OS2020 - Projekat

Višestruki korisnici

Projekat je namenjen kao zamena za treći i četvrti domaći zadatak. Uz projekat je dozvoljeno raditi jedan od tih domaćih zadataka u cilju nadoknađivanja bodova izgubljenih na prvom i drugom domaćem zadatku. U slučaju da se predaje treći ili četvrti domaći zadatak pored projekta, maksimalan broj predispitnih bodova ostaje 70. Nije dozvoljeno braniti sva tri zadatka (projekat, treći i četvrti domaći) i računaće se prva dva odbranjena zadatka.

Unix sistemima je karakteristično što podržavaju višestruke korisnike. Xv6 sistem trenutno ne podržava višestruke korisnike. Implementirati višestruke korisnike unutar xv6 sistema.

Kôd koji treba koristiti kao polaznu tačku za projekat može da se dohvati pomoću komande:

git clone git://github.com/RAFOperativniSistemi/xv6-raf --branch projekat projekat

# 

# Predaja i rokovi

Zadatak se predaje putem mail-a na [bmilojkovic@raf.rs](mailto:bmilojkovic@raf.rs), [mveniger@raf.rs](mailto:mveniger@raf.rs), ili [mzmiko@raf.rs](mailto:mzmiko@raf.rs).

Direktorijum koji sadrži xv6 kod (zvaće se “projekat”, ako je skinut pomoću git komande navedene kasnije u ovom dokumentu) preimenovati na sledeći način: “os\_2020\_proj\_ime\_prezime\_ind”.

Npr. “os\_2020\_proj\_student\_studentic\_rn0101”.

Arhivirati ovaj direktorijum (.zip) i arhivu poslati kao attachment uz mail. Naziv arhive mora da bude u obliku: “os\_2020\_proj\_ime\_prezime\_ind.zip”

Npr. “os\_2020\_proj\_student\_studentic\_rn0101.zip”

U tekstu mail-a obavezno navesti:

* Ime i prezime
* Broj indeksa
* Grupa, po zvaničnom spisku, ili “ponovac” za ponovce

Subject mail-a mora da bude u obliku: “[OS 2020] PROJ ime\_prezime\_ind”.

Npr. “[OS 2020] PROJ student\_studentic\_rn0101”

Rok za predaju je:

* Ponedeljak, 1. jun 23:59:59 za grupu koja sluša OS ponedeljkom
* Utorak, 2. jun 23:59:59 za grupe koje slušaju OS utorkom
* Četvrtak, 4. jun 23:59:59 za grupu koja sluša OS četvrtkom
* Subota, 6. jun 23:59:59 za grupu koja sluša OS subotom

Rok je definisan po grupi kojoj student zvanično pripada. Studeni koju slušaju vežbe van svog termina i dalje moraju da poštuju termine na osnovu zvaničnog spiska. Za sve ponovce se primenjuje najkasniji rok.

Neće se pregledati zadaci (tj. biće dodeljeno 0 poena) ako se desi bilo koje od:

* Sadržaj mail-a nije po navedenom obliku.
* Subject mail-a nije po navedenom obliku.
* Naziv arhive nije po navedenom obliku.
* Predaja se desi nakon navedenog roka.
* Kod se ne kompajluje.
* Kod nije uredno uvučen.

Odbrana projekta je obavezna. Termin za odbranu projekta će biti u toku kolokvijumske nedelje. Odbrane će se vršiti po grupama. Grupe će biti formirane i objavljene ubrzo posle roka za predaju. Ako ste iz bilo kog razloga sprečeni da prisustvujete odbrani, obavezno to najavite što pre, kako bismo mogli da zakažemo vanredni termin za odbranu.

Svrha odbrane je da se pokaže autentičnost zadatka. Ovo podrazumeva odgovaranje na pitanja u vezi načina izrade zadatka, ili izvršavanje neke izmene nad zadatkom na licu mesta. U slučaju da odbrana nije uspešna, zadatak se boduje sa -15 bodova umesto namenjenog broja bodova.

# 

# Opis sistema

Unix sistemi imaju koncepte korisnika i grupa korisnika, i svaki fajl ima skup dozvola (tj. mod) koji opisuje koja prava imaju koji korisnici i članovi kojih grupa korisnika nad fajlom.

