeholder for the next higher power of 10, starting in the first digit with 10 raised to the power of zero.

이 표현식에서 너는 볼 수 있다. 각 숫자들은 다음 10진수를 위해 자리를 지킨다. 처음 시작하는 10의 진수는 0이다.

Computers happen to operate using the base-2 number system, also known as the binary number system (just like the base-10 number system is known as the decimal number system). Let's find out why and how that works…

Computer는 오직 2진수에서만 작동한다, binary number system(10진수 시스템은 decimal number system)을 알고 있다. 자, 왜 사용하는지 어떻게 사용하는지 찾아보자

INFO:

* The base-2 System and the 8-bit Byte.
* 2진수 and 8byt Byte

The reason computers use the base-2 system is because it makes it a lot easier to implement them with current electronic technology. So computers use binary numbers, and therefore use binary digits in place of decimal digits. The world bit is a shortening of the words "Binary digIT."

컴퓨터가 2진수를 사용하는 이유는 쉽게 시행할 수 있고 전기적인 기술이기 때문이다. 그래서 컴퓨터는 2진수를 사용하고 2진수를 10진수 자리에 사용한다.

세계적으로 bit는 "Binary digIT"의 약어다.

Whereas decimal digits have 10 possible values ranging from 0 to 9, bits have only two possible values: 0 and 1. Therefore, a binary number is composed of only 0s and 1s, like this: 1011. How do you figure out what the value of the binary number 1011 is?

대조적으로 10진수는 0~9까지 10개의 값을 가진다, bit는 2가지 값을 가진다:0과 1. 그런 까닭으로 2진수는 오직 0과 1로 되어 있다. 이것처럼 : 1011. 어떻게 2진수 1011은 계산 할 것인가?

You do it in the same way as for 6357, but you use a base of 2 instead of a base of 10.

너는 6357를 할 때와 같은 방법으로 할 수 있다. 그러나 10진수 대신에 2진수를 사용해야 된다.

(1 \* 2^3) + (0 \* 2^2) + (1 \* 2^1) + (1 \* 2^0) = 11

= 8 + 0 + 2 + 1 = 11

You can see that in binary numbers, each bit holds the value of increasing powers of 2.

너는 2진수에서 볼 수 있다. 각 bit는 증가하는 2진수의 값을 가진다.

That makes counting in binary pretty easy.

Starting at zero and going through 20, counting in decimal and binary looks like this:

2진수에서 쉽게 셀 수 있다.

0에서 시작하고 20까지, 계산한 10진수와 2진수를 봐라. 이렇게

INFO

0 = 0

1 = 1

2 = 10

3 = 11

4 = 100

5 = 101

6 = 110

7 = 111

8=1000

9=1001

10=1010

11=1011

12=1100

13=1101

14=1110

15=1111

16=10000

17=10001

18=10010

19=10011

20=10100

INFO

Bits are rarely seen alone in computers. They are almost always bundled together into 8-bit collections, and these collections are called bytes.

컴퓨터 분야에서 드물게 쓰인다. 그들은 거의 항상 8bit 꾸러미로 다닌다. 그리고 그것들의 꾸러미는 1byte로 불린다.

Why are there 8 bits in a byte? Well, a similar question is, "Why are there 12 eggs in a dozen?" The 8-bit byte is something that people settled on through trial and error over the past 50 years.  
왜 8bit를 1byte라고 부르냐? 이것은 이 질문과 같다. "왜 12개의 계란을 한 다스라고 부르냐?" 8bit는 사람들이 지난 50년간 시도와 에러를 통하여 합의를 봤다.

With 8 bits in a byte, you can represent 256 values ranging from 0 to 255, as shown here:

8bit는 1byte다. 너는 256가지의 값의 범위로 0~255까지 표현할 수 있다. 보이는 것과 같이

0 = 0000 0000

1 = 0000 0001

2 = 0000 0010

…

…

254 = 1111 1110

255 = 1111 1111

INFO

Bytes values are also used to hold individual characters in the ASCII character set, each binary value between 0 and 127 is given a specific character.

Bytes 값은 ASCII에서 각각 문자로 사용된다. 각 binary value는 0~127까지 특별한 문자로 구성되어져 있다.

Most computers extend the ASCII character set to use the full range of 256 characters available in a byte. The upper 128 characters handle special things like accented characters from common foreign languages. See included file called Dec2Hex2Ascii.htm

대부분의 컴퓨터는 ASCII를 확장했다. Byte에서 256 문자를 사용할 수 있게 됐다. 상위 128 문자는 공통적인 외국어에서 accent 문자같이 특별한 것들을 조종할 수 있다. 포함되어 있는 파일인 Dec2Hex2Ascii.htm을 봐라.

But I will discuss this further on too.

그러나 나는 나중에 이것을 다시 의논한다.

INFO

* The Hexadecimal Number Base System
* 16진수 System

Now, the next step -assuming you understood the previous- is rather easy to make: from the binary to the hexadecimal system.

이제, 이전 강의를 이해한 것으로 가정할 때 다음 step으로 binary에서 hexadecimal 으로 바꾸는 게 오히려 더 쉽다.

A big problem with the binary system is verbosity. To represent the value 202, it requires eight binary digits.

binary system의 가장 큰 문제는 장황하다. 202 값을 표현하려면, 8자리의 binary 숫자가 필요하다.

*번역 주)Binary: 11001010(8자리), Dec: 202(3자리), Hex: CA(2자리)*

The decimal version requires only three decimal digits and, thus, represents numbers much more compactly than does the binary numbering system.

10진법은 오직 10진수 3자리만 필요하고 2진수 system보다 표현하는데 간편하다.

This fact was not lost on the engineers who designed binary computer systems. When dealing with large values, binary numbers quickly become too unwieldy.

사실 2진수 computer system을 개발한 engineer들은 잃은 게 없다. Large 값을 처리할 때, 2진수는 빠르게 늘어난다.

The hexadecimal (base 16) numbering system solves these problems.

Hexadecimal numbers offer the two features:

16진수 system은 이 문제들을 해결한다.

16진수는 2가지 사실을 제공한다.

* Hex numbers are very compact
* It is easy to convert from hex to binary and binary to hex.
* 16진수는 매우 간편하다.
* 그것은 16 -> 2, 2 -> 16으로 변경하는데 쉽다.

Now, all the previously explained concerning the calculation for the value represented by the binary/decimal notation equally works for this hexadecimal system.

이제, 모든 것들을 설명하기 전에 값들을 표현하기 위해 계산하는데 집중해라. 2진수/16진수 표기법은 16진수 system을 위해 똑같은 일을 한다.

If you like, you can try it out for yourself(the square value calc)

네가 좋아한다면, 너는 네 자신을 위해 도전할 수 있다.(실효값 계산)

INFO

Since we'll need to enter hexadecimal numbers into the computer system, we'll need a different mechanism for representing hexadecimal numbers to display their associated value.

우리가 컴퓨터의 16진수를 사용하기 위해서는 필요할 것이다, 관련된 16진수를 표현하기 위해 다른 mechanism이 필요하다.

In Assembly Language programming, most assemblers require the first digit of a hexadecimal number to be 0, and we place an H(h) at the end of the number to denote the number base.

Assembly programming에서, 대부분 assembler들은 16진수의 첫번째 자리가 0이 되는 것이 필요하다. 그리고 우리는 h를 숫자의 마지막에 number base를 표현하기 위해 붙인다.

To denote the difference with binary, we place a B(b) behind the number and a D (d) for the decimal number.

2진수를 다른 것과 구분하기 위해, 숫자 뒤에 b를 붙이고 Decimal 숫자 뒤에는 d를 붙인다.

The Hexadecimal Number System:

16진수 system

* Uses base 16
* Includes only the digits 0 through 9 and the letters A, B, C, D, E, and F
* 16진수를 사용한다.
* 오직 0~9와 A,B,C,D,E,F 문자를 사용한다.

A hex number example is 1C9EB7F3h

16진수 예로 1C9EB7F3h가 있다.

It suffices to know these basics and to be able to calculate them to the decimal system and vice versa using windows calculator(scientific in hex) (Example is 480163827d)

이것은 기본을 알고 windows 계산기를(16진수에서) 사용하여 10진수로 계산이 가능하고 반대도 할 수 있다.

*번역 주) 즉 쉽게 이야기해서 16진수 -> 10진수, 10진수 -> 16진수 가능하다는 이야기*

(예를 들어, 480163827d)

So far our little detour in bits, bytes, binary, decimal and hex. But we were going to solve the ReverseMe, remember? Let's continue and step over(F8) to see what goes on.

지금까지 bit, bytes, binary, decimal and hex를 약간 우회한다. 그러나 우리는 ReverseMe를 해결하기 위해 왔다 기억나? 시작하자. 그리고 F8로 무슨 일이 일어나는지 보자.

시스템 생성 대체 텍스트: －룝－
一＊--
.CP【j-颱柚thread,飇duler슨yerseㅆ
5 6R00
E：〕E.4t:l요ol:tl〕l:l
Rsr·솝1ㅓ이콥e
1:70997요1‘홑O
L.rlㅇdl.te=HULㄴ
G슨fh。d.I【eH놀『．d【ER
··l·논눕乙밈孔튕
·U·ㅂ白긷0
드크旦르L柏츠O
5닌「‘1ㅓCCO
드H걋급1ㅓ000
77n4OeO
솥‘∼OL- -닙一，b一n밉nH4q긍
=UewedFr드les수습恪즐측＝.츱’j
沈泌茁ㅡ坤mD댜沁D
4e塑袒塑牝
ee騁開－璽盼
청＇1·：l'l
0(140
0040
0040
0040
갯
rU
·LCI--II'LI'lL
百4百I03E-
q7aL松끄D
04
ㅌ：FO크0000
R욥纛4이）C
1:11:17「0이〕l:1
OO
；二〕OZL:I000
RF옴lㅓ000
肉肉
恪「요1극OOC
03
0〔l
03
00〔：I〔：1001:l:l
7징한松DD
eBeZ0000
「8「「
!D
00
00a04Oa0
17204000
ee
D7O20eOO
24020000
83e1OOeo
00
73a14000
46
lF槍1400e
F':-I개∥ :
이LL<JhP.'KEf枕IEL32.6.t＂ㅇdu【．Hand【．R>
너:.1탠’CF【.T뒤DS:드ㅓeal?7그.EFI：、
"O；、Icll.II:.F【FT尸DS:【4以드197〕.4008
n0UDllll〕RDPTRDS:〔4a므l·괏e】,rㅎ，er$eH.oe4OI긔
r.IiJ던[lI:＂교DF.TRDS:〔4eal‘굿F」,0
N〔11.j[lI．姃lf;:DPIRD·ㅌ；:t4l:l2'lR〕〕,0
no늰Eq(,DUORDPTRDS:【402177』
rIOllcllll0RDPTRDS:〔4이크IR7】,타了（
Pl_I3H4
P;_I3HE써＜
댜긔LI.久汁IP.各USER32.ㄴO.dIOㅇnn>
어〕I긔ICF'[rㅣ뒤〔〔．:하4p닌l，ㅓ二〕,하I표
F'l＿ㅣ카뀀7F01:l
F':_13니0
댜LL<J뿌．各USER32.L。．dCurt。rR>
"1:11.1DI..jl:IF극L.F·T궈DS:〔4eaIR=」.타겪
F;_I3HO
PUSHr솥Ver$e".0040216F
F.i」gH
PUSHO
F'lJ'i珊13
F'U'E；ㅐC아〕00인:．」
P＿ㅣ3ㅐr*uer'e".004e2079
[
R습r땁＂긔l:(=4.
h:n$t=0.:10100
Lo功」IconFI
[
I：딩rc∥」r.l'=lrll::_nPRO미
n드t=「IULL
LO」．」〔．.rsl=lf숙
狐編뛔
丕도
OFE
·Terㅣp【ateFLLe
4ttrlb』t슨르=
:r〕＿Lㄴ
REnLll:l더ㄴ’..':H:DD티ㅓ:S'l'3TEll;l-4ㅏ
터。爽=OP다I＿타汀STnㅓ＇一
DqC尸l』rl*’구
3harebㅇd료
111-11_l_
FIL.E_SHqREURERD:F'LE一SHHf.:f
굿＇
·'7
댜뵙ㅗ
CHP
J"Z
PUSㅐ
PUSH
PUSㅐ
PUSH
CnU.
CRLL
J"P
PUSH
PIJSㅏ夕
F’앙－I.B닝
<｀鸞IP.各KERt춈EL32.CreateFi【en)
E아＜.-i
하緞RTreVer를슨숴．oe48lOgR
O
HCCE돈돌
「.【E",=
이잖5궁돋Fi
aE디ERll:_RERD;GE『IEFIC닙IljRITE
''f’낟닛Flll광．d〕t''
【eR
r'V.r맬
상Ver'
.h.004e20e0
.＂츱OO4OZei7
St,ㅗ。=hB－아（Ih터一RPP나KnR【．
Tit盼="Ke,FileRㅎver∥he''
Te、t'맨EvaluatLon。eriod。ut。「dat*.
卜O니r.．〃（HULL
JP.욈USER32.he$'ageB애n>
JP.各KERHEL32.EㅒLtPrOC슬룔룔＞
ㅣ
蓉튠geBOㅐR
PrOCe톨쥴
r·너－r녹7노
饗쁘쁘쁘쁘쁘一一一一一一一一→／노스一一一
굿Ifwewouldcontinueexecuting
theCod타thenit15onthisline
thatthebadmessageappears.
=NULL
rev.rseH.oe4OZ173
45(7e.)
etseh.oe482ilR
0011
·＝-l츤Dㄴ5.
니－UCHE"
PPCㅜ·
ZFOS0000
C0
r노J텀
R·Ul:=:令p0C.RI'’·：.l!．걀Rsll·HCH:=：·:=:I89r규nH88CH889nensnOO무ro·
·bㄷ-ER·b·bEI-.L、b·b'b、b·b、b‘亡E97666ZOEEE6666r댜ES"
…눕椧e7De7F刪e8B09e脇09c驪驪臘
柵柵0040l冊쁑·l.l4Lli縕一一糊eO40l冊0040l溫0040l柵柵004el瀏∼
00401283=<J빼P.얹KERHEL32.CreateFL〔손R>

So far all good. Remember from Part 1 that here, the action takes place? Let me refresh your memory and let's go see the API again.

지금까지는 좋다. Part 1에서 이곳을 얻기 위한 액션이 기억나? 다시 새롭게 너의 기억에 넣어주마. 그리고 API를 다시 봐라.

If we would continue executing the code, then it is on this line that the bad message appears.

만약 우리가 code를 실행한다면 이 line이 bad message가 나타난다.

...unless of course we can avoid somehow this badboy…???

우리가 badboy를 피하지 않는 한

Rightclick

시스템 생성 대체 텍스트: 매h핏풀레이어'
ETCAreVersingtUtorialfornewbiesbylena151PartOZswf
.-ㅁ
놀
Paused
녈삥씌
－쨍r，「른
쐬꾀뾔」쐬」뾔뾔희희꾀工희꾀의끄끄희희스」희협劉쬐
_ FIJ코ljL一一一 ■■·■더·■떼
"TheCreateFilefunctioncreatesor
oper1Sa『lobjectandretumsa
handletoaccesstheobject''
Ogrdmmer'sReference
口回囹
kerl테adWUZer
Ind타《
ㅇ巾esHelp
I너I ■ 一
IJI · !
ㅡ너·
뱀1'
I너l
1닉
－너ㅡ
－너I
04
3F030000
RB21’홑000
O07Fe000
00
C8020000
PUSH
PUSH
댜놈LL
hOU
PUSH
PUSH
Enk 、
<J"P.'USER32.LOad!OOn
DUORDPTR05'【4021RBI,
7FOO
0
eFiI61d'uick..．。1여very.ew】ㅣ＿흐쁘〔＿]
C.LL《J뿌
.'U
PT
SERSZ.LOadCur룔
RDS:〔4021RF】,
蛾뇹createFile伽ctioncreatesorOPensthefollowingobjectsandreturnsahandle^
thatcan騁usedtoaccesstheo비ect
RFZ14aOO
ee
SFZ14000
03
ee
O3
OO00aeCO
79a04000
eB02BBOO
SFF
lD
00
00204000
17ae400e
00
며勵RO
U
r솥냐erse".004B216F
〔0DI:10000
什Ver$ell.00402079
(JP.各KEF꺼EL32.Create
E댜＜-一1
하緞요Trev●rttH.se40ieg
0
tever$eh.Oa402000
teVer'ell.Oe4eZO17
U
히es
pipes
maiISlotS
COmmunicatlonSreSOurCeS
diskde너ces（嘶ndowsNTonly)
COnSOIeS
directories（叩en。nly)
UND(O0e0000
,GE,LEI
00000000
JJ탬Jl뱁1땜I F
R8377,r713Rd,538R83R8RRR09883
6ECHCCCCgC텨6r큐ECH66ER6666666E8
7Eel6Be27CE3Lp79
규34444r()sr구5rr)6666.
UeesaodpeesnoaO8esnUL-l·
p白UnUn킹od긴n디dUnpnUnpnUnUdUn킵ㅡb·
一아e4아아O4아아아아아아여여아메
긷apoa깊0仕po色U白pd긴白뱁00色U白p－뎌·
D白p뱀‘
''e뱀‘
00뱀‘
Uel
Pl'뱁‘
noapsnU
。1114''꼬' :
'’너4「IL-5"8
004e107D
OO4e요07F
00401084
00401089
004e1e8B
00401e90
00401095
OO40109R
0040ie9C
004018RI
004eleR3
00401eR8
004010R9
004010nE
r놈r令』＇춈눼r놈「놀r노
D7e20eee
24e20000
83eioeeo
ee
73214000
46
IR214000
ZFe20eee
Ce
<J"P.各USER32."e'’술9·B
<JP.하＜E사IEL32.EㅐitPr
Thenameㅇfthefilethe
prㅇgramis―ㅇ改ingfㅇrll!
Let'5SeethatintheCOde
r랖p.rs.".0040i21D
.1
r뺨Verse".0e402173
4亡．
55,
rever꿸eH.004021iR
Ef枕
<JP.'KEI괍IEL32.Re솥dFL
E챠＜.EnX
H^NDLECreateFiIe(
LPCTSTRI죤F'IeName,'
DWORDm''[)eslre砂룅cCe
DWORDdWSha＇탕緘浴e,
.·‘··'h· r놀r놈』r닙.rb「딩
VSHsHsHSHSHSHsHLLPzsHSHSHSHLLLLPsHSHSHSHSHLLST、
恥PUPUPUPUPUPUPU매얘개PUPUPUPU釀8刪PUPUPUPUPUmTE·’I
花胡6868胡ESESEg胡68胡68騶Es邵f
…노Z … …
쑤꽤
004el283=<JP.各KE勵［L32.Creat손F료len)
LPSECURITY.-ATTRIBUTESIPSecu吻AttributeS,
DWORDd',,creatronDlstr心utton.
DWOR【）’斛RagsAndAttnbuteS,
HANDLEhTemPlafe戶Ie
);
〃peintertonameofthef
〃access(read-write)mo
〃Sharemode
〃pointertosecurityattril
〃how!0Create
〃悧eat!"butes
〃handleto5leWithsttrib
Rddr．맬＄
004020e0
004021:110
OO4e202O
0040203e
00402040
004020Se
004e2060
004e207e
00402080
004e209e
004020n0
ParameterS
《Il聃I
스二．
KeyfLI'
.dat.K',filei'
notva【Ld.SOrr
,··.......······
0012FFCC
00!ZFFDe
Oe12FFD4
nnl？「「ng
FFFFF「FF
7FFDE000
SO54R6EO
더더'츨「「n요
柏6F觸6E卵卵腑貂乙72엣
7269恥沈閔에盼6C6972맺
rD4FE댜oee98Fdp·
67ZO6eeapZO260·
7661沈乙的00卵66邵騶옛
rpr구4loeOgCOO·
6776apee762ao·
ZCr구seeerOgEe·
r규67600nUZO620·
泌616F臼邵卵에4B6664엣
尸구6eZa구eee99e·
67277ee276ao。
Cr구4r구EeOUerpCdO·
64676eoe660-
69沈6F騶邵嵋에ee4B61엣
6ege3eeee6e·
4仕p'26ape027ao·
白OeZE90D긴eenoe。
2aO726aoe0020·
衿觸邵邵6C盼0eoe7474엣
rjr규a긷4aoeapo뱀IFe。
'b6772白peaO660·
BDn밍뱀‘7e白긴e4EE-
44267000662·
沈邵6E64邵卵에卵2E秘갯
COmm
옥start .〃山抄
D츤dmaltoHe...
Naalnl。O'·
汐Wirl32Progra...
彊no안a麴소
垣）巨）〔DCE）巨）巨）
*■NN스5657 .1'14·
…＿-
----------------
一－그＿_
·O
.0
탭隧

