

TP7 - ASR2 - ENS Lyon - L3IF - 2019-2020

Tracking des pages soft-dirty d'un processus

Intervenants:

Stella Bitchebe (celestine-stella.ndonga-bitchebe@ens-lyon.fr)

Lavoisier Wapet (patrick-lavoisier.wapet@ens-lyon.fr)

Alain Tchana (alain.tchana@ens-lyon.fr)

(Note: ce sujet a été rédigé par Stella et Lavoisier)

Description

Information sur les pages et soft-dirty bits: Les informations concernant le mapping de l'espace d'adressage (virtuelle) d'un processus dans la mémoire physique sont contenues dans le fichier `/proc/$pid/pagemap` accessible depuis le userspace. Ce fichier contient un ensemble d'enregistrement de 64 bits chacune. Chaque enregistrement décrit une page virtuelle. Lorsqu'une page est accédée en écriture (elle devient donc sale) le bit 55 de l'enregistrement correspondant (le "soft-dirty bit") est marqué à 1 par le système d'exploitation.

Réactualisation du soft-dirty bit : Il est également possible de marquer à nouveau toutes les pages comme étant non sale c'est à dire placer le soft-dirty bit à 0. Pour cela il suffit d'écrire "4" dans le fichier `/proc/$pid/clear_refs`.

Le tracking de la modification de ce bit est très utile pour plusieurs applications user space, comme par exemple le checkpointing de l'état d'un processus ou d'un container (le système CRIU), en vu de le migrer vers un autre ordinateur.

L'objectif de ce TP est de tracker les pages virtuelles modifiées d'un processus, par un autre processus.

Partie 1 - La première partie du TP va consister à écrire un programme qui pour un processus donné (pid), parcourt sa table de page (le fichier `/proc/$pid/pagemap`) et à affiche les adresses pour lesquelles le soft-dirty bit est 1.

Partie 2 - La deuxième partie du TP consiste à compléter votre programme afin de mettre à 0 les soft-dirty bit périodiquement et de retrouver les pages modifiées entre temps. Cette boucle s'arrêtera lorsque le nombre de pages sales sera presque constant.

Recommandation: au lieu de parcourir tout l'espace d'adressage du processus (donc `/proc/$pid/pagemap`), servez vous du contenu de `/proc/$pid/maps` pour avoir les zones de l'espace d'adressage du processus qui sont mappées, et ne parcourir que ces zones.

Lien utile:

<https://lwn.net/Articles/546966/>