Kod koji se bavi upravljanjem korisnicima je podeljen na deo u kernel-u i deo u korisničkom prostoru. U kernel-u su korisnici i grupe strogo numerički koncepti. Svaki proces ima UID (user id) i EUID (effective user id) i listu GID brojeva, a svaki inode ima UID, GID i mode parametar. Kernel ne prati koji sve korisnici i koje sve grupe postoje. Drugim rečima, svi UID i GID brojevi su validni. U korisničkom prostoru postoji biblioteka sa rutinama za prevođenje UID-a (tj. GID-a) u tekstualna imena korisnika (tj. grupa), a baza podataka korisnika je tekstualni fajl /etc/passwd (i /etc/group za grupe).

Biblioteku autentikacionih rutina treba implementirati u novom izvornom fajlu koji će se dodati u standardnuxv6 biblioteku ULIB koja se linkuje u sve korisničke programe. Novi korisnički programi se trebaju služiti tim funkcijama umesto da svaki za sebe implementira manipulaciju korisnika, itd.

## Konfiguracioni fajlovi

Format fajla /etc/passwd je sledeći:

username:password:UID:GID:Real\_Name:homedir

Primer fajla:

root:password:0:0:Superuser:/home/root

ken:abcd1234:1000:1000:Ken\_Thompson:/home/ken

dmr:password1:1001:1001:Dennis\_Ritchie:/home/dmr

bwk:123456:1002:1002:Brian\_Kernighan:/home/bwk

Format fajla /etc/group je sledeći:

groupname:GID:user1,user2,user3

Primer fajla:

root:0:root

ken:1000:ken

dmr:1001:dmr

bwk:1002:bwk

admin:1:root,ken

users:2:ken,dmr,bwk

Svaki korisnik ima simbolično ime, tipično identične UID i GID vrednosti, pravo ime (koje je samo sporedni podatak) i lokaciju home direktorijuma. Primetite da za svakog korisnika postoji trivijalna grupa istog imena.

Korisnik sa UID-om 0 je poseban. Tipično se naziva root (u literaturi se takođe zove superuser). Root je administrator sistema i može da čita i piše iz bilo kog fajla nezavisno od permisija (ne može da izvrši program bez execute permisija, ali mu može setovati execute permisiju). Pored toga, samo procesi koji pripadaju root-u mogu izvršiti neke sistemske pozive poput setuid() i chown().

Direktorijum /etc se inače koristi za sistemske konfiguracione fajlove, kojih na pravim sistemima ima mnogo. Za projekat je potrebno pored /etc/passwd i /etc/group dodati i fajlove /etc/issue i /etc/motd. Oni su obični tekstualni fajlovi koji sadrže tekst koje će printovati program getty. /etc/issue se printuje pre “login:” poruke, a /etc/motd, tj. “Message of The Day” se printuje nakon što se korisnik uspešno uloguje, ali pre nego što mu se pokrene shell. Tipično je issue jedan red, a motd kratak paragraf.

## Permisije/mode

Svaki fajl u inode strukturi ima jedan int koji služi kao bitfield permisija (dozvola) koji opisuje koji korisnici imaju koja prava nad tim fajlom. Ovaj parametar se često naziva mode fajla. U njegovim višim bitovima se obično čuvaju i dodatne specijalne karakteristike fajla, u našem slučaju samo setuid bit.

Mod fajla se obično ilustruje na dva načina: u oktalnom obliku i u drwxrwxrwx obliku. Primeri:

0755 = -rwxr-xr-x

0644 = -rw-r--r--

0700 = -rwx------

Najviša oktalna cifra se odnosi na permisije vlasnika fajla, srednja cifra se odnosi na članove grupe kojoj pripada fajl i najniža cifra se odnosi na sve ostale korisnike na sistemu.

Unutar jedne cifre, najviši bit se odnosi na pravo čitanja (r=4), srednji bit se odnosi na pravo upisa (w=2), i najniži bit na pravo izvršavanja programa (x=1).

Ako je u pitanju direktorijum (prvi - je d), r se odnosi na listanje članova direktorijuma, w na kreiranje novih fajlova, a x na pristup članovima.

setuid bit je posebna oznaka koju je zgodno staviti u neupotrebljene više bitove mode-a. Ona je relevantna za programe i označava da kernel treba postaviti efektivni UID (EUID) procesa pokrenutog programa na UID vlasnika fajla, umesto na UID korisnika koji je pokrenuo program. Ovo omogućava sistemskim programima poput passwd da modifikuju sistemske fajlove bez intervencije administratora. Opasno je ovaj bit postaviti neproverenim programima, naročito npr. Shell-u.

Validacija permisija se odrađuje u kernel-u, a ne u korisničkom prostoru.