"The CreateFile function creates or opens an object and returns a handle to access the object"

"CreateFile function은 obejct와 object를 접근한 handle을 return을 만들거나 연다."

The name of the file the program is looking for!!!

Let's see that in the code

Program file name을 봐라.

코드에서 찾아봐라.

Don't forget: in assembler, you will find the parameters of a function pushed to the stack in reversed order (when POP'ed they come in regular order)

잊지 마라: assembler에서 너는 stack에 반대 순서로 넣어진 function의 paramter들을 찾을 수 있다. (규칙적인 순서로 꺼내질 때, POP이라 불린다.)

시스템 생성 대체 텍스트: 매h핏풀레이어'
ETCAreVersingtUtorialfornewbiesbylena151PartOZswf
- .-ㅁ騙
Paused
녈삥씌
-1帑「른
쐬꾀뾔」쐬」뾔뾔희희꾀工희꾀의끄끄희희스」희협劉쬐
l:IO4O10ao 츨
6R00
ES64020000
R3乙＇21꽈000
F-_(3H0
U나죡UIUU匕
00401807
00401eeC
004e181'
004e1020
0040102R
00401034
00401039
00401e3E
0040ie4e
0e401841
00401046
0040104B
004elese
00401e52
0e4e1857
0040105C
004e105E
00401863
004ele65
00401867
004el069
더0·추n1L-l'E
<J『．
D냐ORD
뎨X,D
D냐DRD
대勵휴D
、win32Prog"mmer'sReference
口回回
C7OS97214000
C70드gBa1400e
C70EgF21400O
C7O드R321400e
Bestand Bewerken
nCa..Ret
77Z1’튠OOa
R7a1400e
e4
3F030000
RB21’홑000
O07Fe000
00
C8020000
RD
,D닒
RD
驢kE사솥EL3a.6erhOCI
PTRDS:〔402177],
PTRDS:[482197〕,
PTRDS:【40219BJ.
PTRDS:[40219F],
PTR05:【4021R3],
ORDPTRDS:t4821
PTRDS:t4021R71,
―얘으出沁p嶼ve
텨adWUZerO砂eSH래
-ndex ㅣ
CreateFIle回爾而
DED
Enk
<J"P
DUORD
7Fee
0
《J뿌
山勵RO
U
LLUUuVUVusHSH
대恥恥恥恥舶恥恥PUPU
的髓0e
.'USER32.LOadIOOn
PTRDS볍【4021RB】
재eCreateFile釉nctioncreatesoroPensthef미Io＼納ngo이ectsandretumsahan비e
RFZ14aOO
ee
SFZ14000
03
ee
O3
OO00aeCO
79a04000
eB02BBOO
SFF
lD
00
00204000
17ae400e
00
.各USERSZ.LOadCurt
PTRDS:〔4021RF】,
thatcan騁usedtoaccesstheo비ect
r솥냐erse".004B216F
D白p뱀‘
''e뱀‘
00뱀‘
Uel
Pl'뱁‘
noapsnU
VSHsHsHSHSHSHsHLLPzsHSHSHSHLLLLPsHSHSHSHSHLLST、
恥PUPUPUPUPUPUPU매얘개PUPUPUPU釀8刪PUPUPUPUPUmTE·’I
。1114''꼬' :
'’너4「IL-5"8
0040i07D
OO4e요07F
00401084
00401089
004e1e8B
00401e90
00401095
OO40109R
0040ie9C
004018RI
004eleR3
00401eR8
004010R9
004010nE
r놈r令』＇춈눼r놈「놀r노
〔0DI:10000
什Ver$ell.00402079
(JP.各KEF꺼EL32.Create
E댜＜-一1
하緞요Trev●rttH.se40ieg
0
rever룔e".OO40200e
r?ver$e".00402017
0
히es
pipes
maiISlotS
COmmunicatlonSreSOurCeS
d始kdevices渺뇨‘--｀ㄸ
UND(O0e0000
,GE,LEI
00000000
F
13Rd,538R83R8RRR09883
gC텨6r큐ECH66ER6666666E8
D7e20eee
24e20000
83eioeeo
ee
73214000
46
IR214000
ZFe20eee
Ce
<J"P.各USER32."e'’술9·B
<JP.하＜E사IEL32.EㅐitPr
COnSOIeS
directories(o
50,what15firsther이Wi∥
belastinaSSembler
r랖p.rs.".0040i21D
.1
r뺨Verse".0e402173
4亡．
H^NDLECreateFiIe(
LPCTSTR눋曰eNaα｝e,
rever꿸eH.004021iR
Ef枕
<JP.'KEI괍IEL32.Re솥dFL
E챠＜.EnX
.·‘··'h· r놀r놈』r닙.rb「딩
〃peintertonameofthef
〃access(read-write)mo
〃Sharemode
〃pointertosecurityattril
〃how!0Create
〃悧eat!"butes
〃handleto5leWithsttrib
花胡6868胡ESESEg胡68胡68騶Es邵f
…노Z … …
004el283=<JP.各KE勵［L32.Creat손F료len)
敞엽re맬꿸
004020e0
004021:110
OO4e202O
0040203e
00402040
004020Se
004e2060
004e207e
00402080
004e209e
004020n0
DWORDm''[)eslredAccess,
DWORDdWSha＇탕緘浴e,
LPSECURITY.-ATTRIBUTESIPSecu吻AttributeS,
DWORDd',,creatronDlstr心utton.
DWOR【）’斛RagsAndAttnbuteS,
HANDLEhTemPlafe戶Ie
);
藏欌ㅓERIc－스
nRE._RERD:F
ParameterS
《Il聃I
더
LV:HIDDEN:면
L
Csi6D4F
KeyfLI'
.dat.K',filei'
notva【Ld.SOrr
,··.......······
0012FFCC
00!ZFFDe
Oe12FFD4
nnl？「「ng
FFFFF「FF
7FFDE000
SO54R6EO
더더'츨「「n요
柏6F觸6E卵卵腑貂乙72엣
7269恥沈閔에盼6C6972맺
rD4FE댜oee98Fdp·
67ZO6eeapZO260·
7661沈乙的00卵66邵騶옛
rpr구4loeOgCOO·
6776apee762ao·
ZCr구seeerOgEe·
r규67600nUZO620·
泌616F臼邵卵에4B6664엣
尸구6eZa구eee99e·
67277ee276ao。
Cr구4r구EeOUerpCdO·
64676eoe660-
69沈6F騶邵嵋에ee4B61엣
6ege3eeee6e·
4仕p'26ape027ao·
白OeZE90D긴eenoe。
2aO726aoe0020·
衿觸邵邵6C盼0eoe7474엣
rjr규a긷4aoeapo뱀IFe。
'b6772白peaO660·
BDn밍뱀‘7e白긴e4EE-
44267000662·
沈邵6E64邵卵에卵2E秘갯
COmm
옥start .〃山抄
蔯珊님toHe". r.碼굵麻「仄7
汐Wh32Pr어ra...
彊no안a麴소
垣）巨）〔DCE）巨）巨）
■ N H 스 5813 .1'14.
----------------
一－그＿_
탭隧

So, what is first here, will be last in assembler

그래서, 처음이 무엇인지 봐라, assembler에 마지막에 위치할 것이다.

시스템 생성 대체 텍스트: OO40iOSC
0040105E
釀）401063
00401065
00401067
e04e1069
0040106E
6n00
686F214000
6Re3
6n00
6ne3
88eO0aaOCO
6879a04OeO
ESOBeZ여nB
83F8FF
.v7드!D
6Roe
68OOaa스000
'817a卜흥이沁
6nee
ESD7OZ000O
ES24020000
.vEg83010000
> 6R00
6873a14eeo
6R46
68후PZ堆000
0
r솥Ver튤．".004e216F
'e「ile
r HUㄴL
REnDONLY:HIDOEH:S부STE『1:RF
OPE'LE)<ISTIH6
슬
COO0000;:l
reVer'.H.004e2e79
<J뿌．各KER"EL32.CreateFi【ep>
EF盼＜,-i
하抑RTrever룔eh.0040109n
0
r'V.r''".OO4eZeeo
rtUer'.H.OO4OZei7
t그=NLjLL
d·=FILLSHpRLRERDIFILL아뛰RE
=6타fRI익R타괘DIGENERlc-URITE
e=미（·，{i!..d솥탭－
aU ·10 hF
幌燃
■탭■■■■■탭■탭■■■■■■
＇따．묘■＇따·■■■■■■■·
ol:14e11:17：〕
0040!07P
0040107D
0040107F
0더4Olee4
00401089
e040108B
00401090
00401095
0040109p
0040109C
004eioni
004010n3
슨=nB，伊
. = -7즈뺨，
:r뙤－RPPLn0LnL
F(【.R.V'r엥솥h."
0
<J뿌．흽USER32."
《J뿌．各KERr쨍EL32
직O닐n른.
鱗놉luat｀애perIod。ut。「d.t..
HULL
J"Pr슝verserl.0040
PUSH너
PUSHr.V.r별．".004
Andit15notabigmysterywhatfi-e
theprogram15―ㅇ이겸ngfor,right???
-PUSHPUsHPUSHPUsHPuSHPuSHPUSH一醱棚PUSHPUSHPUSHPUsH臘匯
PUSH48
PUSHrtUers.".OO4e

And it is not a big mystery what file the program is looking for, right???

OK. The ReverseMe wants a file called "Keyfile.dat", well then … let's give the ReverseMe what it wants and make one!

이것은 큰 문제가 아니다. File program이 무엇을 찾고 있는지 보이냐???

Ok. ReverseMe는 "Keyfile.dat"라고 불리는 파일을 원한다. ReverseMe에게 원하는 것을 만들어주자.

Open windows notepad

And save Keyfile.dat….

Yes, leave notepad open, we may need it again :)

Notepad를 열고 Keyfile.dat를 저장해라.

예, notepad는 열린 채로 놔둬, 우리는 다시 필요할 때가 올 거야.

So, let's execute the API.

Press F8

API를 실행하자.

F8 눌러.

시스템 생성 대체 텍스트: HrLJe
SRt
＝凶旺’'K
SHSHSHsHSHSHSH
-UUUUUUU
--PPPPPPP
0040105C
0040105E
0040!e63
0040!065
0040!067
004e1069
0040106E
〕O40X'7프
6R00
686F214000
6FI03
6n00
6Re3
68OO0aoe脯I
6879204000
E：크0B02O0O0
S：〕FeFF
0
r솥V솥r룔．".004021'F
'Te춰DlateFi
pttrLb니te욜
竺
「 HULL
RERDONLV:HID。EH:SVSTErl:RF
咸업．=OPE「L타汀STIHG
psecur't닐=臘LL
洲areh。d.=FILE一
니CCe룔룔=
「헌【eHan슨
Creat슨FL
e타fR
옭
RR
RD
"Key
LeEOp
:GEHER
at'-
:FILE一S넘衲E
IC--URITE
CRLL
.-lt-141-l놔갸一F
0040:07D
0040:07F
O04e콘084
OO4e맨059
!O
00
0020」nnp
17204이泌
CllP
IJIㅓZ
.P嵋H
IPUSH
PUSH
야，
TLt
le=
【e'
t =
며（:「뙤－RPPLn0LpL
.,FileRㅎver''"e''
〕｀욕0스eEE
.-14nip：〔
ee
D702
2402
83el
80
73al
46
표欌1
3
COO00eO;:1
r얍Ver륭．H.eO4eZe79
<J뿌．各KERr[L32.CteateFLleR>
Eq<, 1
하蹈尸Trever드eh.ae40!09R
e
r'Ver맬e".00482000
re"ert'".OO402ei7
冬1
FilefOUnd브!
(When『lotfound,ㅌAX
Wㅇ미dbeFFFFFFFFh)
PreSSFStoCC)ntinUe
ㅣ
ho닐ner
"Eva1uatL。nperaod。ut。「d.t..
=H"LL
nes$ageDoㅐR
00401邸--
간연얀e"'- …
<
ERX=00000040
ZFO
C0
ㅣ
EㅐLtPrOCes룔
.r_.길「r「컨=NLjLL
닛te괌· =rev'r'eH.00402173
βe뜨Tl·-」) 46(7e.)
u「fer reverse".00482llR
.「〔뗀르-eeee0O4e(니뢸ndo닐）
Re칡’FL【e
rOR88nH889nH8n"8e86·
766'6EEE6'66rDES떻
J>··
CIH ｀ㅗ
→ 난 1
·1-ㅡ·1겨1
-··'·‘…令· 7송r송』r슴·궁눙r팀』

File found!!!

(When not found, EAX would be FFFFFFFFh)

Press F8 to continue

...and we will jump passed the badboy!

Let's continue.

파일을 찾았다!!!

(파일을 찾지 못했을 때, EAX는 FFFF FFFFh이 될 것이다.)

F8을 눌러 그리고 우리는 badboy를 통과할 것이다.

해보자.

Scroll up to see what the future will bring us :)

And continue stepping over the code

이 사실이 우리에게 무엇을 가져올지 보기 위해 scoll 올려.

그리고 code를 실행해보자.

All right. Now the file will be read by ReadFile. Let's see the API

맞아. 이제 file은 ReadFile에 의해 읽힐 것이다. API를 보자.

See the explanation?

That clarifies a lot, doesn't it?

설명서 보이냐?

꽤 명확하다. 그렇지 않냐?

In short, ReadFile tries to read our Keyfile.dat for a certain number of bytes which it puts in a buffer at a certain address.

요약하면, ReadFile은 Keyfile.dat의 명확한 개수의 bytes를 읽기 위해 도전한다. 그것은 정확한 address의 buffer에 들어간다.

Take a look at the parameters

To understand the following in assembler.

Paramter 들을 봐라.

Assembler를 이해하고 따라와라.

ReadFile reads 46h(==70d) bytes

At 402173

ReadFile은 46h(==70d) bytes를 읽는다.

*번역 주)h는 16진수(hexa decimal), d는 10진수(decimal)라는 것을 이해할 것이다.*

시스템 생성 대체 텍스트: ad,飇dulereverseㅆ
F'lJ.3Hl:l
CRLL<J"P.驢USER32.he''agesoxn>
CRLL(J뿌．各KEF枕IEL32.EㅐitPrOOe룔룔＞
J"Prever$eㅒ．OO4012iD
-PUSHO
PUSHr솥차er’슨h.004e2173
，』：〕硼날t쑈H
)20000
〕20000
〕10000
轍4000
'he$＄슬geBㅇㅐR
EㅐLtPrOOes흡
at402173
7一盧
IFPU)
00000040
7C910드5。n쐽니11.7C91056D
pouerIapoed=臘）LL
p히t슨＄R손ad:rever$eh.00402173
B,tesToRead뜨46(7e.)－■．.=
毓汗f'r,r·난．r'e숴．0e402ll슈ㄴ
hF릍';=0000004e(님indo닐） 겨ㄴ
Reyr【【e '
/
ㅜ叭Inderstandthe
r.. , , _ .. 놀＞
r이IOWmglnaSSem이eF. /
ReadFilereads46h、
方＝70d)b7t-es.7
EIP00401야눙9reVerseh.O04OIORg
t수힁수一수황t
짭21400e10
LRa140OO
JnPSHL汁《. r슝V.r$.n.UU4U
XOREB눙.EB:=<
묘ORD31,ES】
ChF·αJOR[;FTF:D하【40217
JLSHORTfe닌亡r슬e".e0401
hCll」RL,E'.'TEPTRD뱁크:[EB:<
ChPRL.‘겨
JE하沁RTrev.r흡내．OO4el
C"PnL,4
J"Z하刃RTr.V.r룔eh.ee40
!"CESI
IHCE탸＜
01FFFFFFFF)
Mmmm.ReadfileW∥―
succeecl.ButㅇurKeyfi-eis
empty.50,ReadFilewi∥
readallzeros.Let'ssee·
tㅇPlacethemina
bUfferat402llA
e(FFFFFFFF)
0(FFFFFFFF)
0(「FFFFFFF)
7FFDDOOO(FFF
요OFㄴ히＿ll:CESS(00〔
iHO,B,HE,BE,NS.PE,
r刃안IBB80010드0104

To place them in a buffer at 40211A

Mmmm. Readfile will succeed. But our Keyfile is empty. So, ReadFile will read all zeros. Let's see.

