

Nejmenší x86 ELF Hello World

(Že bych mohl dosáhnout)

Konečná velikost: **142 bajtů**

Intro

Tato stránka je kombinace tutorial / dokument o mých pokusů o vytvoření co nejmenší x86 ELF binárky, které by spustit říkat Hello World na Ubuntu Linux. Moje první pokusy začali s C a poté postupoval k montáži x86 a nakonec k HexEditor. Skončil jsem kompromisů a přechod na "Hi World" app místo toho, aby se vešly data řetězec do elf magické číslo. Konečným výsledkem je zcela poškozen x86 ELF Binary, která stále běží.

Od začátku až do konce.

- První věc, kterou musíte udělat, je dostat nastavení je správné prostředí.
 - Nainstalovat Ubuntu (nebo distro dle vašeho výběru)
 - spustíte: **sudo apt-get install g ++ gcc nasm**
 - Systémové verze

```
uživatel @ počítač: ~ $ lsb_release -a
Žádné LSB moduly jsou k dispozici.
Distributor ID: Ubuntu
Popis: Ubuntu 8.04.1
Release: 8.04
Kódové označení: vytrvalý
uživatel @ počítač: ~ $ uname -a
Linux ryanh-desktp 2.6.24-19-generic # 1 SMP st 18.června 14:43:41 UTC 2008 i686 GNU / Linux
uživatel @ počítač: ~ $ gcc version
gcc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu7)
Copyright (C) 2007 Free Software Foundation, Inc.
Toto je svobodný software; viz zdroj pro kopírování podmínek. Tady není žádný
záruka; Ani co se týká obchodovatelnosti pro určitý účel.
uživatel @ počítač: ~ $ nasm -version
NASM verze 0.99.06-20071101 sestaven na 15 Lis 2007
```

- Moje první pokusy začal s C se vkládá je to, co jsem použil pro chello.c

Kód: **chello.c**

```
#include <stdio.h>
int main () {
    printf ( "Hi World \ n " );
    vrátí 0 ; }
```

- Příkaz: gcc

```
uživatel @ počítač: ~ $ gcc -o chello chello.c
uživatel @ počítač: ~ $ ./chello
Hi World
```

- Můj původní spustitelný bylo **6363 bytů** .
- Můžete použít readelf vypsát hlavičky ELF ze spustitelného souboru.

- Příkaz: readelf

```
uživatel @ počítač: ~ $ readelf -h chello
ELF Header:
  Magic: 7f 45 4c 01 01 01 46 00 00 00 00 00 00 00 00
  Třída: ELF32
  Data: 2 doplňkem, little endian
  Verze: 1 (proud)
  OS / ABI: UNIX - System V
  ABI Version: 0
  Typ: EXEC (spustitelný soubor)
  Stroj: Intel 80386
  Verze: 0x1
  Adresa vstupní bod: 0x80482f0
  Začátek programu hlaviček: 52 (bajtů do souboru)
  Začátek sekce záhlaví: 3220 (bajty do souboru)
  Příznaky: 0x0
  Velikost tohoto hlavičky: 52 (bytes)
  Velikost programu záhlaví: 32 (bajty)
  Počet programových záhlaví: 7
  Velikost sekce záhlaví: 40 (bytes)
  Počet sekcí záhlaví: 36
  Sekce záhlaví index řetězec tabulka: 33
```

- ldd je užitečný pro zobrazující všechny dynamické knihovny spustitelný je propojen.

- Příkaz: ldd

```
uživatel @ počítač: ~ $ LDD chello
linux-gate.so.1 => (0xb7f77000)
libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0xb7e18000)
/lib/ld-linux.so.2 (0xb7f78000)
```

- Soubor vám dá popis toho, co je soubor.

- Příkaz: file

```
uživatel @ počítač: ~ $ file chello
chello: ELF 32-bit LSB spustitelný, Intel 80386, verze 1 (SYSV), pro GNU / Linux 2.6.8, dynamicky propojeny (používá sdílené libs)
```

- "Not zbažený" vrátil z příkazu souboru znamená, že ladění symboly nebyly odstraněny z executable.

Příkaz: strip

```
uživatel @ počítač: ~ $ strip -s chello
```

- Po odizolování spustitelný byl nyní **2984 bajtů**, stále nepřijatelné! Čas přijmout drastická opatření ...
- Poškrábal jsem pokus C a zrušena pomocí printf, místo toho zvolit pro montáž nasm x86.

file: hello.asm

```
ČÁST .data
zpráva: db "Hi World", 10
len: equ $ -msg
```

ČÁST .text

Globální hlavní

main:

```
mov edx, len
mov ecx, MSG
mov ebx, 1
mov eax, 4
```

```
int 0x80
mov ebx, 0
mov eax, 1
int 0x80
```

