

Documentation des fonctionnalités implémentées et manquantes :

NB :

- Le programme est conçu pour analyser et modifier des fichiers ELF 32 bits (ELF32) au format Little Endian (LSB).
- Toutes les fonctionnalités se limitent à ce type de fichier.
- Les autres types de fichiers ELF ne sont pas pris en charge.

Fonctionnalités implémentées

1. Affichage de l'en-tête ELF

- Lecture et affichage des champs principaux de l'en-tête ELF, incluant :
 - La plateforme cible (architecture et système).
 - La taille des mots (32 bits).
 - Le type de fichier ELF (relogable, exécutable, etc.).
 - Les informations sur la table des sections (position, taille, nombre d'entrées).
 - L'index de la table des chaînes de noms de sections.
 - La taille de l'en-tête.

2. Affichage de la table des sections

- Lecture et affichage des caractéristiques principales des sections :
 - Numéro, nom et taille des sections.
 - Type des sections (PROGBITS, SYMTAB, STRTAB, etc.).
 - Attributs principaux : allocation en mémoire, permissions (exécutable, modifiable, etc.).
 - Position (offset) des sections dans le fichier.

3. Affichage du contenu d'une section spécifique

- Lecture et affichage du contenu brut (sous forme hexadécimale) d'une section désignée par son nom ou son numéro.

4. Affichage de la table des symboles

- Lecture et affichage des informations relatives aux symboles :

- Nom, valeur et type (NOTYPE, SECTION, FUNC, etc.).
- Portée (LOCAL, GLOBAL, etc.).
- Numéro de la section correspondante.

5. Affichage des tables de réimplantation

- Lecture et affichage des informations de réimplantation pour fichiers ELF ARM :
- Cible de la réimplantation.
- Type de réimplantation.
- Index du symbole associé.

6. Renumérotation des sections

- Modification de l'organisation des sections en supprimant celles contenant des tables de réimplantation une fois résolues.

7. Correction des symboles

- Mise à jour des valeurs des symboles (adresses absolues) et correction de leur index de section selon la nouvelle organisation.

8. Réimplantations de type `R_ARM_ABS*`

Ajuster les sections de code ou de données en complétant les adresses incomplètes en fonction des informations de réimplantation. Elle prend en charge les types de réimplantation `R_ARM_ABS` pour des tailles de 32, 16 et 8 bits.

9. Réimplantations de type `R_ARM_JUMP24` et `R_ARM_CALL`

Extension du programme pour gérer les types de réimplantation `R_ARM_JUMP24` et `R_ARM_CALL`. Ces réimplantations permettent de modifier les instructions de branchement pour qu'elles pointent vers les adresses correctes après le chargement en mémoire.

10. Interfaçage avec le simulateur ARM

Ajout d'une interface pour charger les sections allouables du fichier ELF dans un simulateur ARM. Le programme initialise le compteur ordinal au point d'entrée et lance l'exécution pour vérifier l'application correcte des réimplantations.

11. Production d'un fichier exécutable non reloctable

Création d'un fichier ELF exécutable, non reloctable, avec les sections localisées aux adresses demandées. Les segments de programme nécessaires sont définis pour indiquer les zones mémoire à allouer au chargement.