Denny Shen

Design Doc

CSCI 350 Proj 4

Files modified:

proc.c

procdump():

Implemented information dump for process. Loop through the page table to acquire VPN and map it to PPN with writable bit after checking flags (write bit, shared bit...), and the display the results.

vm.c:

declared an array of char for shared page counter and a spinlock for shared lock on the shared page counter.

kvmalloc():

Initialize shared page counter and shared lock on the shared page.

inituvm():

Added code to add 1 to the counter when there is one child.

allocuvm():

Added code to add 1 to the counter when the only process is the parent.

cow():

Implemented copy-on-write replacing the original copyuvm(). First, we need to check the validity of the pte using setupkvm(), walkpgdir() and the present bit. After getting the physical address of the pte and setting up the shareable and writable bits in pte, map the flags of the child process with the parent process. And finally, increment the counter with lock on the counter array.

pagefault():

Implemented to handle cow page fault. First, we need to check the validity of the addr acquired from rcr2() and the pte acquired from walkpgdir() similar to checking process in cow(). After the validity check, we get the physical address of the pte, and set the shareable and the writable bits if there is only 1 process pointing to this page. In case there are more than 1 process pointing to this page, we need to allocate space for a new page with initialized bits while holding the lock to modify the counter array. Finally lcr3 function is used in order to properly re-install the TLB so we have only valid page tables.

trap.c:

trap():

Added case for pagefault.

syscall.h syscall.c sysproc.c usys.S:

Implemented syscall for procdump().

fork():

Change copyuvm() to cow().

Makefile:

Added testcow to Makefile.