Buffer의 40211A에 저장한다.

Mmmmm. Readfile은 성공할 것이다. 그러나 우리의 Keyfile은 비어있다. 그래서 ReadFile은 zero들을 읽을 것이다. 봐라.

시스템 생성 대체 텍스트: [：딘
Op
00000000
INFO:
ㅜheirlfo(ifany!)afterreturning
frㅇmanAPIISALWAYSineaX
《
ERX:Oe000001
Seethat
ReadFile
SUCCeeded.
en따eapao仕킹eeapedOC
e白O000000eapee

See that ReadFile succeeded.

ReadFile이 성공한 것을 봐라.

INFO:

The info (if any!) after returning from an API is ALWAYS in EAX

API에서 Return 후에 정보는 항상 EAX 안에 들어있다.

*번역 주)중요한 정보. 항상 return값은 EAX 안에 저장 됩니다. TEST EAX, EAX 같은 것들이 이해될 겁니다.*

And so, the jump is taken.

Continue.

그래서, jump를 한다. 계속 하자.

Do you still remember that

XOR REG, REG

Makes the register zero.

(XOR'ing the same register)

This is done to prepare EBX and ESI for the next operations.

아직 기억하냐?

XOR REG, REG

Register를 zero로 만든다.

(같은 register를 XOR 한다.)

다음 명령어들을 위해 EBX와 ESI를 준비했다.

*번역 주) 0으로 초기화 시켜서 다음 명령어가 들어와도 문제 없게 해놨다는 뜻*

시스템 생성 대체 텍스트: e04eion3
004010R8
O心4010습승
매401OR亡
OO401pBO
釀）401OB2
Oe40lOB’흗
004010E:8
너∥4U긺‘正…:
0t:1401OEF
00401〔llEI
0040i0C7
毓）40t0C.)
004010CB
OO4OIOCD
e04O10CF
004010De
eO40lODI
004010D3
004010D6
004010D8
oe4o1ODD
004e10Dㅌ
004010ㅌ2
004010E3
OO401OEJ
004010「
더더4더,
rever!.H.OO4e21IR
Enk
(JIP.'KERHEL32.ReadFL【e>
Eg뗀.E庶．:
SHORTrever$e".O040leB4
하緞RTrever＄애．e04el0F7
EE::.::,〔eX
E：·I.LSI
Acomparewithl().
Now,what1510구
囑UsH服聊限
PPCTJ
IRZ1400O
aF020000
I그O
l:l손
43
DE:
F컨．
Andwhatis【402173]?
'.snO01:Iㄷ닝ㄷ－.BS긍스·
-6LpE乙l-llE各구=：·
7 8貂02321
ㅓ吻IJ
36
83IRa14OaO
00
e8
47
0l
＿메卯汀卞TRO읽[40기73긔l니
SHt)RTr.V.t흥슬n.004e10F7
---> 10h==16d
(10hexadecimal1516decimal)
EE
「Ees
lF
28018000
J"P
XOR
겨：즈I
C驢IP
Jㄴ
"OV
C"P
JE
CllP
Jl쨍Z
IHC
IHC
J"P
C"P
JL
JllP
[I[;
「ㅣ더
RL.머，TEPTRDS:〔EBX+402llFn
RL-‘〕
하拓RTreVㅎr룔e".O04el0Q3
RL.47
하細요Tr.ver''h.00401000
ESI
EBY
하緞盼TreVer룔슨h.e040iOCI
ESI,8
하細RTreVer슝eh.OO4el0F7
rㅎv。r'e".00481285
수·t
이、*'1'·｀·LL.
뒤
:;。'
니i--1 一 _
CnHC4Cr갑'3B3CgO
78373탕r44E87Ee
갼＞·갼·갼->＾·＞갼갼
IU더넓너qUn빕d긴떽
(-1lt-1-UonUO糾
∼--n尸샅bb.．놀BBBI
--LI니J니DDD一
UU4닙--_
004e10E8
e04o10E9
004010ER
00000000
0e
0e
0e
00
0e
00
0e
00
00
Wel-,Ialreadyt이dthatinaprogram,ALL
valuesareinHE)((hexadeCimalsystem).

A compare with 10.

Now, what is 10?

And what is [402173]?

10과 비교한다.

*번역 주) 여기에서 10은 10h다. 10d와 헷갈리지 말자. 10h == 16d)*

왜 10인가?

[402173]은 무엇인가?

Well, I already told that in a program, ALL values are in HEX (hexadecimal system).

나는 이미 프로그램에서 모든 값은 hex라고 말했다.

---> 10h == 16d

(10 hexadecimal is 16 decimal)

시스템 생성 대체 텍스트: （노Zr＝』」’r，밉－』-----－』
더Cp【】,mainth刪미飇dulereyerseㅆ
004Ote89
e040108B
OO40iegO
0040109드
0040109n
004010'?l-
이’4On".
a더40'一
·H'J _
볍R
E8
0O
D7020이ao
PU':;H니
돌r쑈「4
32.He$'ageBoxp>
Andwhatexactlyis
.Eㅐ脾PrㅇOe룔r>
1D
Lhe$sageBoHR
EㅐLtPr。ce듀S
Infact,thisisthenumber
ofbytesthatwereread
넌‘rL」「「，c
comparedWith16d가？
!73
11R
PUSFk又nk
ㅣ
- 一「-「'
… :T-F
-+드r 一
.ㅏI。슨 .
ReadFi【e
걀희
■」=NULL
=r슬V슨r룔eh.OO4
1=4'170.)
173
reVerse".0e4021IR
0000e0ei
ECX
ED옥
EP닛
EEF
E드ㄷ‘
EEI
[〔I
Ll『
00''自I·욥더ㅡ츤saU
cPCHr'l=-TDO
2F0200Oe
］天I
l:l손
43
DE:
F8
3nr祗그4더걍닌1;:l
36
8319214OaO
e0
08
47
ei
CnLL
TEST
IJllZ
JnP
SHO배
하刃RT
.各KERHEL32.ReadF쵱【e>
ER·；
'teV순r룔e".0040i0B4
순vers슨h.0040!0「7
논＿IR
ZOR
C"P
JL
"OV
C"P
JE
C"P
JHZ
!HC
1"C
J「IP
ChP
JL
JnP
EE:::,「「'
E도1,B51\
튄驪RpPTR-DS:
〔4UI1rg〕
《콘《콘쵱빻（륫'（〃니
0040i0C7
SHDRTr．난er룐eh.00래
RL,BVTEPTRDS:〔E
RL,0
eF7
、略021in】
50,theReverseMecompareS
somethingWith16d
7C80189e.
7CgeEH94nl
000O8eoe
eolZFFC4
O〕츠鉀[‘「O
000000Oe
一 '_07 :
O04OIOB6':
ㄴ.0··.:
;:_ 0「신〔
그．꽈 Uㅑ :
DE O긔l프 E
「프 ―더 ㅏ
l' l
하【RT
nL,4'
하細RT
ESI
E타＜
하抑요T
ESI,:E
하츨ORT
r.v'r흡상n.O04el
EFL
La맬tErrERRI
000嶼尸빼‘(NO
r.V.r'.rl.oe4eleDO
EE
FEes
lF
28010000
reV
r샅ver톨e".
。1
STOenotyeeUHORh
STIehpt,e.0
STZehpt,0.0
ST3ehptye.e
ST4emptye.0
Sㅜ드er.lpty0.0
Sㅜ6땁l.lp',1.0叭＇1:11
·프T7ehptu1.0000,
nOOI:Iㄷ-rpB…、心OCnHC4CrD63B3Cgap
r규E〃c’·E ·879373744E87Ee
什＞·J·갼·＞＾·＞갼갼
D크T
ㅏI_m
4020
027F
ㅣ＿〔「…〕
「rr-
一。、。eeeeeoee·0000··
0040!0C9
00401eCB
0040t0CD
004el0CF
004010De
004010DI
0040!0D3
004010D6
00401OP·
L:IO401')
q더4p'' .
썹
UU4넌1ULb
앴
4010EC
<'
-'l'-
l므닙U넌
l맸맸
【402371]meansthevaluefrㅇmwhatisat
theaddress402371.HenC이inourCaS티itis
comparedWiththevaluefrㅇmthedwordat
address402371.Trytㅇunderstandbutd이ft
wㅇrrytㅇ0muchyet.Allisrepeatedagain
later.OneCanalSOf이IOWtheseValUeSinthe
dumpWindowwhichwi∥darifythingsalot.
Wewi∥seethesethingsinmoredetai-later.
DS:[00492173」＝00000000

So, the ReverseMe compares something with 16d

And what exactly is compared with 16d ???

그래서, ReverseMe는 16d와 비교한다.

그리고 정확히 16d와 비교하는 것인가?

[402371] means the value from what is at the address 402371. Hence, in our case, it is compared with the value from the dword at address 402371.

[402371]은 402371의 address의 값이다. 그리하여, 이번 case에서는 402371 address의 dword값과 비교됐다.

Try to understand but don't worry too much yet. All is repeated again later.

이해하기 위해 도전해라. 그러나 아직 걱정하지 마라. 나중에 다시 반복됐다.

One can also follow these values in the dump windows which will clarify things a lot.

그것들의 값을 dump windows에서 따라가라. 그것은 꽤 명확하다.  
We will see these things in more detail later.

우리는 그것들을 꽤 상세하게 나중에 볼 것이다.

In fact, this is the number of bytes that were read

사실, 이 숫자만큼의 bytes가 읽혔다.

*번역 주) Lena의 오류다. 실수할 때도 있죠.ㅎ 402371이 아니고 402173 이다.*

Let's continue

계속하자.

Aha, and the code jumps to the badboy if the number of bytes read by ReadFile is lower then 16d

아하, code는 ReadFile에 의해서 16d보다 적게 읽었을 때 badboy로 jump한다.

Mmmm, in short : the number of bytes that ReadFile reads from Keyfile.dat must be at least 16d (==10h). Ok, let's remember that for later. Now, let's make the program think it has read at least 16d bytes :))

요약하자면: bytes의 숫자만큼 ReadFile 읽기는 Keyfile.dat에서 최소한 16이어야 한다. Ok, 나중을 위해서 기억해라. 이제, program이 최소한 16d bytes를 읽는다고 생각하자.

시스템 생성 대체 텍스트: 'ld맣·lees
CPCHZSTOO
.:1041:110RE
1:104010E0
Co l
0굽 -
스3 -
DB I
F昏es _-
욘날뻑뾰呱츨
831露21生姃따 J
SHORT
5ㅐ【】RT
En·：
teV.r룔e".e04eleB4
「IPE
톄· r
,EE::
.V뺨r맬싣숴．004e!0F7
RememberthiSp
I,E기
汗굿［;FTF;:Dg:【4e요l
JLS냈承Tr.V.r룐e".ee4e
죠i흐一一
「7체■■「
T
（。ZPRRF.
TJJ一→·．＾一p-
r흽밉 군탬굼뱁펩뱁J뱁J뱁긷떵순탬굿탬
Lel234567
FTTTTTTTT
ESSSSSSSS
nL.던7TㅌPTRDS뿜[EBY+40211自】
RL,0
SㅐDRTreVer룔e".O04el0D3
RL,4-
하蹈RTr.V.r!.h.oe401晄增
l「
28010000
ESI
E탸＜
하緞RTreVerseh.e840iOCI
ESI,8
하【요TreVer슝야흥．e048leF7
r'Verse".O04el28S
'1너
rI00nnl:ll:ln
PZCCPP P
대개대川棚얘JL刪EIE:DDDBDBDBDBDBDBDBDBEIBDBDBI
BeB847BIEEEee
F
邵花臼33貂的松騙8C749c7s4648Ee8a沈Egee
VV V· V V '^ · \'》
· ·、·노Z · ·'·노Z ……노r··노r ·r
FST'
FC.〔
曲腑恥애脯腑卵卵卵盼閔一
一eeseeeeeBeeeeeeeeeeeeB、
히一4、匕01-.Fi79BDFel3ZUSDE234rp6789CLBC'
npE:E:nuEIㄴCCCCCDDDDDDDEEEEEEEEEEEL
硼刪씬緇刪緇010緇刪8le刪刪刪elee10刪拙헤
刪溫꽤刪溫OO4刪刪棚溫溫刪溫004刪棚굶
J녜＇pLstak손。
oa40leF7:rev빵r$'h.0e4010F7

Remember this?

이것을 기억해?

Right. We don't jump to the badboy anymore.

맞아, 우리는 더 이상 badboy로 jump 하지 않는다.

시스템 생성 대체 텍스트: 여여여여여
나SSDS円GS
'e·닌ee
rRZST
00401
0040노
00401
00401
‘걔E
0ㅌ
0ㅌ
0E
33DB
33F6
833D732140
7C天·
．벼흽．활ina14e
3Ceo
74BS
3C47
75el
45
43
EBEE
93「E08
7CI「
Eg2801000B
ee
OoeOOape
ee
6e
ee
.e
B0
0n
:=.:clR
:^:OR
E텨＜,ㄸ＊:
ESI,ESI
궤ll;F[PTR「I승=【4L-，드1.'::〕.10
놉＂U"■ re난er룔en·UU쟈UIU「r
―뗘뺘―·
0041:11더죠7
00401더〕）
0040!OCB
0040t0CD
oa4e10CF
004el0De
004010DI
004010D3
0040!ODE
004010D8
004eleDO
OO4eleDE
0040i0E2
004010E3
00401OE4
004010E드
0040!0E6
0040iOF7
RL,BVTEPTRD읽[EE::.:+4OZllFn
RL,‘〕
하【RTrev.r룔얘．OO4e10D3
RL,47
하拓RTr.V.rS'h.oe4010De
ESI
E탸＜
하緞요TreVerseh.0040iOCI
ESI,프
하拓RTrever룔eh.e04010F7
r·v*r'e".OO4012eS
Nowwhat15next?Wel-,wewi∥
havetodealWithaloophere'First,
lookatthenextlinesofcodeand
trytounderstandwhatwi∥happe『1.
00e00000
00
00
eO
0O
O0
0O
ST7
FST
FC.
＇펩ty
enoty
4020
e27F
P UP P뱃LCCPP P
i:'1JLm애JE대硼m새棚애JL刪EIE:DDDBDBDBDBDBDB

Now what is next? Well, we will have to deal with a loop here. First, look at the next lines of code and try to understand what will happen.

이제 다음은 무엇이냐? 우리는 여기에 있는 loop와 거래할 수 있다. 먼저, 다음 lines의 code를 봐라. 그리고 무슨 일이 벌어질지 이해하도록 노력해라.

시스템 생성 대체 텍스트: ㄴ--
「伶논
더CP【j-颱inthread,飇duler아erseㅆ
匹同因
!ee
PPt'
嚼隅隱膣
^
ee
D7숑ZOaoe
24승毓pOO
8301여no
PIJSH.1
CRLL<JllP.'USER32.h.$'ageB애R》
CpLL(J뿌．흼KERHEL32.EHLtPrㅇ。e륭5>
Jr1Pr슬verse".0040121D
h…닐ner =
TheapPliCatiㅇn15goingtomㅇve
thevalueat【印）(+402llA]inAL
'"e''ag.B。ㅐR
EH'tPrOCe｀룝
姃贓er【ap。e（」=
pB닛te$Rㅎad=
i톨tersIFPU)
n"8e9
'EEE
-…六
＂··
r솥Verse".Oa482173
4617..】
Xe000鱗땁l
X7C801898k솥rn솥【32.7C8el898
X7CgeE요딧4n'd【L.KLFastsy't·nCaltRet
盼J800e00e
V뺨r｀성숴．
000．君〔
a4821IR
I麟1’。)
ZFFC4
슛FFFO
'BerDCHC뱁‘38
·89999nHRnH
탤仕pnp祛UaodU0dO
1홑펩．솥·■홑··솝뱁■욥·■춈J．욥뱀흩
00O0aUOeO
44444444
odU000000
－너仕빕apoaU仕paUO
、｀'．■．lesl//
-
Dyte'TORead
鱗』ff.r:r'
hFi【．=0..
Re台dFi【e
75eZ
E843
33DB
33F6
833D7술2140001
7C36
앴·’갰9al.10클O'
'쑈:.
JHZ
JP
지〕R
시〕R
ij'1。
JL
NOV
CllP
JE
C"P
J"Z
IHC
IHC
JhP
이IP
JL
JhP
SHO
rev횬r$ell.ee4010B4
r.v.r｀빼．004Ol0F7
Inthebeginningofthisroutin타
印×wasXO힘ed,henCeEB×
15ZerOatthismoment
·>.갼＞··
r특l너－O쐽
nHEB
뱁‘－이·흩一·．수．■욉·．ㅗ
np石plapl=-
4ㅡ4ee4계닉
a빕石p一apl-li
白p苒UlaVee너ㅡ
0040
緞응權뿡
004010E；캬
004010EF
뜨팟죠
깬11_;FD
0040
0040
0040
0040
0040
0040
0040
0040
e04o
0040
004e
0040
0040
0040
0040
0040
0040
匯140
0040
0040
0040
0040
하（RTr.
RL,E:'./TE
RL,0
팟DS:〔4q노173〕,'0
.r룔낡1.e04010F7
FTRDS=〔EE:：·：+4021
EE
FEoe
1F
2801aO00
하拓RTrev。r흡애．O040l0Q3
nL,47
하【RTr.V'r'.h.0e40100e
ESI
E먀＜
하K땄Ttever$eh.00401毓〕1
ESI,；「
하艇戚TreVer룔삵1.8e4019F7
rㅎv'r'en.0040120드
Whi-st【402llA]beCㅇmesthe
firstbytereadbyReadFi-eand
isputinthebufferat4O2llA
-->[.......．」mea『15"valueof"
De
Oe
EFL
La'tErrERR아L있rCESS(eoaeoeee)
00000207(NO,B,"E,BE.NS.PE,6E,6)
STOe"pty케HOR"BBSOele릉0104000eeeoe
STle써땁，e.9
STZef뼁）t,e.e
ST3e"pt,e.e
ST4e"ptVe.8
STS.＇탤嘴，0.8
ST6."pt,1.e00eoaeeeeee00eeaoe
ST7e"pt,1.80080aO0Oe880080000
「요1ㅣ F푸PUO:D
FST4eZe COnd1eee Err001e..0
FC닒e27F PreONERR,53 ＂솔＄k 료1111
盼e847el
날·α·:v
Yㅇucanfo∥ㅇwthisinthepanewindow
tㅇ（LBTW,AListherightmosttwo
digitsfrㅇmㅌAX.SeealsothatReadFi-e
hasreadthevalueo()forthefirstbyte
>·．·＞>.、． ·
40
D0
r놈r졉
녹순
ac74ac花4649EB田7CEgeo皿舶盼애ee朋B888開eeeeeeh
79BDFS'1씌구68DE234rp＇뇬l】솥!1HrLJC'
CCCCCDDDDDDDEEEEUJEEEEEEL
OOdOO0dpapO00쵸OoapnpnU0lUn·너-UIUdUaU什
괘
에一r
〕S:【（珊402!lq]=00겯，
ㅓL=01
Jun。「r。hOO4e1ODI

The application is going to move the value at [EBX + 40211A] in AL

Application은 [EBX + 40211A]의 값을 AL에 이동한다.

