UNIVERSITÉ TOULOUSE III PAUL SABATIER

Millian Poquet
Maître de conférences
Univ. Toulouse III, IRIT, équipe Sepia
IRIT2-469, 120 cours Rose Dieng-Kuntz, 31400 Toulouse
millian.poquet@univ-tlse3.fr

Visualisation d'espaces d'adressage virtuel et physique

Contexte

Le système d'exploitation est le programme principal en charge de la gestion d'un ordinateur. Les utilisatrices agissent sur le système en créant et en contrôlant diverses instances de programmes appelées processus. Sur des architectures matérielles et des systèmes d'exploitation conventionnels, l'occupation mémoire des processus est gérée par le système d'exploitation par un mécanisme de virtualisation mémoire. Ce mécanisme consiste à faire une indirection à partir des adresses virtuelles utilisées par chaque processus vers des adresses physiques. Chaque processus a son propre *espace d'adressage* virtuel, ce qui permet entre autre à plusieurs processus de manipuler la même adresse virtuelle alors que la mémoire physique utilisée est différente. Ce mécanisme est enseigné aux étudiant·e·s à différents niveaux mais il n'est que très rarement maîtrisé, par manque de manipulation et de visualisation.

Objectifs du projet

Le but principal de ce projet est de développer un logiciel de visualisation des espaces d'adressages virtuels et physiques. Les images à générer devront pouvoir être vectorielles. Le logiciel final sera une bibliothèque en Rust. Des prototypes avec des langages et bibliothèques spécialisées dans la visualisation comme `R` + `ggplot` pourront être réalisés. Les tâches suivantes devront être réalisables grâce au logiciel développé pendant le projet :

- Visualiser les segments de l'espace d'adressage virtuel d'un processus
- Visualiser le *mapping* des pages virtuelles d'un processus vers les pages physiques
- Visualiser l'espace d'adressage virtuel de plusieurs processus en même temps, afin de montrer les pages qui sont partagées entre eux.

Dans un second temps et selon l'avancement du développement de la partie principale du projet, la suite du projet sera dédiée à l'écriture d'un programme interactif de contrôle et de visualisation dynamique de processus. Ce programme fonctionnera de manière similaire à un debugger : l'idée est de contrôler finement comment un processus cible s'exécute tout en permettant de visualiser les espaces d'adressages qui y sont associés. Techniquement, cette partie reposera sur l'interception d'appels système du processus cible via `ptrace`.