Podsetnik: u C-u se brojevi mogu izraziti u oktalnom obliku ako im se stavi prefiks 0. U argumentima komande chmod se ta nula može izostaviti.

Struktura inode-a na disku se treba modifikovati tako da se pamte novi parametri (uid, gid i mode).

# Sistemski pozivi

## getuid() i geteuid()

int getuid(void) - dohvata pravi user id procesa koji ga je pozvao.

int geteuid(void) - dohvata efektivni user id procesa koji ga je pozvao.

## setuid()

int setuid(int uid) - postavlja pravi user id i efektivni user id trenutnog procesa. Dozvoliti da se ovaj sistemski poziv može pozvati samo od strane root user-a (proces sa uid = 0).

## setgroups()

int setgroups(int ngroups, int \*gids) - postavlja listu grupa kojima vlasnik procesa pripada. Dozvoliti da se ovaj sistemski poziv može pozvati samo od strane root user-a (proces sa uid = 0).

## stat() i chmod()

int stat(const char \*pathname, struct stat \*buf) - modifikovati postojeći stat da vraća sve nove članove inode strukture.

int chmod(const char \*pathname, int mode) - menja mod bitove za zadati fajl. Ovaj sistemski poziv može da pozove ili trenutni vlasnik fajla ili root user.

## chown()

int chown(const char \*pathname, int owner, int group) - menja vlasnika i grupu za određeni fajl. Dozvoliti da se ovaj sistemski poziv može pozvati samo od strane root usera (proces sa uid = 0).

## open() - modifikacija

Kreiranje novih datoteka u xv6 se radi tako što se open() sistemski poziv izvrši uz flag O\_CREATE. Treba izmeniti ovaj slučaj sistemskog poziva tako da novokreirane datoteke po default imaju mode 644. Ovo inače nije standardno ponašanje u Unix sistemima, već treba da postoji poseban parametar za open() preko kojeg se navodi mode za nove datoteke.

# Korisnički programi

## getty

Korisnički program getty se izvršava na početku rada sistema. Izvršava ga init proces. Glavni zadatak ovog programa je da dobije korisničko ime i šifru od korisnika, i da nakon toga startuje korisnikov sh proces u njegovom home direktorijumu.

Kada se proces startuje, on očisti ceo ekran i ispiše “Pre-login” poruku iz fajla /etc/issue, ako postoji. Nakon toga se ispisuje poruka koja traži korisniku da unese svoje korisničko ime. Kada korisnik unese svoje korisničko ime, traži mu se šifra. Datoteka /etc/passwd se koristi za autentifikaciju korisnika. Pri unošenju šifre ne prikazivati unešen tekst na ekranu. Nakon što se korisnik uloguje, ali pre nego što se pokrene sh, getty treba da ispiše “Message of The Day” poruku, ako je ona definisana. Ova poruka se nalazi u /etc/motd.

getty proces treba da pripada root korisniku, dok korisnikov sh proces treba da pripada korinisku koji se ulogovao u sistem.

Kada se sh proces završi, getty treba da obriše ekran, ponovo ispiše issue i ponovo traži korisniku da se uloguje. Izlazak iz sh procesa može da se izvrši kombinacijom tastera CTRL+D.

## passwd

Sintaksa:

passwd [USER]

Korisnički program passwd služi za izmenu korisničke šifre. Obični korisnički nalog može da menja samo svoju šifru, dok root nalog može da menja šifru za bilo kog korisnika. Naziv korisnika za kojeg se menja šifra se navodi kao opcioni argument na komandnoj liniji. Ako se taj argument izostavi, podrazumeva se da se menja šifra za trenutnog korisnika. Kada obični korisnički nalog menja svoju šifru, prvo mora da unese svoju trenutnu šifru. Root korisnik može da preskoči ovaj korak.

Pri unošenju nove šifre, ona mora da se unese dva puta identično. Šifra mora da bude duža od 6 karaktera. Ako bilo koji od ova dva uslova nije ispunjen, šifra se ne menja i program se završava uz ispis odgovarajuće poruke o grešci.

Pri unošenju svake šifre (stare ili nove, oba puta) ne prikazivati unešen tekst na ekranu.

passwd je jedini program koji mora imati postavljen setuid bit (da bi mogao modifikovati /etc/passwd kada je pokrenut od strane običnog korisnika).