In the beginning of this routine, EBX was XOR'ed, hence EBX is zero at this moment

시작하는 routine은, ebs를 XOR 했다. 그리하여 현재 EBX는 zero다.

Whilst [40211A] becomes the first byte read by ReadFile and is put in the buffer at 40211A

40211A는 ReadFile에 의해서 첫 byte가 읽힌다. 그리고 그것을 40211A buffer에 넣는다.

*번역 주) 여기에서 왜 byte가 읽히냐면 담는 그릇이 AL이기 때문이다. AL은 byte, AX는 2byte, EAX는 4byte다.*

---> [……] means "value of"

[…..] 의 뜻은 "값을"

*번역 주) 즉, [40211A]의 주소에 들어있는 값*

시스템 생성 대체 텍스트: 뎌CPtj-mainthred미飇dulereyerseㅆ
口回睡
ister룝IFPU)
000000el
7C801898kerne【32.7C8.le98
7C9eE894n탭dL1.K1Fa'ts부＇tel祗a11Ret
Oaaaaoas
0
0
ZFFC4
으FFFO
險啄層
^
00401089
0040!e8B
00401090
00401095
0040!09n
0040109C
0040!eRI
004e10R3
0e4010R8
OO4010Rg
004010RE
.:104010E0
""4011〕E·［
.1114nl‘〕「4
J너4UI"b=
.-tU401I-IE：윙
.:104011:IEF
USH4:l
卜0내눙er:
ESD7OZ0008
ES24송毓변Oe
.vEg83010驪뺌
이뵙L<JHP.各USER32.".$'age毓悧n>
CRLL(J뿌．各KEF枕fL32.EHitPrㅇOe＄룔＞
J"Ptever'eㅒ．Oa40121D
'he'sageBOHR
EㅐLtPrOCes흡
TheapPlication15goingtㅇmㅇve
thevalueat【印X+402llA]inAL
匪ㅣuerLapoed=NLILL
p텨βt슨＄R손ad:r*verseH.0a4a2173
B,te'ToRead=4617B.)
4821iR
"dq却J
、｀·11--//
毓흩「f·r=r·닌뺨r｀·＂.
hF'【．=88e000.tf
ReadF[【e ,
r■■■■·■■L
75eZ
E843
33DB
S3F6
833D73al4000父
7C3E.
驕운
rever룝단＂.Oe40i0B4
r슨V.r‘상＂.0040!0F7
꽉D읽[4q크。’콰〕,11:1
.r휴eh.004e10F7
Inthebeginningofthisroutine,
印×wasXO힘ed,henCeEB×
15Zerㅇatthismㅇment
' ㄴ찧근＇LLLJ
Ln.r ' 'rP
L>.갼＞··
鱗첩O·＇【自손！4嶼犯
얏緞RTr.
RL,B모TE
RL,0
TRDS:〔EB:．仁＋402!
Oe
Oe
EFL
L.'tErrERR하LSUCCESS(000eoeea)
00000207(NO,B,ㅒE.BE,NS,PE,6E,6)
눼졀HOR"SBBOele50lO4OO0eoeoe
e.8
e.e
매8847el
a킬ee
■ · ·
n길ea룅뱁‘뱀‘
醱畛畛醱叩畛叩畛
eeeee··e
e뱀‘234567
TTTTTTTT
55555555
0040
0040
0040
0040
004e
0e40
0040
0040
0040
0040
0040
004e
004e
0040
0040
0040
0040
아〕40
筏140
J040
.1l〔140
EE
FEoe
lF
2801800B
하細RTrev.r$.h.O04el8D3
RL,47
하拓尸Tr.V.r''h.se40lODO
ㅌ51
EB요
하섯PTreVer룝e숴．0040iOCI
ESI,6
하【RTreVer룔eh.O04010F7
r솬v손r'e".00481206
! //
Whi-st【402llA]beComesthe
firstbytereadbyReadFi-eand
isputinthebufferat4O2llA
-->[.......．」means"valueof''
.080800aOO00
.sea000000e6
PRO히P UP PZCCPP P
刪功：^:::l네JL恥대JE대걔m川刪cllJL硼
LOU
·0
。0
∼ 뱀홑
Yㅇucanf이lowthisinthepanewindㅇw
toc).BㅜW,AL15therightmosttwo
digitsfromㅌAX.SeealsㅇthatReadFile
hasreadthevalue00forthefirstbyte
…7／히－一―---
솥｀■··■■．·．■L
낚戚■ V V ·n >V
·더■ ■ · ·…노Z ·Xr ·、· ·
FST4820 COnd
FC닒e27F Pr.O
N탸뛰，53
08.oeOOe
80..0000
프콩Pl_;f:IFD:
Erre01e.eoe
"ask 111111
D0
r톨r겸
녹걋
邵74部花’648盼盼7cEgee매馹8e脯ee盼卵ee卵開凹馹一
79BDFel·櫛68DE234rp'-r】‘무슥HFㄴCC
CCCCCDDDDDDDEEEErJEEEEF"EL
onUOOaUaonpnoa긴0e0凸pnpdp00一너너－너Up·aU什
．홑‘·홑·■욥·프쨍쵸．홑뱀쵸．홑··홑·．홑·■욥꿸쵸.．흡'''．톨·．춈·■‘·■‘'1^눼‘'.1ㅗ''·흩·
0e40
r놀r노』戚놀
r'涵1-
苗窈前花F市芷篩’
RL=el
小』hOFroh0040lODi
勳

You can follow this in the pane window too.

BTW, AL is the rightmost two digits from EAX.

See also that ReadFile has read the value 00 for the first byte.

너는 pane window에서 결과를 볼 수 있다.

By the way, AL은 EAX에서 2가지 값이 있다.

ReadFile은 first byte의 값인 00을 읽은 것을 봐.

시스템 생성 대체 텍스트: rr 쨍·TTT쌩■뺑‘7■무． 츤qC
'STDOE蝦驢눙If'1''r6s FF
P UP P,'
nLOZllE"H
一닐JNCJCJ
833D
7C36
楓」卿l어
곈JORDF'rRD하〔4Ca17
하緞RTrever$eㅂ．0e401
昌즙
；隨”盼’-
꽈髓EE '
RL,BVTEF'TR
nL,0
D드:【Ee맷＋4021In」
r.Ver룔eh.OO4el8D3
r.V.r'.h.8040i0D0
Inshort,theReverseMewi∥
reVer즐eh.004OIOCI
→ 무· 무·-
T4R R '
터H ·n닙T‘스UnLUt'
HDRLSH"EBSHES
> 83「E08
.v7CIF
.vEg28010008
Be
Oee000pa
Be
O0
山O
하【RTrever슝e".804010F7
r솥VerseH.00401285
I1I1
eo슬000eO
0e
Oe
여여
movethefirstbytethatReadFi-e
hasreadinㅇurKeyfile.datinAL
T
CPP P
IH刪C:r1JL硼EIE:DDDBDBnP

In short, the ReverseMe will move the first byte that ReadFile has read in our Keyfile.dat in AL

요약하면, ReverseMe는 Keyfile.dat에서 ReadFile이 읽었던 첫번째 byte를 AL로 이동할 것이다.

시스템 생성 대체 텍스트: →－*'－므一一一－→ … --.-
■■■■■■■■■■■■■■■■ -－·■■■■■■뻐…」그 -.．■■．-'
디CPU-耐inthrㄼd,飇dulereyerseㅆ
匡回
h。내놓el
UnUaPP''''P''뱁쵸
DBSBSDtl
EEEEEEECP
D7·毓日Oe
24송枕．ae
8301엣변0
0.
73a1400o
46
1會Zi48ae
ZFOZ000O
PUSH
댜뵙L
C自LL
J"P
PUSH
PUSㅐ
PUSH
PUSㅐ
PUSㅐ
CRU.
T드〕T
Jl눼Z
<JhP.各USER32.h.5'.g.B애n>
<J뿌．힁KEf꺼EL32.EH쵱tPrOC슝룔룝＞
rever'eh.OO4O12iD
0
r솥Versell.OO4e2173
4':
r빵닌er별’".OO4e2lln
탸楸
L".''ageB。xR
EH효tPrOCe‘룝
^IReg룁$ter'(FPU)
ER》
EC》
OO00000e
液〕801898
pouer【aOOed=
臘므L
r솥V.rseH·00482173
k솥rne【32.7Ceele딧8
ntdll.KLF.'ts,'te뻐Dat'R·t
psyte$Read
Byte'TㅇRead
Buf「er
hFLI후
= r·
=4817e.)
냐뺨r｀·＂.eB4821iR
7CgeEB딧4
OS000eee
0.12FFC4
ee1ZFFFe
=ROLL-〕〔u。·dc""
비텅嶼텀轍b
鸞［910738
：乙
slAlisthencomparedwithzerc).
0으央〔
ESSL、딕
ntd【'.갔［91e738
r.v.rteh.ee401戚7
32b1to(FFFFFFFF)
…什＞……
J"P
XOR
21)R
f:'nP
JL
r1CI：」
C".
JE
C"P
J"Z
IHC
!HC
JllP
CllP
JL
JP
OB
DD
OB
DB
DB
DB
DB
nq
EB성，Ee:::
ESI.ESI
LllJOF〔；F'rR
하（戚rrㅎV.
08B躍眺緞隅旺榴0B4臘
.Oe4e
73뇨10
1.F7
23 6
aoaO4B6D3
C DF3
ESEeEg해68卽68馹E8邵花印貂貂的7C
둔엠7麗
D드:〔EE::.l+4021Inl
하蹈RTreVer各슨＂.OO4ei0I)3.
RL,4-
하【RTr.v.rt'h.8e40i0D0
IfAL==0
(likeit15now)
thenwejumP...
BUtㅌ5115ZerㅇbeCaUSe
itWaS×O힘edbefore
Startingth15routine니!
STOenpty떽꺼ORllBB8ee1050iO4OO0eeeoe
STle"pt,e.e
STZe"pt,0.0
ST3·”pt,e.8
ST4e"pt,e.e
STS'＇맬嘴ye.e
ST6."pty1.ee08e000OeoeeeeeeOO
Sr,enpt닛1.80OB0000008806B000e
「요노. E픈F-_OF
FSㅜ4eZo COnd100e Err001eee
FC닒B27F PreOHERR,53 "a$k 료111
74部花拓·8盼SS7cEg馹
洽룐締ocFODS楓驪eDDeDEOE20E30E40E5eE6eE7eE80Eg櫛照
e0O0aU0eea킵es白peedoweo-U轟f-l石p쵸으nlllUㅡ니es-Ua킵a룅
444444444es4ㅡ4-444）뺙4ㅡ4－괘그ㅓㅡ44ㅓ뺙
00d씹OnUO00e百p蠱묘n탭IUaU白－0一on길we너es-U－너너
oapo仕U白U白pd긷白긴白U再므a모l:l白UO白－白브白깊凸Use·1est-1·－너、너·
OO4e
0040
0040
0040
0040
0040
0040
004e
0040
0040
0040
004e
0040
004e
OO4e
0040
0040
0040
0040
r놈了놈4r놈
EE
FEes
lF
28Olaa.B
ESI
E마＜
하솔ORT
ESI,a
.....tㅇacompare
하．RTre쁘鞭뺏：했鉀leF7、∼
r샅Versell·U盼묵Ui乙bb ∼＜乙
00
OOa鱗珊ee
0O
00
0a
Oe
00
n습
…〕
믹
ㅌ51With8......
[
wejumptothebadboy
ifㅌ51151ㅇwerthens
一Be··ee。geee··eeee·
U빅
·^ VV
■노Z ·XZ ·、· ·
<1.11-1-
탭■■·탭■■■■·
긔긔〕e
Ah미50,weneedㅌ51greater
orequalthen8toCㅇntinue
tㅇjumptothegoodboy!Il!

AL is then compared with zero.

AL은 zero와 비교된다.

If AL == 0 (like it is now) then we jump…

AL이 0과 같다면 우리는 jump 한다.

*번역 주)여기에서는 당연히 jump한다. 왜냐하면 ReadFile에서 읽은 게 0이니까. EAX의 값 0과 0을 비교하면 Z-Flag를 1로 set한다.*

….to a compare ESI with 8

ESI가 8과 비교된다.

But we jump to the badboy if ESI is lower then 8

그러나 만약에 ESI가 8보다 작다면 우리는 badboy로 jump 한다.

Aha, so, we need ESI greater or equal then 8 to continue to jump to the goodboy !!!!

아하, 그래서, ESI 가 8보다 크거나 같아야 goodboy로 jump할 수 있다.

But ESI is zero because it was XOR'ed before starting this routine !!!

Let's continue stepping to see it in the code

그러나 ESI는 zero다. 왜냐하면 이 routine을 시작할 때 XOR 연산을 했기 때문이다.

Code를 넘기면서 보자.

시스템 생성 대체 텍스트: ……*-－뺨··■■■■■．-
더CP【j,mainth隅d,飇dulereyerseㅆ
^IRe
En>
EC》
XU八PPtl.'
DBSBSD
EEEEEE
l논，나니ner=톄LjU·
'n．륭sag.bOHH
E써LtPrㅇCe듀룝
Douer【apped=NUL.L
psyt솥＄Read:r솥ver흡eH.004a2173
Byte드TOR.ad=4617e.)
Bu「「er=r.”광r별·숴．ee4aZllR
hF'【e=ROLL-〕〔UmdC'「
Re술dFLle
r■■■■··■■L
00401089
ee40108B
OO4OIOge
0040!095
ee40lOgR
0040legC
004010RI
004e10n3
0040i0R8
ee4Ole自9
OO401ORE
OO40lOBO
004010B2
PUSㅐ
CnLL
CnLL
J"P
EIP〔
SHSHSHSHSHLLr
UUUUURrl
PPPPPCT
凸p삽‘O긷le88e
CPpZSTDO
刪刪麗
OO4O10e증
Oe4띠醵「
eO4010CI
004010r'
Y丁j■미■탭
ESD7OZOaoe
"24송毓．ao
.vEg83010淺뺌
> 'Reo
687요214Onn
'R46
681自2148ae
5e
ESZFO29Oeo
85C0
7502
.vES43
> 33DB
33F6
833D稗Zi4Oeg
.?7C36
> 8R83lfn14C입걋
3C01:1
<JhP.&USER32.H'$'ageB애n>
<J뿌．各KEf갑EL32.EㅐitPrOOe룔S》
re"er드eH.oa4012iD
0
r솥Ver별e".00402173
4':
r빵per!.".eO4e2llR
티枕
(JP.흽KERtIEL32.ReadFL【e>
E써＜'티r·；
SHORTr.v·r룅슨＂.e040ieB4
하拓RTrev솬r｀상h.0040!0F7
ER<,EB::.:
단81,E.SI
Lll.l．〕F〔；F'TRD하t4L:'a17〕
하艇尸Tr.V.r$eㅂ．ee4Ole
즙1「
"CI：」nL.E'..Tㅌ:TF::D드::EE·：+40크1In」
11=11'n인－.. →
하艇只TreVer룔손h.OO40i0D3
PL,4-
하【RTr.v'rt.h.se4010DO
Let'S『IOtjumphere
beCaUSeWefirstneed
ESI>=8
ZPRRP
H"00터llL
JJ·’^·．＾→n닙」
lee시뻐―
뱁‘
EFL〔
.v7CIF
.vEg29010009
.e
ESJ
E마＜
하緞RTreverseh.0040i0CI
ESI,a
하蹈RTrever룔e".004e10F7
rev슨r$e".OO4elZeS
00
OOeOOees
STe
STI
STZ
ST3
ST4
STS
ST'
ST7
FST·
FC님〔
沁腑恥l:le開腑開腑明閔卵一
E飇配lfc肌飇11PL飇BDBBBBBBBBBBB'
JCJllJCJJDDDDDDDDDDDDDn
一BB8e·eB、ee·Beoe·ee·
DpDFe!各O견DSDE234rprO789CHBCC
weU츤LCDDCDDDDEEEEEEEEEEEL
더너dU凸OedOOe0OdodO凸UOOee습UdU凸UaU什―
L11이이이이이이이이el이이이이이이el이0l이이m쎄
444444444444444444444괘
맨I1I1、OO沈애에腑騁醜腑00騁騁00嵋OO嵋嵋腑애에一<
Juhp【죌taken
00401eD3=reVer'eH.004e1003

