PWNABLE.KR - flag

Download the Binary

It is said that this is a simple reversing task

Lets list the files,

```
ra@moni~/P/p/flag> ls -la
total 340
drwxrwxr-x 2 ra ra   4096 Jun  3 09:37 ./
drwxrwxr-x 6 ra ra   4096 Jun  3 09:36 ../
-rw-rw-r-- 1 ra ra 335288 May 16  2019 flag
-rw-rw-r-- 1 ra ra    120 Jun  3 09:38 flag.md
```

Now lets check the file type using `file` command,

```
ra@moni~/P/p/flag> file flag
flag: ELF 64-bit LSB executable, x86-64, version 1 (GNU/Linux), statically
linked, no section header
```

So it is a `stripped` binary

By using `strings` we can see whether there is any ASCII or Unicode strings in it or not,

But it is a lot for this binary

Lets try running this binary,

```
ra@moni~/P/p/flag> ./flag
fish: The file “./flag” is not executable by this user
ra@moni~/P/p/flag> chmod +x flag
ra@moni~/P/p/flag> ./flag
I will malloc() and strcpy the flag there. take it.
```

Its time for debugger,

```
ra@moni~/P/p/flag> gdb ./flag
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later
<http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```

```
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
pwndbg: loaded 194 commands. Type pwndbg [filter] for a list.
pwndbg: created $rebase, $ida gdb functions (can be used with print/break)
Reading symbols from ./flag...
(No debugging symbols found in ./flag)
pwndbg> info functions
All defined functions:
pwndbg> disassemble main
No symbol table is loaded.  Use the "file" command.
```

Since it is a `stripped` binary we cannot see any symbols in it

So i did more digging with `strings`

```
ra@moni~/P/p/flag> strings flag | grep flag
mthNflag
ra@moni~/P/p/flag> strings -10 flag
'''' (0h''''HPX`
np!f@(Q[uIB(0Tc
Is\AQ9@&9;0
|>_ g(o0|y
*94D@yq 9j
DEo0?^ T3   xF5
D+Hf]X)J(~ `
3n<_t6<-t2<.t.
^'oxMlqM|M
%!,L98v(>F
 ^=9X~"1[p
wwu62#}w"I
!</x    dSt62A
DUt]c[y;"Z
ps0/(HTDsxb
d"; Ut;IZ,
FFFF|vpjFFFFd^XR
9.Q_q_: !m
B like a de
vic$:)oI wi{3{
oc()VjHtrcpV
_STDERR_U4
^0HMdZp)->? & 0+03
`*r(])iP!ph
?0v[2*i+3]U
m0"|| / (GDU
]15QQen",p
?../:deps/x86_64
ck_worYd$6
6@?GCONVFTH
F_DIRlOSpIASE
?_OUTPU1YNAMIC_WEAK
_~SO/IEC 14652 i18n FDC
 !"#$%&'()
*+,-./0>3x6789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
C`.usr/sJe/
-c+21474836
la.HfB[-1]
ANSI_X3.4&968//T
;][tpVt8isT
[{"=q"q ,(0t\Q
p_9ILP32_OFF
: O_CREATE
'k$~[9]->d
D 0no*RRx_
#J?3O2ABI
NCYL 5GSCO0
-10 6/UCS4/H
,$PZT(ALJ7
-4LE/SleWle[u%
LEPHONiASU2
 "9999$%&/999956799999:<DG9999HI_`
#6''''<dej''''k
 ''''!#$`''''abcd''''efgh''''ijkl''''mnop''''qrst''''uvwx''''yz{|''''}~
't*uv#oWCa
Q2R''''STUV''''WXYZ''''[\]^''''_
MNONNNNPRTUNNNNVWYZNNNN[\_`NNNNabcdNNNNefhi
 rrrr!"#$rrrr%&'(rrrr)*+,rrrr-./0rrrr1234rrrr5678rrrr9;
<=rrrr>@ABrrrrCDFJrrrrKLMNrrrrOPRSrrrrTUVWrrrrXYZ[rrrr\]^_rrrr`abcrrrrdefgr
rrrhijkrrrrlmnorrrrpqrsrrrrtuvwrrrrxyz{rrrr|}~
 !"9999#$%&9999'()*9999+,-.9999/0129999345699999789:9999;<=>9999?
@AB9999CDEF9999GHIJ9999KLMN9999OPQR9999STUV9999WXYZ9999[\]^9999_`ab9999cdef
9999ghij9999klmn9999opqr9999stuv9999wxyz9999{|}~9999
'12Wr%W345%Wr%67x!Wr892
b'cdr%WrefgWr%Whij%Wr%klr%WrmnoWr%Wpqr%Wr%str%WruvwWr%Wxyz%Wr%ABr%WrCDEWr%W
FGH%Wr%IJr%WrKLMWr%WNOP%Wr%QRr%WrSTUWr%WVWX%Wr%YZ
_r%W;k'MGEp%WTu
Fri.at7day=
pchuilqesyuustuw
 $9999(/6>9999HQXa9999eimq9999uy}
Vng1XENIX#
&9223372036854775807L`
<http://w(
PROT_EXEC|PROT_WRITE failed.
$Info: This file is packed with the UPX executable packer http://upx.sf.net
$
$Id: UPX 3.08 Copyright (C) 1996-2011 the UPX Team. All Rights Reserved. $
/proc/self/exe
```