- Kompilace ASM

```
uživatel @ počítač: ~ $ nasm -f elf hello.asm
uživatel @ počítač: ~ $ gcc -o Dobrý den hello.o -nostartfiles -nostdlib -nodefaultlibs
uživatel @ počítač: ~ $ strip -s ahoj
uživatel @ počítač: ~ $ ./ Ahoj
Hi World
```

- Před odstraňováním souboru bylo **770 bytů** po stripping **448 bajtů**. Nicméně stále je zbytečné hlavičky a oddíly zničit.
- Otevřete binární ve svém oblíbeném hex editor, já používám kletby HexEditor a ghex2.

```
File: hello          ASCII Offset: 0x000000AD / 0x0000018F (%39)
00000000  7F 45 4C 46 01 01 01 00 00 00 00 00 00 00 00 00  .ELF.....
00000010  02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 00  .....4....
00000020  F8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 00  .....4. ...(.
00000030  05 00 04 00 01 00 00 00 00 00 00 00 00 80 04 08  .....
00000040  00 80 04 08 A2 00 00 00 A2 00 00 00 05 00 00 00  .....
00000050  00 10 00 00 01 00 00 00 A4 00 00 00 A4 90 04 08  .....
00000060  A4 90 04 08 09 00 00 00 09 00 00 00 06 00 00 00  .....
00000070  00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
00000080  BA 09 00 00 00 B9 A4 90 04 08 BB 01 00 00 00 B8  .....
00000090  04 00 00 00 CD 80 BB 00 00 00 00 B8 01 00 00 00  .....
000000A0  CD 80 00 00 48 69 20 57 6F 72 6C 64 0A 00 54 68  ....Hi World..Th
000000B0  65 20 4E 65 74 77 69 64 65 20 41 73 73 65 6D 62  e Netwide Assemb
000000C0  6C 65 72 20 30 2E 39 39 2E 30 36 2D 32 30 30 37  ler 0.99.06-2007
000000D0  31 31 30 31 00 00 2E 73 68 73 74 72 74 61 62 00  1101...shstrtab.
000000E0  2E 74 65 78 74 00 2E 64 61 74 61 00 2E 63 6F 6D  .text..data..com
000000F0  6D 65 6E 74 00 00 00 00 00 00 00 00 00 00 00 00  ment.....
00000100  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
00000110  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
00000120  0B 00 00 00 01 00 00 00 06 00 00 00 80 80 04 08  .....
00000130  80 00 00 00 22 00 00 00 00 00 00 00 00 00 00 00  ....".....
00000140  10 00 00 00 00 00 00 00 11 00 00 00 01 00 00 00  .....
00000150  03 00 00 00 A4 90 04 08 A4 00 00 00 09 00 00 00  .....
^G Help ^C Exit (No Save) ^T goTo Offset ^X Exit and Save ^W Search
```

- Smazat vše, včetně a kolem offset 0xAD, bude to hodit ho dolů na **173 bajtů**

```
File: hello          ASCII Offset: 0x00000000 / 0x000000AC (%00)
00000000  7F 45 4C 46 01 01 01 00 00 00 00 00 00 00 00 00  .ELF.....
00000010  02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 00  .....4....
00000020  F8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 00  .....4. ...(.
00000030  05 00 04 00 01 00 00 00 00 00 00 00 00 80 04 08  .....
00000040  00 80 04 08 A2 00 00 00 A2 00 00 00 05 00 00 00  .....
00000050  00 10 00 00 01 00 00 00 A4 00 00 00 A4 90 04 08  .....
00000060  A4 90 04 08 09 00 00 00 09 00 00 00 06 00 00 00  .....
00000070  00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
00000080  BA 09 00 00 00 B9 A4 90 04 08 BB 01 00 00 00 B8  .....
00000090  04 00 00 00 CD 80 BB 00 00 00 00 B8 01 00 00 00  .....
000000A0  CD 80 00 00 48 69 20 57 6F 72 6C 64 0A          ....Hi World.
```

```
^G Help ^C Exit (No Save) ^T goTo Offset ^X Exit and Save ^W Search
```

hello.asm

SECTION .data		
msg:	db "Hi World",10	
len:	equ \$-msg	
SECTION .text		
global main		
main:		
mov, edx, len		00000000 7F 45 4C 46 01 01 01 00 00 00 00 00 00 00 00 .ELF.....
mov, ecx, msg		00000010 02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 004...
mov, ebx, 1		00000020 F8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 004. ...(.)
mov, eax, 4		00000030 05 00 04 00 01 00 00 00 00 00 00 00 00 80 04 084. ...(.)
int, 0x80		00000040 00 80 04 08 A2 00 00 00 A2 00 00 00 05 00 00 004. ...(.)
mov, ebx, 0		00000050 00 10 00 00 01 00 00 00 A4 00 00 00 A4 90 04 084. ...(.)
mov, eax, 1		00000060 A4 90 04 08 09 00 00 00 09 00 00 00 06 00 00 004. ...(.)
int, 0x80		00000070 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 004. ...(.)
		00000080 BA 09 00 00 00 B9 A4 90 04 08 BB 01 00 00 00 B84. ...(.)
		00000090 04 00 00 00 CD 80 BB 00 00 00 00 B8 01 00 00 004. ...(.)
		000000A0 CD 80 00 00 48 69 20 57 6F 72 6C 64 0A4. ...(.)