시스템 생성 대체 텍스트: ----'t一－7흐r'-'.---
.CP【j-耐柚thred비飇du흰ereyerseㅆ
00401089
0040!e8B
OO4Ole90
00401095
0040109n
0040!09C
0040!opl
O04el0R3
0e4010R8
004010自參
OO401ORE
O盼콩OIOB인
OO4OIOBZ
Oe40lOB4
USH
뉵0닐n숀r:
져洲P.各USER32."e'$.9.Boxp>
(J뿌．흼KERtfLSZ.EㅐitPr0Oe룔룝＞
r슨uer룝eN.OO4012lD
O
r솥Verse".Oa4eZ173
4〔
revert.".OO4eZIIR
탸耐
<J뿌．흴KEF꺼EL32.ReadF'躇＞
E써＝..ER，뻐；
SHORTreve,$ell.ee40leB4
하抑RTr슨v.r룔eh.oe4el0F7
姃릅＜,EBX
D81,esㅍ
Lhe$$ag·B。ㅐR
EㅐLtPrOCes룝
^IRe
-
=NLjLL
=r솥V손rse".0a482173
B,te$T。Read=45(7B.)
당u「「er:r’나광rs팝＂.0B402llR
hF나e=n0U길-〕(닝孔肌廊刃
Re魯dFi【e
EFIX
ECX
EDX
EBX
ESP
EBP
ESI
EDI
EIP
e뱀‘eaO08es
CPpZSTDO
qLLL飇USH囑USHUSH囑祗戮附
CCJPPPPPCTJ
PRRP :I·P
HOO터IIL·－lI"
J…－.＾一츤닝J"C
n띠ee o묘 a띵 n킵 n띠
O띵a띵e o묘 O뙤 n룅 4
一驟瞰麴旽蹶 一
743·솝요600 F 23
D28ez412盼e4DBF6沁
ESES댜胡68胡68騶E8韶花E8盼貂897c騙：f：겨9c花拓貂EB8s7cEgBS
…갼＞……큔갼＞··
O04OIOB승
00生쵸坤BEJ.?
0040i0Cll >
Lll.10F【：:
하【RT
F'TR帆조共댔盤九』
r.V.rs뎡I'I…，UIU'r
004010IE7
nL.E(TEPTRD잇:ES·：+402요In」
4--IL,.
e뱀‘234r。'7
TTTTTTTT
55555555
JF一하拓RTr.V.r$.h.eO4el0D3
C"PPL.4
JHZSㅐt】RTr.V.r'.h.0e4010D0
1"CESJ
IHCE鱗＜
J"P하柏RTrev순r＄순h.00401戚緻
cllPESI,6
JL하【효Trever룔삵1.004010F7
J千r샅v슨rse".00401205
DB00
DDoeo鱗珌00
OB00
DB00
DB08
DBoe
DB00
OB00
DBOe
EFL
EE
「Eoe
lF
28010000
r．■■．·■·＇놈
J·J->＾·＞>.갼
FST
FC닒
一·eeaeae。Bsee--
DDDFOI-蔘68DE234r구6781
ㅡUCCDDbDDDDEEEEEEE-
=■Uoaonplba긴000凸U白pdpO000-
’·｀''.．쵸펩■춈·．흩·■솝．■욥．■욥.．춈뱁욥·．춈．·홑·．욥·■춈·■춈‘■욥.｀춈
너■너1000-UOO白peap0000apOI
44444444444444444
뎌■너·000－너O00000000a킵O-
너■너·色papaUIDn긷白paU色paU白디aUaUnUapnp-

Let's not jump here because we first need

ESI >= 8

Jump하면 안 된다. 왜냐하면 첫번째로 우리는 ESI가 8이상이 필요하다.

What's up next?

무슨 일이냐?

Aha, notice that AL

(which is our first byte read, remember) is compared to 47h

아하, AL이 47h와 비교됐다.(AL에 있는 내용은 우리가 처음에 읽은 byte다.)

"But what is 47h ?" you ask.

Well, every char, number, sign, has its hexadecimal counterpart.

(See explaining in info's before).

"47h이 뭐야?"

모든 문자, 숫자, 부호는 16진수로 이루어져 있다.

(이전에 설명했다)

I have prepared a html page with these for you and also included it in this package.

나는 이미 이것과 함께 너를 위해 html page를 이 package 안에 포함해서 준비해뒀다.

BTW, there exist also tools to do this. A very good one is "Crackerstool" which I have also included in this package.

Remark: place the included comdlg32.ocx in the same dir if it doesn't run on your machine.

By the way, tools가 일을 하기 위해 존재한다. 매우 좋은 "Crackerstool" 이다. 이미 이 package에 포함해뒀다.

주목: comdlg32.ocx를 같은 directory에 포함됐다. 만약에 이것이 없으면 너의 컴퓨터에서 실행되지 않는다.

시스템 생성 대체 텍스트: 凹凶O
This15thechartWiththedec/hexZascii
representatiㅇns·(Seealsㅇbefㅇre).ㅌachbyte
蝦．驪
「OOr〃쵸SV)ecZHexZA
canhaveavaluerangingfrㅇm00toFF
DecHexASCn
DecHexASCn
DecHexASCn
a b
Ic匡LF口FF口
nU ．■1 하‘ a구
rU rV <V <V
rU 착I OO Cy
凸걋 a7 C7 C>
흉ABC
DecHexASCn
0 00 NUL
I OI SOH
Z OZ STx
.기03-ETX
「r脯－툐bT
64 40
65-41
66 42
'67143
E프첵柳
、뗀쁘쌕l03
阿阿H『
SP!· 。꾸s
쇼U 뱀■‘ 勺－ ajes계썅
竗I 魯I 魯－ 鑽I…讓I
47h==Ginascii
竗I 솔J l빵 r｀닐es<U겯
esa그 墺J 섯J 魯그esaJ 71
5 05
Indeecl:
O6
「棘걋－
ACK
l推규
朽汁8
円뚜。、
48…49…4A4B4c泌
NOP
E F O
4 4 rJ
勺一一섶Jesl썅esr그 rU 、1 00 C7 쵸U
캬／-뺑＇!캬／-빵＇ 착＇ 뻐（、 착＇ 착／ 00
「：苽－陣r 阿陸，…
一그I匯陋〔닙一匹呱〔毓
InourexamPI티47h(equals71d)
standsfㅇrtheletterG(inasCii)
BCD ErO
讓－ 魯－ 야- 魯I 魯－ 슬」
rU 、＇
曰빤더큭
WFFCRsosl
11 OB
12 OC
13 OD
14 OE
15 OF
16 10
듀－Yr
18 12
19 13
-43
…44
…45
+
石T동「
「一一一一一
1()569
1066A
1()76B
1086C
1096D
1106E
1116F
「k
l
DLE
DCI
…48
阿阿「1
「一－--
11270
n]7r
陸百
DC3
50 32
51 33
〔阿「『
82-52－「冗一
83 53 5
11472
11573

This is the chart with the dec/hex/ascii representations. (See also before). Each byte can have a value ranging from 00 to FF

이 chart는 dec/hex/ascii 표기법이다. 각 byte는 00~FF 까지 범위를 가진다.

*번역 주)2진수, 16진수 이해가 안되면 힘들어요. 쉬운 책부터 시작하세요. 꼭 2진수, 16진수를 이해해야 됩니다.*

In our example, 47h (equals 71d) stands for the letter G(in ascii)

Indeed : 47h == G in ascii

이번 예제에서는, 47h는 (71d와 같다) G 문자를 뜻한다(ascii)

47h == G (ascii에서)

시스템 생성 대체 텍스트: UU댜Ui
00401
00401
00401
00401
0040'
00401
00401
00401
00401
00401
:1더4ni
.1ㅑ14冬쌩l
너너4UI
00401
00401
00401
00401
00401
00401
00401
e04OI
ZF02eOOO
C0
02
43
DB
F6
3D732i4eeo
36
83IR214000
ee
08
11요「뜨簒,7-nULㄴ
('"e욥。「LL낟
Mmmm,sㅇtheReverseMeverifies
ifthefirstfㅇundbyteiSG
reVer
UsrorOB333CnHC4
bE87E3387837
…갼갇＞··갼＞·J
』■■■■■【
「Ub" LH즈
Cn【．L<J"
TEST타拐
JHZ하솥DR
J"P하송DR
XORE마《,
XORESI,
C"P며）OR
JLSㅐ（尸T
ㅒOVRL,B
C"PnL,D
JE와蹈RT
CNPRL,4
큔겯뻔’'-
r.Ver'.h.8040leD0
ESI
E마＜
하柏RTreVer륭슨h.ee401여〕1
ESI,：뜨
JNz匯甛硼ChPJL硼
》 ^
■■r · ·노Z ·노Z
0000
8 1
e 0
1' E FS
ap EEI'2
F
r걍63B3C90
744E87E0
00000000
匹：
DD
하蹈RTreVer슝삵1.e04e10F7
rㅎv。r'eN.00401205
0e
00000000
緇緇e82eB4eB6驟ocl0C7뇨ljl-L10cFOD80D10DSeD6eOS溫

Mmmm, so the ReverseMe verifies if the first found byte is G

음, ReverseMe는 첫번째로 찾은 byte가 G인지 검증한다.

시스템 생성 대체 텍스트: J■■맥·탭■·J■■··J탭닐맥'· ■＇.'- 섭’탭 귀딕L쫴J탬J체J체귀J맥令、새∥｀'궈치딕계∥、、궈닉디
하솔DRTreVer룔e".oe40leB4
하抑RTr'v슨r'eh.oe4el0F7
E테＜,EBX
ESI.ESI
732140001
Mmmm,ifthebyte15not"G''
擺XORXORChPJL溫JE淄綴部JL棚認
02
43
B6D
DF3
36
83
in214000
PTRDS:[402!73],le
r.v.r룔e".ee401eF7
TEPTRDS:【EBX+402llR】
thenwejumpoVertheCommand
r.v.r$.".OO4B10D3
開盼47酊
“〔
rO8aO33CpC4C
7E358783떻r3
데勵RD
驪（RT
nL.BY
RL,0
하【RT
RL.4
하拓RT
ES!-
EBk
INCㅌ51
Let'5SeeWhat!NCmeans
>··갼＞·
oe4010Be
0040!0B2
004010B4
004010B6
004Oi0B8
OO40lOBF
OO4O10CI
004010C7
004010〔·긔
n〔140【더EE;
OO4t:tl이F
004010D0
.:IO401ODI
ee4ele03
eO4OI0D6
004010D8
0040!ODD
OO4e1ODE
nn4nl더「夕
4〔
4뜨
EEEE
saFE08
7C!F
Eg28010000
O0
000000Oe
더슭
앉柏RTrev순r＄순h.00401毓〕1
ESI.6
하抑RTreVer룔삵1.004010F7
r샅v·rseh.004e1205
00
oeo鱗沁oe
qn

Mmmm, if the byte is not "G", then we jump over the command

만약에 G가 아니라면 명령을 jump한다.

INC ESI

Let's see what INC means

INC 의미가 무엇인지 보자.

ASSEMBLER INFO:

INC (Increment)

Syntax:

|  |  |
| --- | --- |
| INC | Register |

INC is the opposite of the DEC instruction it increases values by 1.

INC can set the Z/O flags.

INC 반대는 DEC 명령어다. INC 뜻은 값을 1만큼 증가시킨다.

INC는 Z/O flags를 set한다.

시스템 생성 대체 텍스트: I『lshㅇrt:INCincrementsby1.
RememberthatWeneedtㅇhaVe
ㅌ51>=8
－·－14rn'히」
rL겯’ =
F-E:=．·j윅 -.
r·E凸수＞궁요l
VV
· ·노7 ■ ·
1..*』
14「1[
14ㅑ11
l」+'"
→b l너10一
덕－꽈·－－너－너
r f·' 뺑·
IR' R ㅛr
ot'U쇼OIR
LHSBHSO
nSEESEH
S
P뱃JCCPP P
hㅐnkjnkjn"hL"
CJllJCJJ[1니、긔
·Ll卜11141
rr르＝Or·
갇、2·J
－너＝'4、뇬 心Flr
rEELr[-ㄷEELUL·I=
너=-=니너－너I－나ㅡ
ㅢ4이길
J401
니O요
rJ【브1-J
一F
'·꽈ㄷL'·-qrCU
4ㅓ뺙EO·一 EO
r■■놀
V ^ 》V
를 * …
山孩즈竅，’訂니〔■
r.V'r
reV
reVe
r손V슨rs슨N.
려
.,길1긺'L'rL·.'
Andher터weincrement
thevalueofㅌ51byㅇneif
wedㅇNOTjumpoVerit.
Let'5stepagai『1.
맬뺏맺
'· HO닿DE-Jl·:
Fㄴn나n너rㄴrnFJ-∼

In short : INC increments by 1.

Remember that we need to have

ESI >= 8

요약하면, INC는 1을 증가시킨다.

기억해. 우리는 ESI가 8이상 되는 것이 필요하다.

And here, we increment the value of ESI by one if we do NOT jump over it.

Let's step again.

그리고 이곳은, ESI value를 하나씩 증가한다. 우리는 이것을 jump 할 수 없다.

계속 해.

시스템 생성 대체 텍스트: 『＇'.,'-
하섯떰T
RL.47
하솔ORT
ESI
EB낭-
하솔ORT
tever$eh'0040
r.ver‘솥h.se뒈
印XwaszerO,butiS
nㅇwincrementedbyl
r·vers순h.ee
ESI,프 \
하섯尸TreVer슝eh.OO401
r솥V솥r흡e".Oa40120S
너추첵
CㅇntinuestepPing.
r PZCCPP P
∥Ehㅒnㅒ11HH∥HnL"긱
닙JCJllJCJJ너
(
륵―
一
441나4
－너＝너■너·-"
EE
FE08
!「
2801
B3Cgn
E87En
·＞갼갼
■■■■1·■흼■■，1■1■■
·－닉bDDF너1ㅗ층걀68n
ㄴ-L―·■l-r-ㄷ…nLDDDn
'-lI--1-너너n·b…－UoaoeR
ㅗ탬‘一’■‘1'｀ㅗ·1'뱀ㅗ’횹·．춈.'.．■
、－l4-lwe너UL'l-니I넌-UOOn
驪

EBX was zero, but is now incremented by 1 Continue stepping.

EBX는 zero, 그러나 이 step을 계속하면 1이 증가된다.

시스템 생성 대체 텍스트: 2468Ft'
RCHnHCHCnBBBBBBC
dpo습U凸U伋UOO凸O00apnp
gL나D
00401
00401
00401
O04e!
00401
OO4ei
00401
00401
00401
00401
Oa4Oi
嶼〕4el
00401
00401
00401
앵珊ql
00401
0040[
09R
09G
Andwejumpbacktostartthe
verifyingofthenextbyteㄴ
Let'5seetheimportantcode
> 33매 Z》／
33F67二낟＇
83욥貶t크U쑈
.v7며P36
004e1OCr
004010D0
．伊수O노더El
OO4OIODa
0040!0D6
004010D8
0040!eDO
O04e10DE
nn4nln「夕
0000
XORE하《，EBX
XORESI,ESI
C"P뎨【ROPTRDS:【402173〕,요0
JL하（RTr.v.r$eh.ee4e10F7
"OUnL，텨，TEPTRDS흩[EBX+4021In】
CllPRL,O
JE하섯）RTreV손r를e".004810【）3
C"PnL.47
J"Z하勵RTr.v.r''".804010D0
IHCESI
1’결〕EE::
J"P하緞楡Trever룝슨h.0040iOCI
ChPES1.욥
JL하贓）RTreV슨r룔eh.O04el0F7
JPrtV솥r별eh.OO401206
DE:0t:1
DDOoe00000
nPnn
3Cgap
87Ee
Oee00000
더白

And we jump back to start the verifying of the next byte.

Let's see the important code

그리고 우리는 다음 byte를 검증하기 위해 jump 한다.

중요한 code를 잘 봐라.

시스템 생성 대체 텍스트: UddU r驪《1'ppb iU
s硼변泓밂籤1綸碌분”
RL.타7TEPTRDS:[EBk+4021IR]
RL-‘〕
하拓RTreVer를eh.OO481eD3 '
nL,4
하勵RTr'v.r'.".se4e10Oe
ESI
EE::
하拓RTreVerseh.ee40i이〕1
ESI,：ㅌ
하拓RTreVer슝삵1.8048i0F7
r슨vㅎrse".00401205
0e
00000000
Weneedtoincrementㅌ51
-->thiscanonlyifthebyte
8
''G''
DS
옭L ·Lr
r驟侃擬綸ChPJL棚DD
n밍
a랖
牝 ee
肥 8泌盼
61·871 EeF88盼
33eaO4e EE12 e
S F e
7c8PSC겨Sc7S46枋EB貂7cEgee卵
··＞더·＞갼α
UU냐UIUbU
004OIOBF
004010CI
0040i0C7
OO4elOCg
004이aC탸
ee4o1OCD
며쵱후F넌O「F
1'-68DF
ㄷ…ㄷDDDnL
nf…口·ao凸pdU
'ㅗ섭IL.'뱁‘.．춈1쵸
n긍너ㅡ…ㅡ너－nUnUnUn킬
4…444444
凸긍U臼·0000
더습·1－…－·1·aU麴UaU、날
-

We need to increment ESI to 8

--> this can only if the byte is "G"

우리는 ESI를 8로 증가시키는 게 필요하다.

--> 오직 byte는 G만 된다.

