Instructies

Geef een duidelijk antwoord op alle vragen die hieronder worden gesteld.
Geef een toelichting als dat wordt gevraagd !
Maak van je antwoorden een verslag in **pdf**-vorm en lever dat in op de **DLO**.

Opgave 1a

We gebruiken in deze opgave een simpel c programma.

```
#include <stdio.h>

char * msg ="hallo wereld!";

void toon(char * tekst){
  printf("%s\n", tekst);
}

int main(){
  toon(msg);
  return 1;
}
```

je vindt dit programma in bestand : `opg1a.c`

Dit programma is een variatie op **`helloworld.c`** uit *SMP_C_Gereedschap_1.pdf* .
Hier is een (globale) variabele **`msg`** gedefinieerd met de tekst die getoond wordt.
Er is een functie gedefinieerd : **`toon()`** die **`msg`** als parameter mee krijgt.

Actie 1

Compileer dit programma en geef de executable de naam : **`opg1a`**.
Je kunt dit in 1 commando doen !

Vraag 1 : geef het gebruikte commando.

Actie 2

Run het gemaakte programma **`opg1a`** en bepaal de return waarde van dit programma.

Vraag 2 : geef beide gebruikte commando's

Actie 3

Onderzoek met het **`file`** commando het type bestand van `opg1a.c` en van `opg1a` .

Vraag 3 : geef 2x het file type

Actie 4

Controleer / onderzoek of de volgende bewering juist is :

"het file commando gebruikt de extensie om te bepalen of opg1a.c een c-source code bestand is. "

Vraag 4 : hoe heb je dit onderzocht en wat is je conclusie?

Actie 5

Analyseer het programma `opg1a` m.b.v. het `strings` commando.

Vraag 5 : zijn **de namen** van de nieuwe de definities, glob. variabele `msg` en functie `toon`, terug te vinden in de strings output ? Geef een duidelijke verklaring waarom dit zo is !

Actie 6

Voer het volgende commando uit :

```
strip -s -o opg1ab opg1a
```

dit commando verwijdert alle symbols van het programma `opg1a` en plaats het resultaat in output bestand `opg1ab`. Zie : *`man strip`* voor alle details.

Vraag 6 : wat geeft het `file opg1ab` commando als output ?
Vraag 6b : dezelfde vraag als bij actie 5 nu voor opg1ab.

Voer nu het volgende commando uit :

```
strip -s -K main -o opg1ac opg1a
```

Dit verwijdert alle symbols maar behoud het symbol 'main'. [-K voor Keep]

Vraag 6c : wat geeft het `file opg1ac` commando als output ?
Vraag 6d : kun je concluderen dat bij file output = ' …. not stripped' ook alle symbols aanwezig zijn ?

Actie 7

in de strings output zien we de tekst : `puts`

Vraag 7 : wat is '`puts`' (een variabele, functie, of ??) en waar is het gedefinieerd ?
Vraag 7b : dezelfde vraag voor `puts@@GLIBC_2.2.5` , en waarom zie je dit alleen bij `opg1a` ?
[opmerking : de cijfers aan het eind, 2.2.5, kunnen iets anders zijn in jouw omgeving.]

Actie 8

onderzoek beide programma's opg1a en opg1ab met readelf -h . (dit geeft de header informatie.)

Vraag 8 : is er een verschil in aantal program headers, aantal section headers en het entry point tussen het originele programma en de ge'strip'de versie ?

Actie 9

kopieer het programma `opg1a` naar `opg1ad`. We gaan nu `opg1ad` bewerken in een hex-editor.

Wijzig de byte op offset positie 5 (vanaf het begin v/h bestand) van **0x01** naar **0x02**.

Vraag 9 : Geef de output van `readelf -h opg1ad` na wijziging.
Is hier iets gewijzigd ? Verklaar eventuele verschillen.
Kan het programma nog uitgevoerd worden ?