-
- Přesunout 0xA4-0xAC na 0x7 a změna offset 0x86 od 0xA4 do svého nového umístění 0x07. Odstranit 0xA2 a 0xA3

File: hello		ASCII Offset: 0x00000000 / 0x000000A3 (%00)	
00000000	7F 45 4C 46 01 01 01 48 69 20 57 6F 72 6C 64 0A	.ELF...Hi World.	
00000010	02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 004...	
00000020	F8 00 00 00 00 00 00 00 34 00 20 00 02 00 28 004. ...(.)	
00000030	05 00 04 00 01 00 00 00 00 00 00 00 00 80 04 084. ...(.)	
00000040	00 80 04 08 A2 00 00 00 A2 00 00 00 05 00 00 004. ...(.)	
00000050	00 10 00 00 01 00 00 00 A4 00 00 00 A4 90 04 084. ...(.)	
00000060	A4 90 04 08 09 00 00 00 09 00 00 00 06 00 00 004. ...(.)	
00000070	00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 004. ...(.)	
00000080	BA 09 00 00 00 B9 A4 90 04 08 BB 01 00 00 00 B84. ...(.)	
00000090	04 00 00 00 CD 80 BB 00 00 00 00 B8 01 00 00 004. ...(.)	
000000A0	CD 80 00 00	

- ^G Help ^C Exit (No Save) ^T GoTo Offset ^X Exit and Save ^W Search
- Soubor by měl být **164 bajtů** a nyní je čas pro vstup do šedé zóny ... Zbytek je hodně vysvětlit, v podstatě jsem se pokusil najít to, co jsem mohl změnit v elfi hlavy ven s to segfault na me.I doplněny jmps a zcela poškozen spustitelný soubor, nicméně to stále běží :). Zde je několik užitečných informací: V x86 0xD9D0 je nop nebo žádný provoz, použitelné jen za vyplnění prostoru, pokud potřebujete. 0xEB následovaný jediným byte je relativní JMP. Opravdu byste měli přečíst [intel dokumenty](#) týkající se instrukcí x86 [AM a NZ](#).
- typedef struct {
 unsigned char e_ident [EI_NIDENT];
 Elf32_Half e_type;
 Elf32_Half e_machine;
 Elf32_Word e_version;
 Elf32_Addr e_entry;
 Elf32_Off e_phoff;
 Elf32_Off e_shoff;
 Elf32_Word e_flags;
 Elf32_Half e_ehsize;
 Elf32_Half e_phentsize;
 Elf32_Half e_phnum;
 Elf32_Half e_shentsize;
 Elf32_Half e_shnum;
 Elf32_Half e_shtrndx;
 } Elf32_Ehdr;

```
File: hello          ASCII Offset: 0x00000000 / 0x00000080 (%00)
00000000  7F 45 4C 46 01 01 01 48 69 20 57 6F 72 6C 64 0A .ELF...Hi World.
00000010  02 00 03 00 01 00 00 00 80 80 04 08 34 00 00 00 .....4...
00000020  00 B8 04 00 00 00 CD 80 EB 58 20 00 02 00 28 00 .....X ...(.
00000030  05 00 04 00 01 00 00 00 00 00 00 00 00 80 04 08 .....
00000040  00 80 04 08 A2 00 00 00 A2 00 00 00 05 00 00 00 .....
00000050  00 10 00 00 01 00 00 00 A4 00 00 00 A4 90 04 08 .....
00000060  A4 90 04 08 09 00 00 00 09 00 00 00 BA 09 00 00 .....
00000070  00 B9 07 90 04 08 BB 01 00 00 00 EB A4 00 00 00 .....
00000080  EB EA 3B 00 00 00 00 B8 01 00 00 00 CD 80 .....
. ^G Help ^C Exit (No Save) ^T goTo Offset ^X Exit and Save ^W Search
```

Závěr.

Konečná velikost: **142 bajtů**

[helloworld.tar.gz](#)

Jsem si jist, že existují způsoby, jak získat ještě menší. Mohou existovat také další věci, které mohou být odstraněny z hlavičky ke zvýšení velikosti, ale nechtěl jsem strávit dostatek času plně zkoumat ELF formátu záhlaví. Další možností by mohlo být použití formátu a.out verzi namísto ELF mohou vám umožní získat ještě menší.

Komentáře, návrhy a kritická kritika přijala: henszey@gmail.com

[Home](#)