시스템 생성 대체 텍스트: 020000
LRe詢卜L【e--－一一
LU' rb71Ur겹
r' r'·．J
O닙빻．
CRLI.
TEST
JHZ
(J뿌．各KEF꺼EL32.R슬adFLle>
ERX.티놈X
하솔ORTteVerS황h.ee40leB4
하緞RTr'vers손h.ee4el0F7
E하＜,EBX
ESI,ESI
Andifthebyte15zero(enclofthefile),
it15Verifiedifㅌ5115atleasts
7aZI4Oae
iRa14eOO
예예DPTRDS:t4e2173긔le
8ㅐORTr.V.r冬en.004010F7
RL.타군TEPTRDS:【EBX+4021In】
RL-‘〕
하【RTrever$eh.OO4010Q3
nL,4
하勵RTr.v.r!’∥．se40leDe
ESI
EE::
하緞RTrev슬r출삵흼．e04010CI
ESI,8
하緞RTreVer휴야1.804010F7
r슨v슨r$eh.00401205
00
OOa여珌ee
蠱蠱
-->Werethereatleast8times"G"?
擺XORChPJL驟JE髓1HCI'I:’邵JL棚옜
F 23 ' e871
20aO4B6033·e4IU
C DF3 8
騙貂花E8貂貂馹7C覇SC74scX·46柏EB8a7cEg開
··갼J>··J>·
HHrHㄴr간
OO4OleF19
OO40leRE
OO4010Be
0040!0B2
0040!0B4
004010B6
004el0B8
.:104O10BF
.:IO4O10CI
.:10·b〕10C7
O며a媤Cg
·1'14더1聊：E:
,1414010CD
,1'14O10C「
더n4nln「n
EE
「EOS
l「
28010008
·＞J갼
.:10·추010〔므
:1더41--11‘갸다프
0040료0D8
0040!eDD
004010DE
仇仇杜蝨I白「츌
Alsㅇrememberthatinthebeginning
wasverifiedifthereareatleast16d
bytesreadinthefile
O000O00e
더a

And if the byte is zero (end of the file), it is verified if ESI is at least 8

--> Were there at least 8 times "G" ?

그리고 만약에 byte가 zero라면(file의 마지막), ESI가 8보다 작은지 검증됐다.

--> G의 갯수가 8보다 작다면?

Also remember that in the beginning was verified if there are at least 16d bytes read in the file

또한 기억해라. 처음 시작할 때 File에서 최소한 16d bytes 이상이어야 읽는다는 것을 검증했다.

시스템 생성 대체 텍스트: VP
ㄴOHll
나"C
C4Cn·'gBa겸CgaO
373744E87EaO
*-ll-lㄴfl-l人l-l亡뇬
恥떼口덥．긔
.:10·41:11디E∼
乙r틤큐띵I r'V'r돌톱nl·00,OIU『r
ZC(-PP P
"H'IhHllLh
JllJCJJ
글하―--·
0040뙨l“…－》
.1。14너1더「
.1너4너헨니더
.1.14뗘니…EF
〕더4011〕더〕
〕O·수0노eEI
,-1,-1·b꺼11꺼「-
t닙 콥O
8R83iRal4000
RL
RL
.B모TE
,U
PTRDS:[EB：《＋402llFi」
00
08
47
0l
EE
「Eee
lF
28010釀蹈
갼＞·
JE하拓RTreVer룝e".O040ie
CllPnL,4-
하【RT
ESI
EE:::
하刃RT
ESI,프
하柏RT
r.V.rt.h.se40ieDe
논
reVer
reVer
r슨V손rseN.
DㅇyㅇuunderstandthatㅌB×hasther이e
아…counter구EBXwasincrementedtㅇ1.
Thisvalue15addecltㅇthepointer'svalue
fromthefirstreadbytㅌ｀50,wewi∥verify
thesecondbytefrㅇmKeyfi―ㅌ｀datnow.
Ifthis15notclear,takeyourtimeand
studyital-overagai『1.Let'scontinue·..
·＞갼갼
一eeee··Be·eees·、ee、

Do you understand that EBX has the role of counter? EBX was incremented to 1. This value is added to the pointer's value from the first read byte.

So, we will verify the second byte from Keyfile.dat now.

EBX가 이 role의 counter라는 것을 이해했어? EBX는 1씩 증가된다. 이 값은 첫번째 byte를 읽어 pointer 값에 추가됐다.

그래서 우리는 Keyfile.dat에서 second byte를 검증한다.

If this is not clear, take your time and study it all over again. Let's continue…

만약에 이게 명확하지 않다면, 너의 시간을 들여 모든 것을 다시 공부해라. 계속하자.

시스템 생성 대체 텍스트: 그륵―…
00401이뜨1
1:104010〔7
> ：뜨R광크1只alㅓeeo
Onceagairl:iftheverified
byte1500(emPty)．··
·l;1I'l구71
·1;11·－140
C4Crr)'383C9-
=.')’·3744E87E-
r■■■■■■．■吟
00401더［E:
O()4010〔O
a040101:「
L:1I〕4010D0
쌜의
401
g
D
OO4O10D6
004010D8
.뇩7
·녹r
EE
FE08
lF
2801000e
＂毓InL.E(TE＝ㅜRD읽[EE,:+40크쵸In】
12llPnL,0
JE하拓PTrever＄슬h.00481003
CnPRL,4·
JNZSㅐDRTr'V'rs'h.ee4eleoe
IHCESI
1『卜〕EE::'
JllP하拗RTrevers슨h.OO4O10CI
CHPES1.「
JL하緞RTreV슨r흡eh.e04e10F7
J"Pr솥V솥rs.".00401205
드T:
ST3
ST4
STS
公T'

Once again : if the verified bytes is 00 (empty) …

다시 한 번 : 검증된 bytes는 00 이다. (비어있다)

시스템 생성 대체 텍스트: 驪矗擬Illl?JhPCllPJL棚
--!뺙―
8R881Ralㅓeeo
- T T·
T4R R 'T
驪騙땠ESIEE::！洲ESI驪
OO
.-1」
47
Ol
EE
FE08
lF
28010000
qL.E:'（ㅜEPㅜRDS:[EE:：·：+402쵸IFU
RL.I'
r.Ver흡eh.004010Q3
r.V'r'.h.oe401晄炤
....thenwejumptoverifyif
톄iSequalorbiggerthen8.
reVe
r슨Verseh.
0e
r슝삵찧．804010F7
00401205
l:4C二·'3B3Cge
콩날r·3744ES7Ee
r·■■■．■．■■U
-'2··…＞＾·＞J갼
雕灐

…. Then we jump to verify if ESI is equal or bigger then 8.

Scroll up.

그래서 우리는 ESI가 8보다 같거나 크다는 것을 검증하기 위해 jump 한다.

Scroll 올려.

시스템 생성 대체 텍스트: 004e
.:1040
l勿디.^E텨邙 p"P盼며T
노드」> 83FE08 -L다IPESI-〕
n國미.vr쵸추IF .rJLSHORT
r.V.r冬슨”.0040i0Ci
reVer드etl.
.vlEg280100e0
IOB
J"Pr슨verseN.00401
DB00
DD0000e000
DB00
DB00
Dg더n
ee4010F7
Ze드
AndifthereWere
notyets"G"
thenwejumpto
thebadboy.
一ee··、·ee··
.:1이b〕p〕DE
0040!OD〔
.1'1411표l＊「「
.1;'4띠‘·U
ㅢ∥4011工
.:1.:140二DE4
-1－口'-'l-「ㄷ
11.1 1·
11 ' l
1 1
길쵸긴n빕仕p
UdpdUaU
니BBB
날DDD
0e401o「2
OO4O10F3
004010F4
004010FS
0040!eF7
004010F9
004O10FE
00401183
00 …
00
e0
0e
ee
e0
0e
e0
00
00
EB00
)6Ree
6800aO400O
6886204이泌
6Reo
ES5D0200e0
ESRROI0000
Ege9010000
J"P하緞RTrever룔eh.e040i0F7
PUSH‘〕
"a－여（I『I8.-RPPL戚汨RL
r'Ver맬
*U슨r입
e".OO4OZeeo
.".eO4eZeS6
[
)t닐
ㅜ't
T〔ㅐ
【슨
t'
t
【eReV'r맬솥h."
isn。tvaL(d.s。rr,...
ho니rlel
"K.,F
"Ke,f나。
:NULL
00401105
0040110p
JP
｀洲P
.몫US
.흽KE
00401!OF
J"Pr슨ver드eN.
ER32.he$'ageBoㅐR>
맙fLSZ.Eㅐi零PrOCe5S)
0040121D
nes$ageB。HR
EㅐLtPrㅇCe륭S
rnpZ녹r노
SHSHsHLLLL
UUUCnnn
PPPCC
V V
·노Z · ■ · …
…………………………………………

And if there were not yet 8 "G"

Then we jump to the badboy.

그리고 아직 "G"가 8이 되지 않았다면 우리는 badboy로 jump 한다.

시스템 생성 대체 텍스트: P하緞RTreVer륭e".8840i0Ci
P P
님nHllL"·
-JCJJI
｀恪짠＇뜨．.‘】．
0040lOD8
t-rever$e".ee4010F7
erseN.0040i20드
ESI'
ㅐORT
∥1.l卵卵嵋沁盼觸
EBEE
83FE08
7CIF
E')2:E:010000
0e
00더O000e
BBBBBB
DDDODD
티S타wejump
tothegoodboym
SㅐORTrever룔슝＂.0e401eF7
너
r'ver'e".004e20e0
reUer'e".OO4eZe8B
O
<J뿌．얹USER32.hes$.geB애R>
<JP.흼KEF꺼ELSZ.Eㅐ죙tPrㅇOe룔S)
r슬uer$eN.0040i21D
st,te=＂탸－여（:r뙤－RPPL『
T'tl·="Ke,FLleR.'
T.-;t="Ke,fileL'n。t
ho’ㄲer:HU【．L
"es$ageB。xR
PSHSHSHsH
UBBBHllUUUU
VDDDJPPPP
004010E3
004O10E4
004010E드
0040lOE6
004010E7
0040!0E8
0040!0E9
004e1OEp
0041:11OEB
OO401OEC
01:1'4010E「
.T1더41-1InF
.1。14추1:1
.1.14ㅑ'
길닙국딩'딩．'
0040iOF2
OO4O10F3
004010F4
004010FS
004010F7
0040!OFg
00401eFE
Oe4O1183
0040110드
OO40110R
0040!1eF
00
00
00aO4000
86204000
OO
5D02Oe00
nRe10000
09010e00
CRL.L
CRLL
J"P
EHLtPrOCe륭S
[
開腑매四開開開四開開開開에卵卵觸e0四卵EB胡6868鹹ESESEg一
굳’흼흽Ies·．.eses흽．.eses
‘· · 갼＞··…갼
<l.lll
0040120드＝reVer$en.00401285

Else, we jump to the goodboy !!!

아니면, 우리는 goodboy로 jump 한다!!!

So, let's avoid this situation ;)

We have enough information now to make ourselves a right Keyfile.dat

우리는 이 상황을 피하겠다.

우리는 올바른 Keyfile.dat를 만들 충분한 정보를 가지고 있다.

시스템 생성 대체 텍스트: 乙驪rF
ㅒnHIIHIILHllBDBBBBBBBBBBBBBBPE:BBBHIlUUUUnHnnHlI'
IJCJJDODDODODDODDDDDDn令구ODOJPPPPCCJ、
盼EB田7CEg卵
>＾·＞什갼
00401ODe
004Oi0DI
004010D크
r－一·－므＝=＝므＝尸브므므므뿌므－－므’브놈귿｀'''====＝→＝∼∼＝--∼
0〔I'40
0040
004e
0040
0040
0040
0040
0040
0040
0040
Oe40
e04o
0040
0040
0040
ee4o
0040
0040
004e
0040
0040
0040
0040
0040
0040
0040
EE
FE08
lF
28010000
더nnonnn더
.Kevfile·dat-Kladblok
Bes山ndBewe楸enOplna毓Be님dH리p
GG아刃Gq沁p000000000O00q
/--－一 一一一一一一一一- - ―一一-
ㅌxactly:atleasts"G''5andatleast
16bytesal-together.Therearea-otof
goodKeyfi―이dat'5poSSibleofcourse.
NㅇW,Savethefileand....verify!!!
88nH889C
6ZO6EEEe
V
… …
103
105
10R
10「
·.J별
00
80
츠aaa强po
882a4이泌
80
5D020000
RRO10000
e9010000
^Uaon긷ononononUnUaO0nonoeaonUnoapnOB
aU凸OeapeaO白Oap。ps仕p白UdoaododpeaOSE
DE234rD6789nHBCDEFnO뱁12a규4rO79E
DDㅌEEEEEEEEEEEEEFFFFFFFFF
oa다쵸UodUnpaO仕U쇼UO0dU色podUO쵸Ooaonp繭p00凸UO
溫刪一꾀
醱00매嵋一r
Ju「pt'taken
004el0F7=reVerseH.004810F7

Exactly : at least 8 "G"s and at least 16 bytes all together. There are a lot of good Keyfile.dat's possible of course.

Now, save the file and … verify !!!

정확 : 최소한 "G"가 8개이고 최소한 16bytes 가 되어야 한다. 좋은 Keyfile.dat를 만들 가능성이 꽤 많다.

이제, file을 저장하고 검증하자 !!!

*번역 주) 저는 이곳에서 꽤 헤맸습니다. 첫번째 조건은 G가 최소한 8개이고 두번째 조건은 최소한 16bytes 입니다. 두 가지 조건을 모두 만족해야 됩니다.*

Save the changes

바뀐 것을 저장해라.

Restart the ReverseMe

ReverseMe를 재시작 해라.

1. **Testing the keyfile**

시스템 생성 대체 텍스트: hT슨휘p【ateF"슨=HULL
RttribUt。｀=RERDOHLV:HIDDEN:S무STE":RF
h어늦=OPE『LD汀STIHG
psecur．치흩
ar'"O」．
Ce료술 '
·■■■··■■■■．．■＇■■
딥겯띱，-V띵』'팀튠、7.
004010드C
004010드E
00401063
00401065
00401067
00401069
0040!06E
0040_073
〔ㅓ;I -一
YeP,thejumpoverthe
firstbadboy15taken
스 닐．·＇■｀''' 己｀
느 2 0
·0F3白p30
n06eeee
792O400e
OBOZOaOO
11눠콰건쨋.1
U04ele7F
·’·'4·나0C4
-”「”·눠나eCg
OO4O10EE
Oe401090
0040!0드5
0040109R
0040109C
O04Oionl
004010n3
OO40ions
004010R9
00401OnE
r녹r녹눼r놈'날딩『닐J졈
PUSHreuer$e".004e2079
CnLL<JP.各KERt茫L32.CreateFi【en>
C"PE아＜,-i
JHZ하柏RTrever룔e".0040109p
PUSH너
PUSHr'ver'.".oe4OZeeo
PU'3Hrtvert.".eO4OZe17
PU3HO
。룁L(J뿌．'USER32·hes'ageBct::R
CnLL(J尸．各KERHEL32.EㅐL七Pt。c.;귿』
JnP鷺슬ver$eN.e040121D
Fi!쥰Ha'즌
면NULL
=FILE-SH叩E.REnDIFILL하뛰차
GENERI익R타D:GEN티국Ic－닙R1TE
="KeVf11..dat"
CreateFi【en
‘·뱁b닙
ㅣ
St,lㅎ="8－며（:he一RPP나섯乙R【．
Title="K.,FILeRe臘r∥＂e''
Te卄t="EvaIuatL。nperl여」얘t。「dat＊·
h（눙ner HULL
"e'＄술geB。ㅐR
E卄LtPrOCe드룔
0
r'ver'.n.0048217s
4F
r빵ver륭．".OO4eZ1In
타礪
<JP.'KEf꺼EL32.ReadFI【e>
ㅣ
이沁er【apDed
ytesRead
=NULL
=reV.r$eH.00482!73
tetTㅇR.
.d'45(7e.)
r'Vers'".e84e21ip
e=0e00004e(닐Lndo닐）
Re毓IFLte
티枕．ERX
SHSHSHSHSHLLST!
UUUUUnHE뻐
brPPPPCㅜ·
孺鑑柵淵一
D겻뉜n즉L더슥r4I,IU勺요‘nn F C
円l요眄1더·DZ·닝·uz41ZCec
r,－胡68胡胡胡6868E8貂花매盤柚태아ESEg애68해68騶E8邵－,~
<兩-
J'hpistaken
00401OgR=r·V솥r'.H.0040109p

With Keyfile.dat in the same directory as the ReverseMe, let's follow and verify

Yep, the jump over the first badboy is taken

Keyfile.dat는 ReverseMe와 같은 directory에 있어야 한다. 따라가보며 검증하자. 첫번째 badboy를 Jump했다.

시스템 생성 대체 텍스트: - r■■■■■■■■L
0040le95
0040109R
OO4el09C
0040i0Fll
004010F13
OO4O10RS
OO4010R9
.1.’미1쨍',HE
83010000
ee
73a1400O
46
IR214OeO
YeP,Keyfi―ㅌ｀dat
WaSread
→――펴―-…
2「020
CS＇겸
驪贓＂【apoed=NULL
p햐t.$Re.d=re"er$e".OO4e2173
Byt"T。ReacI 46(70.)
Buf「er인reve脯슬h.e0482llR
hFi흰e=00000001
Re추IFile
닐gnH8psnO8ro
-E66665E8
3 6 e87뱀1
4B6D33ee4e
; DF쵸구 8
……갼갼＞··
·1't4t-1
.:1더40
0040
0040
0040
0040
0040
0040
004e
0040
0040
0040
0040
0040
004e
0040
0e40
0040
0040
e040
0040
0040
0040
0040
0040
0040
0040
73Zi4O0e
1F松1’홍00e
EE
FE08
lF
2801eeee
000000
333CnHC4CLr)'383C9eO잉noaps仕pnUa0000a팅C
3387837a걍'r44E87Een깊dodO0aoedoaoapdpe
갼＞·갼-갼·＞＾·＞갼갼
'468F179BDFel368DE234rp6789RC
ㅌEBBBCCCCCCDDDDDDDEEEEEEEEEL
니Udpdp00e000nU쵸UOdU凸U00白p仕paUO쵸Oodpapdpap^
·令J딩계냘첨
궂
Juhp
0040
아（L《JnP.乙KE째벵EL걀굡．ReadF딩Ie>
TESTE댜《，ERX
J"Z하솔DRTr.V'r흡솥h.0e40!eB4
JhPSㅐDRTr.v.t룔’h.004010F7
k0RE鱗t.EBX
XORESI.ESI
ChP며勵RDPTRDS:t402i73],10
JLSHDRTreVer술e".ee401eF7
"OURL.타군TEPTRDS.【EBX+40211
CllPPL,.
JE하섯）RTr'V'rt'n.eO4eleD3
C"PnL,4一
J"25ㅐORTr●ver'e".0e4010D0
1"CESI
IHCE마《
JllP하勵RTreuer룝슨＂.oe40iOCI
C"PESI.;3
JL하細RTre닌．r맬．".e04eleF7
JllPtever룝e".e04e12e5
DBeo
DDOe000000
DB00
OB00
DB00
OBoe
DB00
DB00
OB00
DB00
OB00
「b「닐 뱁팀r눕
(Staken
iOB4:r·v슨rseh.004ei0B4

Yep, Keyfile.dat was read

예, Keyfile.dat를 찾았다.

