

Groupe 10  
HOUNY Julien  
DE VIALA ROSALES Gerard  
ERGIN Seçkin Yağmur  
LOMBARD Antonin  
LORGNIER Théo

## Descriptifs de fonctions:

### elf\_header:

#### load\_elf\_header:

Il s'agit d'une simple fonction de lecture d'une partie du fichier donné en entrée, nous sommes positionnés au début de ce dernier car c'est ici que commence le HEADER du fichier ELF, copiant ces données dans une structure appropriée pour nous permettre de les lire.

#### check\_elf\_header:

Cette fonction vérifie les différents éléments contenue dans le header qui pourrait révéler une erreur dans la création du fichier ELF, à travers les fonctions suivantes.

#### check\_magic\_number:

Cette fonction vérifie que les valeurs de `e_ident[MAG0,1,2,3]` ont bien les valeurs par défaut car n'importe quel autre cas signifie une erreur dans la génération.

#### check\_object\_validity:

Cette fonction vérifie que l'adresse `e_ident[EI_CLASS]` ne contient pas la constante d'erreur `ELFCLASSNONE` sinon retourne une erreur.

#### check\_data\_validity:

Cette fonction vérifie que l'adresse `e_ident[EI_DATA]` ne contient pas la constante d'erreur `ELFDATANONE` sinon retourne une erreur.

#### check\_version\_validity:

Cette fonction vérifie que l'adresse `e_ident[EI_VERSION]` ne contient pas la constante d'erreur `EV_CURRENT` sinon retourne une erreur.

display\_elf\_header:

Enfin cette fonction print tous les éléments du header ELF.

sections\_header:

load\_sections\_header:

Une autre lecture du fichier ELF, cette fois-ci nous commençons la lecture à l'adresse se trouvant dans le header ELF, stocké dans `e_shoff` pour pouvoir lire le header des sections et le stocker dans une structure pour pouvoir vérifier et lire les données.

display\_sections\_header:

Affiche les header de toutes les sections du fichier ELF.

ExtractName:

Récupère le nom de la section passer en paramètres, caractères par caractères (fait une copie)

ENUM\_SectionHeaderType:

Un switch pour trouver de quel type est le `sh.type` de la section courante.

ENUM\_Flags:

Un switch pour trouver de quel type est le `sh.flags` de la section courante.

GetHeader:

Retourne l'indice de header, avec en paramètre la section contenant les noms de section, un tableau de `Elf32_Shdr` (headers sections), le nom de la section recherché et le nombre de sections total existantes.

## sections\_content:

### ReadAllSections:

Cette fonction récupère le contenu de toutes les sections ainsi que leurs noms. Renvoie un tableau de `SectionContent`

### ReadSymtab:

Lis dans le fichier les sections de type `Elf32_Sym` puis renvoie un tableau de `Elf32_Sym`, avec en paramètre le contenu et le header de la section correspondante.

### ShowSymtab:

Affiche le contenu de chacune de la table de symboles, avec en paramètre le contenu et le header de la section correspondante.

### GetContent:

Lis le fichier, avec en paramètre le fichier puis le `Elf32_Shdr` correspondant. Renvoie un `SectionContent`

### CopyContent:

Copie le contenu de la section depuis un `SectionContent` donné en paramètre, puis renvoie le contenu copier sous un `char*`.

### disp\_section\_content:

Affiche, le nom, et taille d'une section avant d'afficher le contenu de cette section. Pour toutes les sections.

### ENUM\_TableSymbolType:

Retourne la chaîne de caractères associée à la valeur lue de l'attribut `st_info`.

### ENUM\_TableSymbolBinding:

Retourne la chaîne de caractères associée à la valeur lue `st_info`

### ENUM\_TableSymbolNdx:

Retourne la chaîne de caractères associée à la valeur lue de l'attribut `st_shndx`.

### ENUM\_TableSymbolVis:

Retourne la chaîne de caractères associée à la valeur lue de l'attribut `st_other`.

## sections\_relocation:

### ReadRelatab:

Lis dans le fichier les sections de type Elf32\_Rela puis renvoie un tableau de Elf32\_Rela

### ReadReltab:

Lis dans le fichier les sections de type Elf32\_Rel puis renvoie un tableau de Elf32\_Rel

### ShowRelatab:

Affiche le contenu du tableau de type Elf32\_Rel selon la forme donnée par readelf

### ShowReltab:

Affiche le contenu du tableau de type Elf32\_Rela selon la forme donnée par readelf

### ENUM\_TableRelocType:

Retourne la chaîne de caractères associée à la valeur lue de l'attribut r\_info.

## sections\_fusion:

### FusionFile:

Cette fonction est le point de départ de la fusion des 2 fichiers, appelant les sous fonctions faisant le reste.

### FusionSectionProgBits:

Concatène les sections de code correspondantes de deux fichiers binaires translatables et de mémoriser le possible changement de numéro des sections du second fichier ainsi que l'offset auquel la concaténation a eu lieu.

La fonction renvoie un `FusionContent` contenant les headers des sections fusionné et le contenu des sections, avec pour base le contenu des sections du premier fichier.

### FusionSymtab:

Concatène 2 tables de symboles provenant de fichiers donnés pour réaliser la fusion, en respectant les consignes données.

Cette fonction renvoie un `SectionContent` à 2 indice: le premier indice contient la table de symbole fusionné (.symtab), et le deuxième indice contient le noms des symboles (.strtab)

## relocations\_fusion:

### CreateFusionTab:

Concatène les tables de relocations provenant de fichiers donnés pour réaliser la fusion, en respectant les consignes données.

Cette fonction renvoie un `FusionReloc` qui contient la liste des différentes sections de relocation qui elle même contiennent les structure `Rel` et `Rela`.

## sections\_structs:

### SectionContent:

contient:

- Le nom de la section `char *`
- Le contenu de la section `char *`
- La taille en octets du contenu de section `int`

## FusionContent:

contient:

- Un tableau des Headers de sections `Elf32_Shdr*`
- Un tableau du contenu des sections `SectionContent*`
- Le nombre de sections `int`

## RelocTab:

contient:

- Un tableau des Headers de sections `Elf32_Shdr*`
- Une structure Rel `Elf32_Rel*`
- Une structure Rela `Elf32_Rela*`