```
GCC: (Ubuntu/Linaro 4.6.3-1u)#
ild-id$rela.plt
__libc_f8e
g?ojxS*/?8/
OT/?d''''[
/p#o]/P8vQj
call_gmon_start
DEH_FRAME_BEGINf
@pleted.6036
curi,cntrSd
|85408.mLm
afKjump_`B`
_PRETTY_FUNCT0Na
@#1{59#3a%[
.<e;5wdP9H*
~|2adDdoJs;0
C_>YPE/NUMERIC?
<tbltoCto?
mkpl'GLOBAL_O,
I^Mx868uA8"k
F    sa    WUH
```

And here is some interesting part from the `strings -10 flag` output,

```
$Info: This file is packed with the UPX executable packer http://upx.sf.net
$
$Id: UPX 3.08 Copyright (C) 1996-2011 the UPX Team. All Rights Reserved. $
```

It gives us some info that this `stripped` binary is being `packed` by `UPX`

So we should use `UPX` to unpack it

More on UPX,

https://en.wikipedia.org/wiki/UPX

https://upx.github.io/

We know that this binary is being packed by `UPX`,

Now lets install `UPX`

```
ra@moni~/P/p/flag> sudo apt install upx
[sudo] password for ra:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'upx-ucl' instead of 'upx'
The following additional packages will be installed:
  libucl1
The following NEW packages will be installed:
  libucl1 upx-ucl
0 upgraded, 2 newly installed, 0 to remove and 46 not upgraded.
Need to get 417 kB of archives.
After this operation, 2,158 kB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 libucl1
amd64 1.03+repack-5 [25.0 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 upx-ucl
amd64 3.95-2build1 [392 kB]
Fetched 417 kB in 1s (288 kB/s)
Selecting previously unselected package libucl1:amd64.
(Reading database ... 339326 files and directories currently installed.)
Preparing to unpack .../libucl1_1.03+repack-5_amd64.deb ...
Unpacking libucl1:amd64 (1.03+repack-5) ...
Selecting previously unselected package upx-ucl.
Preparing to unpack .../upx-ucl_3.95-2build1_amd64.deb ...
Unpacking upx-ucl (3.95-2build1) ...
Setting up libucl1:amd64 (1.03+repack-5) ...
Setting up upx-ucl (3.95-2build1) ...
update-alternatives: error: no alternatives for upx
update-alternatives: using /usr/bin/upx-ucl to provide /usr/bin/upx (upx)
in auto mode
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
```

Its time to unpack this binary,

```
ra@moni~/P/p/flag> which upx
/usr/bin/upx
ra@moni~/P/p/flag> upx -d flag
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2018
UPX 3.95        Markus Oberhumer, Laszlo Molnar & John Reiser   Aug 26th
2018

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    883745 <-    335288   37.94%   linux/amd64   flag

Unpacked 1 file.
```

Now lets check the file type using `file` command,

```
ra@moni~/P/p/flag> file flag
flag: ELF 64-bit LSB executable, x86-64, version 1 (GNU/Linux), statically
linked, for GNU/Linux 2.6.24,
BuildID[sha1]=96ec4cc272aeb383bd9ed26c0d4ac0eb5db41b16, not stripped
```

Yeah! Our `packed stripped` binary became `not stripped` binary

Now we can easily analyze it with our debuggers

Opening our decompressed binary in debugger,

Disassembling `main()`

```
pwndbg> disassemble main
Dump of assembler code for function main:
   0x0000000000401164 <+0>: push   rbp
   0x0000000000401165 <+1>: mov    rbp,rsp
   0x0000000000401168 <+4>: sub    rsp,0x10
   0x000000000040116c <+8>: mov    edi,0x496658
   0x0000000000401171 <+13>:    call   0x402080 <puts>
   0x0000000000401176 <+18>:    mov    edi,0x64
   0x000000000040117b <+23>:    call   0x4099d0 <malloc>
   0x0000000000401180 <+28>:    mov    QWORD PTR [rbp-0x8],rax
   0x0000000000401184 <+32>:    mov    rdx,QWORD PTR [rip+0x2c0ee5]
# 0x6c2070 <flag>
   0x000000000040118b <+39>:    mov    rax,QWORD PTR [rbp-0x8]
   0x000000000040118f <+43>:    mov    rsi,rdx
   0x0000000000401192 <+46>:    mov    rdi,rax
   0x0000000000401195 <+49>:    call   0x400320
   0x000000000040119a <+54>:    mov    eax,0x0
   0x000000000040119f <+59>:    leave
   0x00000000004011a0 <+60>:    ret
End of assembler dump.
```

It looks like we got our `flag` from the comment which is being loaded into `rdx`,

```
pwndbg> x/s 0x6c2070
0x6c2070 <flag>:    "(fI"
pwndbg> x/s *0x6c2070
0x496628:   "UPX...? sounds like a delivery service :)"
```

Done! we got our flag,

`Flag: UPX...? sounds like a delivery service :)`