## useradd

Sintaksa:

useradd [-d dir] [-u uid] [-c name] login

Program useradd se koristi za dodavanje novog korisnika u sistem. Korisničko ime za novog korisnika se navodi kao obavezan poslednji argument na komandnoj liniji. Ovaj program takođe automatski kreira novu grupu za ovog korisnika koja se zove isto kao korisnik, kao i njegov home direktorijum.

Ovaj program ne podešava šifru za korisnika.

Ako se navede opcija koja nije podržana, program prijavljuje grešku i završava se bez kreiranja korisnika.

Opcija -d se koristi za navođenje direktorijuma koji će se koristiti kao home za ovog korisnika. Default vrednost je /home/korisnicko\_ime. Ako navedeni direktorijum ne postoji, kreirati ga. Ako dođe do greške pri kreiranju direktorijuma, prijaviti grešku i završiti program bez kreiranja korisnika.

Opcija -u se koristi za navođenje UID za korisnika. Ova vrednost mora biti jedinstvena. U slučaju da nije, program prijavljuje grešku i završava se bez kreiranja korisnika. Default vrednost za ovaj parametar je najmanji broj koji je veći od 1000 i nije već zauzet.

Opcija -c se koristi za navođenje pravog imena korisnika.

Ako program ne može da pristupi /etc/passwd ili /etc/group datoteci, prijavljuje grešku i završava se bez kreiranja korisnika.

## groupadd

Sintaksa:

groupadd [-g gid] naziv\_grupe

Program groupadd se koristi za dodavanje nove grupe korisnika. Naziv grupe se navodi kao obavezan poslednji argument na komandnoj liniji.

Ako se navede opcija koja nije podržana, program prijavljuje grešku i završava se bez kreiranja grupe.

Opcija -g se koristi za navođenje GID za grupu. Ova vrednost mora biti jedinstvena. U slučaju da nije, program prijavljuje grešku i završava se bez kreiranja grupe. Default vrednost za ovaj parametar je najmanji broj koji je veći od 1000 i nije već zauzet.

Ako program ne može da pristupi /etc/group datoteci, prijavljuje grešku i završava se bez kreiranja grupe.

## usermod

Sintaksa:

usermod [-l login] [-u UID] [-c name] [-d dir] [-m] [-G grupa…] [-a] login

Program usermod se koristi za modifikaciju postojećih korisnika. Korisničko ime korisnika koji se modifikuje se navodi kao obavezan poslednji argument na komandnoj liniji. Ako ne postoji korisnik sa tim korisničkim imenom, program prijavljuje grešku i završava se bez izmene korisnika.

Ako se navede opcija koja nije podržana, program prijavljuje grešku i završava se bez kreiranja korisnika.

Opcija -l se koristi za promenu korisničkog imena za korisnika. Novo korisničko ime treba da bude jedinstveno. Ako nije, program prijavljuje grešku i završava se bez izmene korisničkog imena.

Opcija -u se koristi za promenu korisnikovog UID. Novi UID treba da bude ceo broj i da bude jedinstven. Ako to nije slučaj, program prijavljuje grešku i završava se bez izmene UID. Sve datoteke u korisnikovom home direktorijumu koje pripadaju ovom korisniku treba izmeniti tako da reflektuju ovu izmenu.

Opcija -c se koristi za promenu korisnikovog pravog imena.

Opcija -d se koristi za promenu korisnikovog home direktorijuma. Novi direktorijum se kreira ako već ne postoji. Ako se uz ovu opciju navede i opcija -m, čitav sadržaj starog direktorijuma treba premestiti u novi. Ako dođe do greške pri kreiranju direktorijuma, ili premeštanju podataka prijaviti grešku i završiti program bez toga. Opcija -m treba da ne radi ništa ako nema opcije -d.

Opcija -G se koristi za dodavanje korisnika u jednu ili više grupa. Grupe se navode razdvojene zarezom, i treba da sve već postoje. Ako neka grupa ne postoji, prijaviti grešku o tome, ali dodati korisnika u grupe koje postoje. Ako je korisnik već bio u nekim grupama osim svoje lične grupe, on se briše iz njih, osim ako se navede opcija -a uz ovu opciju. Ako je opcija -a navedena, onda korisnik ostaje u grupama u kojima je prethodno bio. Opcija -a treba da ne radi ništa ako nema opcije -G.

Ako program ne može da pristupi /etc/passwd ili /etc/group datoteci, prijavljuje grešku i završava se bez modifikovanja korisnika.

## chmod

Sintaksa 1:

chmod mode file...

Uz pomoć sistemskog poziva chmod() postavlja mode fajla (ili fajlova) na prosleđeni mode.

