

AMARCHA Mohamed
DABOUSSI Akram
D'HERIN Arthur
MAKOUNDIKA KIDZOUNOU Ifrel Rinel
NTYAM Kevin
ROUSSEAU Lenaïc
SCHWARTZMANN Victor

Phase de réimplantation d'un éditeur de liens :

Liste des fonctionnalités implémentées

Dans un premier temps, nous avons un programme **custom_readelf** permettant de reproduire le fonctionnement de la commande **arm-eabi-none-readelf** :

Ce programme attend notamment en argument une des options suivantes à l'aide de la fonction *getopt_long* :

- *-h* ou - *-file-header* : consulter l'entête du fichier ELF
- *-S* ou - *-section* : consulter la table des sections
- *-x* ou - *-hex-dump* : consulter le contenu de la section spécifiée (argument additionnel)
- *-s* ou - *-sym* : consulter la table des symboles
- *-r* ou - *-relocation* : consulter la table de réimplantation si elle existe.

La fonction adéquate est choisie selon l'option donnée.

Le programme attend également en argument le fichier ELF à consulter.

La phase de réimplantation est effectuée par le programme **reimplantation** : ce programme prend en argument le fichier ELF dont on veut effectuer la réimplantation. La structure **ELF32_FILE** correspondante est donc remplie et la table des sections est ensuite affichée. Le fichier *resultat* est ensuite créé et contient donc la version réimplantée du fichier donné en argument.

L'interfaçage avec le simulateur ARM se fait à l'aide du programme **ARM_runner** qui prend en argument le fichier ELF contenant le programme ARM que nous souhaitons exécuter.

Pour conclure, les fonctionnalités suivantes sont implémentées :

- Réimplémentation de la commande readelf, de ses options et des affichages suivants :
 - Lecture et affichage de l'entête d'un fichier ELF
 - Lecture et affichage de la table des sections d'un fichier ELF
 - Lecture et affichage du contenu de la section nommée
 - Lecture et affichage de la table des symboles
 - Lecture et affichage de la table de réimplantation d'un fichier ELF si elle existe
- Modification d'un fichier ELF permettant les parties d'implantation suivantes :
 - Renumérotation des sections
 - Correction des symboles
 - Réimplantations de type R_ARM_ABS32, R_ARM_ABS16 et R_ARM_ABS8, R_ARM_CALL et R_ARM_JUMP24 implémentées.
- Interfaçage avec le simulateur ARM