Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Αθηνών

Λειτουργικά Συστήματα (K22) / Περίοδος 2021-2022
2η Εργασία

Γενικές Σημειώσεις:

Στο αρχείο «spinlock.h» ορίζεται η δομή του extern struct ref_spinlock το οποίο χρησιμοποιείται για να κρατάμε έναν μετρητή αναφορών για κάθε φυσική σελίδα, όπως αναφέρει και η εκφώνηση της εργασίας, μέσω του πίνακα ακεραίων «reference_count». Στο αρχείο «kalloc.c» φτιάχνεται ένα τέτοιο global struct, με ονομασία «ref_c».

Readme:

Αρχείο «kalloc.c»

Συνάρτηση *freerange()*

Στο υπάρχων for loop προστέθηκε η αρχικοποίηση του μετρητή αναφορών της σελίδας της τρέχουσας επανάληψης με την τιμή 1 (ο μετρητής αυτός θα γίνει ξανά 0 μέσω της kfree()). Αυτή η ενέργεια είναι απαραίτητη καθώς στη συνέχεια καλείται η kfree() η οποία μειώνει τον μετρητή – άρα πρέπει πρώτα να γίνει αρχικοποίηση.

Συνάρτηση *kfree()*

Όταν κληθεί η kfree(), αρχικά μειώνεται ο μετρητής αναφορών της δοσμένης σελίδας. Έπειτα, γίνεται έλεγχος αν αυτός ο μετρητής είναι ίσος με την τιμή 0. Σε αυτή την περίπτωση σημαίνει ότι καμία διεργασία δεν χρησιμοποιεί τη συγκεκριμένη σελίδα άρα μπορεί να απελευθερωθεί, επιστρέφεται η σελίδα στη λίστα με τις διαθέσιμες σελίδες (ο κώδικας όπως ήταν αρχικά στην kfree()). Αν δεν είναι 0 τότε δε χρειάζεται να γίνει κάτι άλλο.

Συνάρτηση *kalloc()*

Προστέθηκε η αρχικοποίηση του μετρητή αναφορών της συγκεκριμένης σελίδας με την τιμή 1 (γραμμές 101-104).

Αρχείο «trap.c»

Συνάρτηση *usertrap()*

Προστέθηκε ένα else if(r_scause() == 15) στην γραμμή 68 έτσι ώστε να αναγνωρίζεται το σφάλμα σελίδας 15 - Store/AMO page fault. Όταν συμβεί το σφάλμα σελίδας με κωδικό 15 τότε πρέπει να αναγνωρίσουμε ότι είναι valid CoW fault, δηλαδή ότι πρόκειται για μια σελίδα η οποία είναι valid, user level κλπ. Πρώτα ελέγχεται αν το virtual address (το οποίο παίρνουμε καλώντας την r_stval()) είναι επιτρεπτό. Έπειτα, παίρνουμε το pagetable entry (PTE) της συγκεκριμένης σελίδας με τη βοήθεια της συνάρτησης walk(). Γίνεται έλεγχος αν το pte που πήραμε είναι valid και user level. Αποθηκεύεται το physical address του pte στη μεταβλητή pa0. Όπως αναφέρεται και στην εκφώνηση της εργασίας, δημιουργείται μία νέα σελίδα μέσω της kalloc() και αντιγράφεται η παλιά σελίδα στη νέα μέσω της memmove(). Τέλος, παίρνουμε τα flags του pte της παλιάς σελίδας, κάνουμε το pte να δείχνει στο physical address της καινούργιας σελίδας και γίνεται το PTE_W bit ίσο με 1. Καλείται η kfree() για την προηγούμενη (αρχική) read-only σελίδα.

Αρχείο «vm.c»

Συνάρτηση *uvmcopy()*

Τροποποιείται κατάλληλα η uvmcopy() έτσι ώστε να μη δημιουργούνται καινούργιες σελίδες αλλά να χρησιμοποιούνται οι υπάρχουσες pa όταν καλείται η mappages() (γραμμή 317), ώστε να αντιστοιχίζονται οι φυσικές σελίδες μνήμης του γονέα στο παιδί. Πρέπει επίσης να μηδενίζεται το flag PTE_W στις PTE τόσο του γονέα όσο και του παιδιού. Αυτό επιτυγχάνεται στις γραμμές 316 και 320 με τη χρήση bitwise AND. Σε κάθε επανάληψη κάθε σελίδας, αυξάνεται ο μετρητής αναφορών της.

Συνάρτηση *copyout()*

Η walkaddr() αυτό που κάνει είναι να πηγαίνει και να παίρνει απευθείας το physical address από ένα virtual address χωρίς να περάσει από το MMU του hardware το οποίο σημαίνει ότι δεν έχει προκληθεί page fault. Άρα αυτό που χρειάζεται να κάνουμε είναι να τροποποιήσουμε την copyout() ώστε να χρησιμοποιεί τον ίδιο μηχανισμό όπως τα σφάλματα σελίδας όταν συναντά μία σελίδα CoW. Όπως και στην συνάρτηση usertrap() του αρχείου trap.c γίνονται όλοι οι απαραίτητοι έλεγχοι και διαδικασίες, αν δηλαδή τα virtual address είναι επιτρεπτά και δεν ξεπερνάνε το MAXVA, παίρνουμε το PTE, γίνονται έλεγχοι αν είναι valid και user-level και αποθηκεύεται το physical address του pte στη μεταβλητή pa0. Στη συνέχεια, ελέγχεται αν το PTE είναι read-only, δηλαδή PTE_W bit ίσο με την τιμή 0 (γραμμή 388). Αν είναι, τότε πρέπει να ακολουθήσει όλη η διαδικασία που έγινε και στο usertrap() και έπειτα να γίνει η δουλειά που γινόταν και πριν (στην memmove που ακολουθάει (γραμμή 411), γίνεται αντικατάσταση της παλιάς σελίδας με τη νέα), διαφορετικά παραμένει ως έχει και δεν χρειάζεται να δημιουργηθεί νέα σελίδα, να αντιγραφεί η παλιά στη νέα κλπ.