Sintaksa 2:

chmod mode\_fmt file...

Gde mode\_fmt je sastavljen od tri dela. Prvi deo mode\_fmt je za koga se menja mode (opcije su: u za user, g za group, o za others i a za all). Drugi deo mode\_fmt je oznaka + ili - koja predstavlja da li dodajemo ili oduzimamo neku od permisija odabranom targetu) i treći deo je permisija koju dodajemo/oduzimamo (opcije su: r - read, w - write, x - execute i s - setuid bit).

Primer korišćenja:

chmod a+x file1 - dodaje svima pravo na execute za file1.

chmod o-w file1 file2 - oduzima pravo pisanja za other usere (useri koji nisu vlasnik fajla i deo grupe) za file1 i file2.

Napomena: za drugu sintaksu će biti potrebno koristiti i sistemski poziv stat() pored sistemskog poziva chmod().

Ako format nije validan, ili nije moguć pristup datotekama program prijavljuje grešku i završava se bez izmene datoteka.

## chown

Sintaksa:

chown [owner][:[group]] file…

Program chown menja vlasnika za jednu ili više datoteka. Datoteke kojima se menja vlasnik se navode kao poslednji argument na komandnoj liniji. Navedenim datotekama može da se menja koji korisnik i/ili grupa su njihov vlasnik. I za korisnika i za grupu je dozvoljeno koristiti UID/GID ili korisničko ime / ime grupe. Ako se menjaju i korisnik i grupa, oni treba da budu razdvojeni znakom :. Ako se menja samo korisnik, dvotačka i grupa treba da se izostave. Dozvoljeno je napisati i samo znak dvotačka nakon ID / naziva korisnika i onda se grupa postavlja na korisnikovu grupu. Dozvoljeno je napisati i znak : i nakon njega broj ili ID / naziv grupe. Tada se ovaj program ponaša identično kao chgrp. Ako se napiše samo znak :, onda se ni korisnik ni grupa ne menjaju.

Ako se ovaj format ne ispoštuje, ili navedeni korisnik ili grupa ne postoje, program ispisuje grešku i završava se bez izmene navedenih datoteka.

## chgrp

Sintaksa:

chgrp group files…

Ovaj program menja grupu koja je vlasnik za navedene datoteke. Datoteke kojima se menja vlasnik se navode kao poslednji argument na komandnoj liniji. Grupa može da se navede kao GID ili pomoću naziva grupe.

Ako navedena grupa ne postoji, program ispisuje grešku i završava se bez izmene navedenih datoteka.

## ls - modifikacija

Program ls modifikovati tako da ispis izgleda isto kao kod standardnog Unix ls programa sa -l flagom. To podrazumeva:

* Permisije za svaku datoteku i direktorijum u obliku: drwxrwxrwx
* Naziv korisnika i grupe koji su vlasnici datoteke / direktorijuma.

# mkfs

Program mkfs, koji kreira virtualni disk za xv6 (fs.img), treba se modifikovati tako da kreira direktorijum /etc i popunjava ga. Obični fajlovi (oni koji ne pripadaju u /bin ili /etc) se trebaju kopirati u /home/root umesto u /home.

Svim fajlovima u fajl sistemu dodeliti odgovarajuće vlasnike, grupe i permisije. Većina fajlova pripadaju root korisniku i čitljivi su od strane svih korisnika. Home direktorijumi pripadaju svojim vlasnicima i mogu (ali ne moraju) biti ne čitljivi ostalim korisnicima.

Na pravim Unix sistemima se root-ov home direktorijum obično nalazi u /root umesto /home/root, ali u okviru ovog zadatka nije bitno.

# 

# 

# Bodovanje

Zadatak se boduje na sledeći način:

* Izmena mkfs = 4 boda
* Izmena file sistema = 4 boda
* Auth biblioteka = 6 bodova
* Validacija permisija = 3 boda
* Sistemski pozivi = 5 bodova
  + getuid(), geteuid() = 1 bod
  + setuid(), setgroups() = 1 bod
  + chmod() = 1 bod
  + chown() = 1 bod
  + stat() = 1 bod
* Korisnički programi = 18 bodova
  + getty = 3 boda
  + passwd = 2 boda
  + useradd, groupadd = 2 boda
  + usermod = 3 boda
  + chmod = 3 boda
  + chown, chgrp = 2 boda
  + modifikovan ls = 3 boda

U slučaju da je neka od stavki implementirana parcijalno, biće dodeljeni parcijalni poeni.