*번역 주) JNZ는 EAX가 zero가 아닐 때 jump 한다. TEST EAX, EAX에서 Z-Flag 값이 1이 아니라는 말. 즉, file에 아무것도 없다면 ReadFile 후에 EAX는 0이 되어 TEST EAX, EAX를 하면 Z-Flag가 1이 된다.*

시스템 생성 대체 텍스트: ee
73214여泌
46
IRa1400O
2F020000
PUSHO
PUSHr뺨pert.".eO4eZ173
PUSH46
PUSHre넌·r$e".Oa4eZllR
poverlaDOed
yt'$Re毓j
te'T。F:E瑯〕
BUffer
hFL!e7
reVerse"
뜨e00
Hㅇwmanybytesareread구
ㅣ
eio「7
緇：:lR써Cr1PJL刪醱JE醱
23 6 0871
np04B견bD33白Oe4e
C 灼니F轟큐 8
'nH8nH808SrOB습－.=：·8댜CnHC4CrD63B3Cgnon딩nodoaoaoenoaO0apC
-6666rOE87EOI=·878373744E87Ee白다dodU白Oeeeaodpe^
·－>··…、갼갼＞··什＞
0040109R
0040109C
004010nl
0040ion3
0040i0R8
004010R9
004010RE
하拓RTr.ver$eh.e
EE::(,EE
ESI.E·El
7321ㅓeeole
IRa14O0e
마勵RDPTRDS:〔402!7
하緞PTre난et룔en.ee40i
너시너les－·．1es너
RL.테군TE
RL,p
PTRDS:tEBX+4021ip】
EE
FE09
iF
28810鱗拗
하抑RT
RL,4-
하勵RT
ESI
EBX
하수DRT
ESI.F
S태저T
re튁』즐’
r.Ver'eh.e0401eD3
r.V.r륭e".ee481요DO
r횬v순r＄야】겻泌40iOCi
r.Ver｀낟”.ee4eieF7
건軸e,thereare
17h====23dbytes
inthebUffer
咫駝駝飇mL飇‘DBP∼E:,
JllJCJJ-JDDD一D
·六·갼·＞＾·＞갼갼
ㄴrll.I＊ㅗ ·
HLLL 니
나나 · －너→HH Iㅑ
oe000e
[0040손173]=00000el

How many bytes are read?

Hehe, there are 17h == 23d bytes in the buffer

얼마나 많은 bytes를 읽었니?

헤헤, 17h == 23d bytes가 buffer에 있다.

시스템 생성 대체 텍스트: 겨0시〕l〔·[
85Ce
너，1401OB
.v7502
.vES43
J)4el〔댜4
乙中추01〔尺7
.10．구O_〔E프
．爻』B
3F6
광광D門
C)a
a꺼eeo
TESTERX,ERX
JHZ하돌DRㅜr.ver’판h.e040i0B4
J"P하蹈RTt.ver'e".0040i0F7
굼OREB:<,EEX
：《ORES1.E·드I
1,:l.DnP휀」IDFㄷFT므DS:rㅓe욘17D그.10
IR214000
dO871'
3noe48
8
EE
FE08
!F
2881
JLS냈RTre난er$eh.ee401eF7계■「
"OURL,珊군TEPTRDS:〔EBX+4021!P]
CllPRL,0
JE하【RTr.U'r!'n.ee4eleO3
C"PnL.4'
Jn' bH나HI reVer'e'1·UU댜UIUU넌
IHCESI
IHCEBX
J"P하緞RTreVer슝슨h.0040lOCI
一[No」umPingtothe
W
ㅇ
n
y
ㅇ
b
d
b■남·이啖啖髓一
、螺灌睡
鹹3C74sc花46柏EB的7cEg騁卵0e叩매매騶e8애觸開一
I V V ^ I VV
·노7 ·……노Z··노Z · ■ ·
HOTta홧en
緇溫溫縕0l0Dl柵溫漑柵el0E5elareelareel0E8o10Eg凞。se·一
밉444444444444444444444441 ■h
租開腑卵盼開盼閔騁eO腑盼腑騁盼閔盼騁開腑oo騁開一r-Ju
OO40leF7=t.U슨rseh.004el0F7

No jumping to the badboy now !!!

이제 Badboy로 jump하지 않는다.

시스템 생성 대체 텍스트: 0.:1．추0그eEㅌ
.-ll-1.401nEF
ㅌ：D門요1ㅓ0굿O
36
89IR21400e
00
e8
47
0i
EE
FE08
iF
2881eoa0
Cll.ICFㄷ
S냈읽Tr
「L.BVT
RL..
FTRDS:〔극ea17크그.10
.Ver용낡＇.e040i0F7
EPT효DS.[EBX+4숑Ziln】
.1。14'11너…
O04U10C슝
00401OCB
004010CD
0040IOCF
0040i0D0
0040!ODi
0040!0D3
0040!eD6
OO4e1eDS
004010DD
OO401ODE
OO401OEZ
004010E3
0040!0E4
0040!OES
004010E6
004010E7
004010E8
00401OEg
004010ER
r놓r놈』날눔·f놀『．「놀
Sㅐ야국Tr.u.r$eh.
RL,4'
하緞RTteverse「
ESI
EBX
+
旨 UP P
이JL閔대JE애
O
겸■홑
Thefirstbyte·..
Cr1P
JL
Oa000O
하抑RTrever＄탉1.oe4elOCI
ESI.8
S냈尸Trever’·h.oe40leF7
reper$'".OO4812es
00
0000吟u。u
OO
e0
00
00
더더
뱃‘CCP
"Hkjn"
Jlt'J
갼·＞＾·＞終」
BDBBBBB
DDDODOn니
.,.isa"G"
：·＝＝·－-7C매郞겨SC花4643EB837cE988배觸盼8e卵飇卵觸8e盼一
술｀Ieseseseseseses-,
'tOO4021In)=47f'G.
=01
npFr。h0O4OIODI

The first byte …

…is a "G" …

첫번째 byte가 "G" 다.

시스템 생성 대체 텍스트: 며〕401．〕R「
SSCS
.v7502
.vEB43
겁 요3DB
크3F6
；죠：광D
ERk,EnX
하솥ORTr.V'r맬황h.ee4010B4
아蹈RTr.Ver룔’h.004010F7
·T
苾深綴
貶：alㅓeCOla]:녀P
,EEK
.EEI
F【FT므Dㅌ
36
591麗乙1’홍000
U너
O8
47
Oi
IJI'
r10U
Cr1F'
IJE
JC"P
IJhZ
INC
(INC
a17커,10
Treuer룔eh.0040le
RL,BVT[PTRDS:【EBY+
내,U
F7
402!!R】
·之f쵸r닙.r
bp츤걍니WC닢
헴媤외■匕
.1.14LI:너
ㅣ
하艇】RTr.U.rt.n.eO401eD3
여：f ______
5추【RTr.V.rs슬n.OU4UleDe
ESI
E하＜
갼＞·J-갼
EE ∼
FE08
!F
28010이珌
하【RTrevers삵1.004el이긍l
E요1.8
JL－硼EIE:DDDBDBDBDBDBDBDBDBDB一
·＞＾·＞終갼
叩騁嵋0o脯0e開卵的盼0O一
OaO000
!tis
VVe
nㅇtempty,50
do『lotjump
沈얘驪겨sc邢鞠43EB田7cEg卵배觸閔四開四卵朋卵四一
DFnp오‘368DE234r구6789p【
CCDDDDDDDEEEEEEEEEL
doap00OaU00nUap0O凸pododU仕pdUa-
OOnU00Oap00dO00a킵e仕p00dpa-以
444444444444444444꿰
OOa킵－긴O00aUOOap00eod킵Oa빕e→
olaai沙00la0100a0010a00a轟7찧
Ju「p
0040
LSHOTt냥（슨n
i0D3=reVer'eh.004010D3

It is not empty, so we do not jump

이것은 비어있지 않다, 그래서 우리는 jump 하지 않는다.

시스템 생성 대체 텍스트: 004010C9
.--t니401나〔E
JE
」CllP
하抑RTr.U.rt.n.ee4010D3
RL.47
하緞RTr.v.r룔슬h.004010D0
!HCESI
IHCEBX
sc灌瀷
·갼·、
JNP하抑RTrever룔슨h.oe4BIOCI
·＊1:47el'EE
e8
F
SOlee0O
Itisequalto47h,
r■ ·
e
뇨H
수L
e
p
m
U
·｀■J
ㅇ
n
>
·． 치 UOO00e什
l 沙ㅓU白000白긷e什
｀介沈沁DBDBDBDBDB一
Ell2 a길
F a딩
B3CgnOO딩nononoaoa킵enUn룅nOC
E87Eea다doaO白OeeeaO仕pea
＾·＞갼갼
n·1368Dㅁ」 ∼4U구6789CHC
EDDDDDL=…ㄴEEEEEEEEL
n·ee00白U凸rn-白UoeoaodUnp白U쵸
ㅗ·■홑J·춈．■솝J■욥결．춈·．춈·’홑·1홑．■홉·■옵뱁‘Jl솥볍·솝쌥．솝·．솝써■춈·
aO4O
紛牝柏柏柏40柏牝4O牝柏
nwe0000-U白뱁apnU00
·－1wea뱝000-UaU白．nU0aU
OO4e
白킵Oapape
4444꿰
hpisHOTtJ'en
4OleDO=r·verseH.e04e1ODO
腑觸매00一r-Ju騁

It is equal to 47h, --> no jump either

이것은 47h와 같다. --> jump하지 않는다.

*번역 주) "G"는 16진수로 47h이다. CMP에 의해서 47 - 47 이므로 0이 된다. 0이 되면 Z-Flag는 1로 set 된다. 그러므로 JNZ 는 Z-Flag가 1이 아닐 때 jump하는데 Z-Flag가 1이므로 jump를 하지 않는다.*

시스템 생성 대체 텍스트: LIU낙Utl〕HL
nler'
ㅓe=
0〕4e10E：손
醜ㅣ4이0E4
*:l0401〔IEE.
004010E광
004따pBE
더더4ni白nl
.1。14너1더 '
.1너41-11''… '
JJ401너프E
.-14-14肉!白｀乙
：二
UbbU
7502
EB43
숭30B
33FE.
용L魯·
J"Z
J"P
:^:OR
XOR
l:녀。
Lㅐ즈'LH八
SㅐORTrever∥h.oe4el0B4
하艇）RTr.ver'e".0040leF7
EB:<,EBX
ESI.E‘드1
광D貶：a꺼eCO10
3'
891RZi生QOO
eO
0：〕
47
0i
Dl,II:iF:D舜T팍DS:【
·y
>
·、7
S냈nTreVer룔이1.
nL,e'.'T〔F'TRDS:
a173),-0
4eleF7
EB겨＋402!!pl
r.V'r!'h.e0481e03
r.V.r$eh.oe4eleDO
EE
FE08
lF
28010이珌
000000
r노날팀괘r놈
JL
HOU
CllP
JE
C"P
JHZ
I!IC
11ㅓC
J『IP
C"P
JL
J予
DE:
DD
RL,I갸
하【RT
RL.4-
하柏RT
ES1.
ESX
하緞RT
ESI.：근：
하細RT
rev슨r’순h.e04
Itwasa"G"
Sㅇ
r.V.r맬．h.e04Oi0F7
r뺨Vert.＂·OO4e1205
l-1n
t:101:I000OO
inCrementedfrom
ㅌ5115
OtOI
·＞갼갼
舵牝牝柏舵牝
開eO開卵開開
『팅『놀 r훰날씸
뜨7c如叉걱sc花拓柏EB田7cEg卵배一
白wel368DE씌
tLDDDDDDL
l-f、白UoeO0npn
1뱁스·■춈·．춈1홑뀀l뱁‘·

It was a "G", so ESI is incremented from 0 to 1

이것은 "G", 그래서 ESI는 0에서 1로 증가됐다.

시스템 생성 대체 텍스트: 旨 UP PZCCPP P
:'1JLI.IO띠JE대개Irllll걔대JL刪
프D
以I,:F[FT팍DS:〔ㅓea珏도
－너…''1
-UI:140
L4C드·杜3B3C9
덕·'3rt44E87E
>
7C36
9n99
7옵＝꺼O굿Ola
IRa14OeO
SH아궉Trever룔en.e
RL,B'.'TEPTRDS:
내,．· -
AndEBX15incrementedtㅇ
L
…1극….「-lㄴㅣ
1·〕401이E「
verifytheneXtbyteinthebuffer
L
SH아국Tr.v.r$ell.
RL,4-
하緞RTrever‘삵
EE
FE08
lF
2881eO0e
r슨v슨r륭e숴．oe40iOCI
r.Ver룔．H.OO4O12e드
·’'·＞＾·＞갼갼
:1더4nll--f「1
O수40퇀e【
004810D6
00481ODp
E드工
E驪＜'
하섯）RT
ESI.b
하솥【，T
r．댔．r맬샅h.e04010F7

And EBX is incremented to verify the next byte in the buffer

그리고 EBX는 buffer에서 다음 byte를 검증하기 위해 증가됐다.

시스템 생성 대체 텍스트: '1。1．주너'UHL
.1.140:「卜r
004·10E4
令D40쵸OEF
〕a·추0노eEE
더n·11-11‘가EF
UbLU
.v7502
.vES43
．즈；DB
광3F6
：드광광D7크요1ㅓeCe
7C36
하청3a쵸凡결1ㅓeOO
，〕〔00
74cl：〕
3C47
7591
48
49
EFFF
83FE08
7CIF
Eg2801OOae
aa
·Lbl LHX'LH麗
JHZ하솔ORTrever’애．ee40leB4
J干하勵RTr.ver'e".004018F7
:':OREg뱁,EE'
:<ORESI.E딩1
D盼二FDFT므DS:5홍q크l'겻,珏）
하삿’Tre나er룝en.ee4el.F7
qL,「'.T「旨T덕D하:Esv:+4枕츠츠二nl
뎨,0
하【RTr'U.r!en.ee401eD3
RL,47
아섯》RTrevers슬h.ee40ieDO
ESI
EBZ
하蹈RTrever$e".0040!OCi-
ES1.9
S냈PTr.v.r＄·n.ee4eieF7
rever룔．".Oa4elZe드
藕鐄
」L
∼
Youseethatthisjump
SerVeSasaloOP.
Q· ·jP PZCCPP P
n니LIJhE""HHhnL"O
l:Jt.ll-：맬JCJlrJCJJC
긺나
샐■스
J．■■··．■··．''-
V
…＼궂 ·
、·。·＞一終
O04ei0Cg
OO4OIOCB
004011:llED
OO401t:ll그F
l:ll:14011:IDO
l:l[:l41:1노ODI
004010D3
004e!eD6
Oa4ei0DS
白날力Aa.白nn

You see that this jump serves as a loop.

너는 이 loop에서 jump를 지원하는 것을 봤다.

시스템 생성 대체 텍스트: 「 눙－-------－커따－---－…－-------－… 뻑눌■■■－·．■－ 一
더CPtj-ma沁threa비飇dulereyerseㅆ
PUSHrtv.rt.".Oa402e08
PUSHrtu.t$.".004e20i7
PUSHO
CnLL<JP.各USER32.＂슨륭＇ages애p>
CnLL<JP.各KEf쮜EL32.EHLtPrOOe$S>
TitL슨r..K.yF'LeRev·rs삵눋＂
ㅜ翩t="EvaLuati。nperLod。utㅇ「
ho밝경er HULL
^
Oate■
나늦∥．9eB。ㅐR
EXLtPrOCe딩힁
■．■驢련트L汪크
EnXee004:10‘「
ECX液겸801898、
DOver1aDDed
聊幢菱豈「
E．크F'끄01ZFF〔4
IIP·en-·oee0L
SD'' - F
EEECPnHZSTDOE
O04OIO7F
0040ie84
00401089
OO40108B
0040109e
00401095
0040le9R
oe4Oi09C
0e4010nl
OO4010n3
OO4OIORS
0040요0R9
더더4011즈．E
68요0204000
68h,2041:100
6R00
ESD7020000
ES2402Oe80
Eg83010鱗蹈
6Ree
6e73유14여淨
6R46
68IRZ1400e
5e
ES2F020000
85C0
7502
EB43
숭－3DD
J『IP
PUSㅐ
PUSH
PUSH
PUSH
PUSH
CRLL
TEST
JNZ
J"P
ㅣ
ps닛t'
By
BU
$Read
TORead
:NULL
r'Vers.".eO4eZ173
「f순r=
hF(【e
ReadFL
re닌e
eoag.
y淋찐
001aFFFO
며려쩌며略！
7C9107〕크
··…갼＞……갼J、
73alㅓeeo
eeeeeeee
e뱀‘234rp'7
TTTTTTTT
55555555
7C36
9흽굽9驪C14eeO
「《OR
l:x刀R
101긔IP
IJL
rㅓ0..j
.'1.
IJE
iC"P
.J"Z
.11뺑C
'THC
r손vㅎr룝erl.0040121D
e
rt냐er''".Oa4eZ173
4컨
r슬Verse".O04eZilF흼
탸拐
<JhP.얹KERHEL32.ReadFLIe>
E댜＜,ERX
하緞RTr.ver'eh.ee40i084
하勵RTr.ver'etl.004018F7
EE::(,EE::;
ES1.E드I
Cll.IOF〔:T극DE.:【ㅓea17크],!O
eO4OIOCI
ES0023E
CS001BE
550023E
DS0e23:
FS여囊3B드
GS0000『
La'tErr「
000釀尸02'
e8
F
80!OaOO
J「IP
C『IP
하삿冷Trever죌삵쨍．ee4.1.F7
qL,.'T「pTRD도rEF::<+4더2二m」
네,더
하抑Rlrr’닌’r$.h.eO4eleD3
RL.4-
하섯》RTrev'r5eh.004e10D0
ESI
EBZ
하緞RTrev슨r＄슨h.oe40i戚겠
ESI,；「
하艇FTrev.r$e".ee401eF7
rev·r$.".OO4e12es
00
00000000
noty
npty
not,
hpty
npt,
npty
npty
nptg
-UN(
o.e
e.e
0.e
e.0
0.0
l.el
!.이
Cㅇn(
Pre《
·－1…71 l乙
·U·’1·40 F一
P
LhBDBBBE:E:DLJpjlㅌ
JJDODDDDnUn나n-ㅡJ
El'Z a길
F a딩
C4CS八I츨닢P3Cgen딩eaoe凸000edU0ee
·、→ 5744E87Ee白긷oeoe00edU0e
·＞｀·＞갼갼
J너4UIU터느
.-tU401UE4
0040ll:IE6
,:l040소OEㅌ
00'4010EF
더더4더l山1
더더4더1111'
.1114너겡'ㅔ '
J너40긴U]EE
.:1040lDIED
0040소OIEF
O〔l401OLl（〕
t:104010DI
004010D3
0040!0D6
004010D8
e040leDD
004010DE
004010E2
004010E3
004010E4
이〕401OE弓
더04010E8
.:t더4011:t「
.:10401
0000
FST
FC닒
4020
027F
n뵙Oon·4-1→n-－너－－건
d킵npon·너－너4-1긔
_1.-1·n
0040!OER
<－·뻬
OB00
r놀『놀 了놀r놀
DS:【004021IB」＝47
RL:471'『〕톤｀
Junpfroh004010
('G-)
丁
Seethatweverifythebyte(at402llB)
This15theseCondbytewhich15again
너＂4令Zn륵肉I令白숍白功峯황白自－nn숍白功令더p「白白功功自白令令！r「눙F驪멩自宙白白
rrr브브쯔브r■_

See that we verify the byte (at 40211B) now.

This is the second byte which is again "G"

우리는 byte를 검증한다.( 40211B에서)

이것은 2번째 byte도 "G"를 뜻한다.

시스템 생성 대체 텍스트: ·V
·V
>
JHZ
J"P
OR
J,-14니
긔J40
더.1ㅓn
R
1p
令U
ㅓl
V ㅏ
…\굿 · ·
H■비나H트EF
훰훰…l--l-…―…
' *'·횹 *·욕
더040n:i[
.1'14너1 '…
JL
「1〔I(
1。1→L'
1 1」.
L l
ㅑ ㄷ
'〔，뜨F
:v
I010DO
―「H
돌
1llP
IJE
C"P
JHZ
Il뺑C
[HC
J"P
Loㅇkasecㅇndatth이ooP,
tㅇSeehOWitallWOFks
004010D3
0040!0D6
Oe4O10D8
oe401eDD
>
·V
·V
（구독구｀＇볍》
7502
ES43
슝．크DB
'i:,3FE.
：크즈：D7큐요섧ㅓeCO
7C3'
十슝n931祗스14e이〕
叉00
'40：〕
3C47
7501
48
49
-E「E프
83FE08
7CIF
Eg28elo鱗珊
00
· 뜨‘『틸r느'【닙『lr겨
하緞RTrever∥h.ee40i0B4
하勵RTt.ver룔eh.e040lOF7
EE::<.EBX
ES1.E·크1
DUCf;：【「T므D캬〔ㅓq드17광〕,10
SHDRTre냔etseh.ee40leF7
qL,E낙T「pTRD：죠rEg꺼＋4더21lp」
RL,‘·
하緞RTr.v.r'.n.ee401eO3
RL.4一
아紛RTreverseh.oe40ieDO
ESI
EB냥
하沁RTrever드en.O04elOCI
ESI,：「
SHORTrever맬해．ee40leF7
rtper두．H.Oa4e12es
〔1수l

Look a second at the loop, to see how it all works

두번째 loop를 봐라, 어떻게 일이 이루어질지 봐.

시스템 생성 대체 텍스트: O
1숨
밉■룐차』밉J맥눕r
〕04010E:j
004011〕E4
004010E6
004011:)E：광
ol:141:110EF
호얀柏」l:llDI
004011휘-
00401너l'
004이'XE
.:1040lllD
.v7502
.vEB43
> 승3DB
83FE·
：즈：광D7팝alㅓl:’醜’
.v7C36
> 8R83IRa14000
· '．·－·r、''.r쨍＇'
하緞RTrever｀내．ee40!eB4
하蹈RTr.ver훰’h.004010F7
EE:：切，EE.
큅
븟뇝
· R '7·
SBHSO
EESEH
.:10·41:110더〕
.71더401n「1
0040!0D3
0040!0D6
004010DS
004OieDD
OO4010DE
OO4010EZ
0040!0E3
004010E4
0040!OEg
004010E6
004010E尸
.:1l〕401I:IE.-
郵〕40二「
1.141--f
EE
FE08
lF
28el
J『12
J千
::r[
：《l:IR
이IP
JL
110U
,'.1r'
JE
〔·뎌p
JHZ
IHC
IHC
J'1F'
C『IP
JL
ESI.EEI
DUCF【FT팍D캬》홍eal7겻,10
하緞RTreVer술e".Oe4e10F7
RL,B부T「PT尸Dg:〔Ee:v:+4021!n】
내.너
하【Rlrr'v'rten.ee401eD3
qL,4' ／〃－---
SHORTreVerseh.ee4
And15abOUttㅇadd
thefㅇUrthㅇne
－…1-r'1
*U04公·
Trever룔슨h.ee4
8
V떽 ^
…-·일7 '
r.V'rt'h.
J予rever흡．"-ea4el
ee4ele
2e드
000000
卜1.'
흼1.1111'l'L'L',·
01<.Theprㅇgramhas
alreadyfㅇund31'G"
蝦·3B3Cgea깊doaOO쇼penonUO白UC
44E87Een깊d0000edU8don다ee
>갼」
흽■■■떼
rEr'=一=
n니DD--l
<
ES!=00000003

Ok. The program has already found 3 "G"

And is about to add the fourth one

Ok. Program은 이미 3개의 "G"를 찾았다.

그리고 4번째를 추가하려고 한다.

Now, how can the loop be left?

Not here!

이제, 이 loop를 어떻게 빠져나가야 하나?

여기는 아니다.

시스템 생성 대체 텍스트: DecHexASCH
De리HexIAscll
_DecHex.Ascll_-De。―Hex-Ascll
ab cd e…f…gh
颱巨陸陸隧陸匡R隧糖·6.A씬6C陌阿ha-－勿·ㄱ끄刀
颱除9s99100esl01陋陋陋105tr06ll0,耐陋阿阿Vll3114115
楓）…ABC。EF…GHI
阿恥
65同1
66 42
6' 43
68 44
69 45
'0 46
-1147
P-―·es、
CO→ ㅡ
門U esㅓ．ㅗ 겋·
魯' ㅡ令' 겋'
｀■‘ ㅡ가·
nUesCV
1ㅗㅡ珊-
--S…、…＆…、
rJ It ·익＊es<V · OO C夕
…하' 魯' …하' …魯· …魯- ㅡ令' 가'
3 03
4 04
F ()5
6 ()6
ㅋF
)
…魯-es한그eslt ·‘- 乙Ues·' esOOesC>esCVesㅓ■ㅗ
es勺그 es한그 esrg esrJ rg …rj es신J es쬔그 eslt esㅓt
kl m n o pqr s t u vesw
JKL…3.I…N…。pQR
阿…軫4.A쁘忙始匣
끄刀긱즈拓:阿
4r醜一J
43 ZB 一
' 8 9 A B
凸U CV CV CV CV
l …8…901
ㅡ｀■ㅗ 1■ㅗ
阿颱
巨
5 그D
하· es히」
ㅓ．ㅗ esㅓ■홑
예■ㅗ es갸-
7빽날 7꺼一
｀■‘ 갸-
00 00
16 10
1른 11
18 12
19 13
{陋
jr따
同巨r「F
嘴놉
Fr啄F
80 ro
C…L…E…F
nUesCVesCVesCV
ㅓ낙 ·｀I
1-Hl
150
83 드3 5
阿54 T
85 55 L-
86 56 V
飇57-w
…8858 x
89 59 Y
90 SA Z
阿l.SB [
9긔5C
…93.5D 』
巨T汪
95 SF
74
'
하솔 r-, It ·｀-
하' r今긍 esl톄 ·써·
r夕 히J es히J es한」
'·esr蠟
·익－ ·허一
阿巨히궂F
륵6
119屋7
쎄■l esf■홑 esf■ㅗ 쫴
쎄■l es쫴■홑 es체■ㅗ 쫴
…＄5
…56
同죠。
,8
陋1綽
-26
122첵A
1:3…，B
124첵C
DEL
D E F
· -l、 －수‘
es7f늴es<V I
…하I …하· …魯·
esㅓ■홑 es쎄■l ｀■솥
··…·…－·…·…··…
…3'
陋
…」9
3A
―」B
-3c
阿
阿
阿
· OO C걋 nUesㅓ■‘…하콥es히수
＇빽－ ＇빽날 ＇맥1 <Ues<Ves<Ves<V
SOH
STX
ETX
EOT
D'-Q
ACK
BEL
BS
HT
LF
\'T
FF
CR
SO
SI
DLE
DCI
DCZ
DC3
DC4
NAK
SYN
ETB
CAN
EM
SUB
ESC
FS
GS
RS
US
1.A陋llc…1D…1E
l26샌·…鞠聊…l30
31 IF

But here and only if the byte is "empty", this is at the end of the file

BTW, the counterpart in hex for the "number" zero (ascii) is not 0 but is 30.

만약에 이것이 file의 끝에서 Byte가 "empty"가 된다면 오직 여기다.

이 부분은 숫자 0이 아니다. 숫자 0은 ascii로 30 이다.

00 means empty !!!

00의 뜻은 empty(비어있다) !!!

48d == 30h == 0 (ascii)

10진수 16진수 ascii

*번역 주)즉, 한마디로 ascii에서 파일의 끝은 0으로 쓰임. 꼭 이해하고 넘어가야 됨!*

Be sure to understand this and look at the DEC2HEX2ASCII page for this. If you have no programming background, I'm sure this is a bit confusing at first.

물론, 이것을 이해했을 것이고 DEC2HEX2ASCII page를 봐. 만약에 programing 지식이 없다면, 처음에는 꽤 헷갈릴 것이다.

시스템 생성 대체 텍스트: JHZ
J千
KO팟
'．■－＇【】''.「＇'、
하緞RTreVers황rl.oe4010B4
하【RTt.v.r룔．h.004010F7
EE::<.EE
0040
0040
0〔140
0040
0040
0040
OO4e
0040
0040
gp」O
t:104〔）
.v7502
.vEB45
> ．무콰DB
83F6
：즈：광D
.v7C36
》 8R83
가■■44·
…·盼畛
78alㅓeee10
XOR
］걋】P
in쑨14000
ES1.E．드1
미力DF［二FT덕DS:「ㅓq드17프죠,10
하섯）RTreVer룔eh.0048leF7
RL,B부T「PTR一D요L【E盼披4aZI
니1.1빻
하【RT
니L.4
－·1sr'＊요
*U너4白힙
1－叩
10D
EE
Fㅌ08
lF
28810毓拗
ESI
E하＜
하刃RT
ESI.：》
Howmany"G"5the
prㅇgramhasfound구p
洽죠淅祇輛『－-－하一
'·＞견
'4b…c스Fl一찬·q·BDF一邑므1
橓權一櫛硼1eC溫
匯14t:l
0040
a킵
np
白p
dU
＂·吸。00。。。。。。。
니仕U凸UO白p00色Ud빕仕긷ee
S냈nTr.V.r'.h.ee4eleF7
rever＄●H.OO4eiZes
JLt.1Oul-'.1:'JE潞磻Jl*r幷DDDBDBDBDBDBDBDBDB맸
- ―낙-
000000
J74앗백·褐盼ESBa7CEgee매卵애開卵卵卵卵80盼一
·‘Iesl졈I■l룅
>갼什
치欄測撤釗I
·■‘｀ㅗ뀀‘뱁홑·홑.'．욥.．솝．홑·‘·‘．흡·홑·
·UapnUaU-UaU白킵npn킵dp0CH
44444444444계1
:i：騁騁開盼朋嵋盼騁卵卵一·＜
ES!=O000ee0R
JUnDfroh0040i0C9

I hope you have understood all so far.

지금까지 네가 이해 했기를 희망한다.

So, let's run the ReverseMe till …

… here.

여기까지 ReverseMe를 실행하라.

… 여기.

시스템 생성 대체 텍스트: 〔1〔1냐LllUHL
UbLU
75eZ
E843
요－코〕E:
[3F6
；드크크D
7C36
8n83
〕〕0.-1
74너8
3〔47
닝뺑01
ILbl
JHZ
J"P
::rl
LHX'LHK
하緞RTrever’애．ee40leB4
하勵RTr.ver'eh.OO4018F7
[E::<,EE'
ES1.EEI
D니I:FㄷFT므D하드ㅓq크1,크〕.珏）
04010EF
7크alㅓe枕ㅣ1,:ll:뎌P
IJL
l요a」츠凹 .I.10U
I'1:'
:La크·匕Cl
瀧耐慨
0쵸브n-nl
44eel뺙4
너。范Uee-U白l
U죠U百Ial
SHDRTre니er룔eh.OO48leF7
qL.BVTEPTRDS:[EBX+4021Inl
톤41.''
r.．■··．1·－．우
>·겯·r·＞견＞
히｀gBDF一no
皿蹴10c10clOC邸10D厠
卄40牝盼牝塑犯貂
鉀뾰에에盼嵋嵋
0040!0D6
004Oi0D8
e04010DD
OO4OIODE
ㅓ6
43
ESEE
saFE08
7C!F
Eg28010陝泌
90
JE하竇PTr’냔．rten.e0401eD3
I걋lp니니4
JHZ하緞RTrevers슬h.e048i000
1니CESI
IHCE하《
J∥P騈切RTrever＄해．0040!OCI
ClfESI.;3
JLSHDRTr．니er＄·n.oe40leF7
JhPre닌＂$’ㅒ．OO4e12e5
OB00
DD00000000
一__.- ._.. . .一녹po
OAh==1ndInd尸尸d_ ,p
''''' -'一'''一一一一' 7p0
Ff一!-
ESI=OO0ee80R
JUn。Frone040!8C9

How many "G"s the program has found ???

0Ah == 10d indeed.

얼마나 많은 "G"를 찾았어?

0Ah == 10d 다.

Those BP's won't be needed any longer :)

이제 더이상 BP가 필요하지 않다.

No jumping to the badboy:

We jump to the goodboy :))

Badboy로 jump 하지 않는다.

우리는 goodboy로 jump 한다.

시스템 생성 대체 텍스트: .CP【j-颱柚thread,飇dulereverseㅆ
－찧7一一－
－쑈’츤－섭슬
HHHLLㅒI1
PUSPUSPUS呱CRLRET
00
aOZ040OO
DE204이）O
00
4「010000
gC000釀蹈
e
r슝v'r룔．".004e2000
r슬Verse".eO4020DE
'1
<Jr1P.各USER32.he'$ageBoㅐR>
<J찧P.各KEf꺼EL32.EㅐitPr0C손S톨》
ㅣ
TLt【．
TeHt
■볜（潔I벼LgPPU竇】며L
="KeyFL【eReversehe"
h哄勵er=
VOutea나
"ULL 」Z
dLdit'CㅇngratZ슝
뎌e＄흡ageB。ㅐR
.：엥！P二Z
HereWeCOme !
胡6868胡ESESC30e卵애的觸卵

Here we come !!!

Let's finally run the ReverseMe to see how things should be …  
마지막으로 어떤 일이 발생하는지 보기 위해 실행하자.

시스템 생성 대체 텍스트: 一’쑈트즈졉좁＝ 騙＝，콰큭＝H
回團
“넙inthrea비"x》d미ereverse糾
n’턺r珊『－----＝■m驪「戚－
'
TTh
;-―「―「
《―」］
PUSH
PUSH
PUSH
CnLL
CRㄴL
RETH
OB00
DB00
DB00
DB00
DBee
DB00
DB00
DB00
DB00
DB00
DB00
DB00
OB00
DB00
DB00
DB00
DB00
DB00
DB00
DB00
DB00
DB00
DBeo
DB00
DBeo
DB00
DB00
DB00
DBoe
DB00
DB00
DB00
DBoe
DBee
「눙『b r놈了》
r뺨ver$e".004e2e08
r슬Versell.OO4eZODE
콰棧驪’纖
VㅇureaLt,
Revet룔슨＂e"
dLd댜?以mgratZ?
U
<J뿌．各USER32."e$$ageBoㅐR>
《JP.各KEf꺼EL32.EHttPrOC슨＄S》
it【e
e심t
O닐ne
컨
NLJLL
Lne$$ageB。xR
EHLtPr。ce'$
JunOfro"e0401OD8

;)

1. **Conclusion**

Congratulations !!!

You just keyfiled your first ReverseMe.

축하해.

너는 ReverseMe를 막 끝냈다.

I hope you understood everything fine and I also hope someone somewhere learned something from this.

나는 네가 모두 이해했기를 그리고 누구든지 어디에서 이것에서 무엇을 배웠으면 하고 희망한다.

See me back in part 3 in this series ;)

나는 이 series의 Part 3 에서 돌아오겠다.

The other parts are available at

다른 parts는 사용 가능하다.

<http://tinyurl.com/27dzdn> (tuts4you)

<http://tinyurl.com/r89zq> (SnD Filez)

<http://tinyurl.com/l6srv> (fixdown)

Regards to all and especially to you for taking the time to look at this tutorial.

Lena151 (2006, updated 2007)

모두에게 안부를 전하고 특별히 이 tutorial에 시간을 투자해준 너에게 감사한다.