IX - Sigurnost i zaštita OS

S A D R Ž A J

9.1 - Sigurnost i zaštita OS

- Problem zaštite postoji kod svih računarskih sistema, a posebno kod sistema koji su umreženi tj. povezani na **Internet**.
- Zaštita je postala jedno od **najvažnijih pitanja** u svakom sistemu.
- **Apsolutna zaštita** računarskih sistema se **ne može ostvariti**.
- **Osnovni cilj** je da se obezbedi visok nivo zaštite do koga se dolazi sveobuhvatnim pristupom rešavanju zaštite sa stalnim razvijanjem novih mehanizama u skladu sa bezbednosnim problemima koji nastaju.
- Kod savremenih računarskih sistema primenjuje se sistem zaštite na više nivoa:
    1. na nivou **mreže**,
    2. na nivou OS,
    3. na nivou **aplikacije**,
    4. na nivou **baze podataka**
    5. **proceduralna zaštita**.
- Osnovna potreba za zaštitom u okviru OS nastaje zbog deljenja resursa kao što su **memorija**, **U/I uređaji**, **programi** i **podaci**.
- Operativni sistem ima **značajnu ulogu** u rešavanju problema zaštite.

9.1 - Sigurnost i zaštita OS

- Operativni sistem je osnovna komponenta računarskih sistema.
- OS su usko povezani sa hardverom jer sakrivaju hardverske detalje od programera i predstavljaju najniži sloj softvera vidljiv korisnicima.
- OS možemo posmatrati kao upravljač resursima (*resource manager*), odgovoran za fer deljenje resursa između različitih procesa u sistemu.
- OS kontroliše pristup aplikacija memoriji i dodeljivanje CPU vremena.
- Aplikacije se izvršavaju kao servisi na nivou OS, a programeri aplikacija ne znaju detalje potrebne za razvoj sigurne aplikacije.
- OS je vrlo logično mesto za sprovođenje bezbednosnih mera jer ako OS ne obezbedi bezbednost to kompromituje celokupnu bezbednost.
- Kod današnjih OS bezbednosne funkcije i mehanizmi se retko aktiviraju inicijalno tj. po ***default***-u.
- Bezbedna „***out-of-the-box***“ instalacija je izuzetak pre nego pravilo.
- Da bi se postigao prihvatljiv nivo bezbednosti, administrator mora, nakon završetka instalacije da podesi bezbedonosne parametre OS.
- Obezbeđenje OS nije trivijalan zadatak jer su moderni OS veliki i složeni pa se zahteva **jako dobro poznavane** OS i sistema zaštite.

9.1 - Zahtevi sistema zaštite OS

- Kod današnjih sistema koji se zasnivaju na komunikaciji korisničkog procesa i servisa koji obezbeđuju neku vrstu usluge ili obrade podataka postoje **sledeći zahtevi sistema zaštite**:
  1) **Međusobna autentikacija**,
  2) **Kontrola pristupa ili autorizacija**,
  3) **Zaštićena komunikacija**,
  4) **Neporicanje** slanja, odnosno prijema podataka,
  5) **Ne ponavljanje slanja**,
  6) **Nema odbijanja servisa**.
- **Međusobnom autentikacijom** se obezbeđuje verifikacija identiteta obe strane koje učestvuju u komunikaciji.
- Tek nakon završene međusobne autentikacije se može nastaviti dalja komunikacija između dva računara.
- **Kontrolom pristupa** se obezbeđuje da samo autorizovani korisnici mogu pristupiti traženim podacima.
- U suprotnom, neautorizovani korisnik bi mogao da naruši integritet podataka, tako što bi mogao da ih mjenja.

9.1 - Sigurnost i zaštita OS

- **Zaštićena komunikacija** garantuje tajnost podataka koji se prenose preko komunikacionog kanala.
- **Neporicanje** ima značenje da ni jedna strana koja učestvuje u komunikaciji ne može da poriče slanje, odnosno prijem podataka
- **Ne ponavljanje slanja** eliminiše mogućnost da treća strana kopira celu ili deo poruke i nakon toga vrši ponovo slanje tih istih **podataka.**
- **Zahtev da nema odbijanja servisa** obezbeđuje da nema degradacije performansi datog sistema i garantuje legitimnim korisnicima sistema da mogu da koriste željeni servis.
- **Bezbednost OS** se realizuje kroz odgovarajuću zaštitu **4 elemenata**:

1. **Poverljivost** (***Confidentiality***) - sprečava ili minimizuje neovlašćeno pristupanje i objavljivanje podataka i informacija.
2. **Integritet**(***Integrity***) - osigurava ispravnost podataka sa kojima se radi
3. **Raspoloživost** (***Availability***) - omogućuje dostupnost i upotrebljivost resursa na zahtev autorizovanog sistemskog entiteta.
4. **Autentičnost** (***Authenticity***) - omogućava proveru identiteta korisnika

9.1 - Sigurnost i zaštita OS

- Princip **raspoloživosti** se vezuje za **hardver**, **softver** i **podatke**.
- Hardver je najviše podložan napadima (slučajnih i namernih oštećenja)
- Ključna pretnja za softver predstavlja napad na **raspoloživost**
- **Poverljivost** i **integritet** zasniva se na **3 glavne stavke**:

1. **Model zaštite** (*Protection model*) je najvažniji aspekt zaštite; ako je on slab, OS će biti ugrožen čak iako je sve ostalo u njemu savršeno
   - ✓ Zaštitni modeli treba međusobno da korespondiraju.
   - ✓ Dobro poznati modeli uključuju Bell-LaPadula hijerarhijski model obaveznog pristupa i Biba model hijerarhijskog integriteta.
   - ✓ OS mogu imati tendenciju da koriste nekoliko zaštitnih modela
2. **Mogućnosti** (*Capability)* predstavljaju alate i funkcije koje OS koristi za implementaciju datog modela i uključuju specifične kontrole pristupa ili dostupne privilegije kao i način na koji su one definisane
3. **Uverenja** (*Assurances*) su način utvrđivanja da su modeli implementirani korektno i da ne mogu biti zaobiđeni; primer, korišćenje mikrokernela omogućava da svi aspekti zaštitnih modela budu implementirani kroz jednu tačku, poznatu kao ***reference monitor***

9.2 – Slabosti OS

1. **Nepravilna ulazna provera** - neophodno je pažljivo proveriti ulaze u softverske rutine, tj., izvršiti proveru ulaza (*input validation*). Provera se može odnositi na broj obezbeđenih parametara, tip svakog parametra, ili jednostavno osigurati da količina ulaznih podataka nije veća od dodeljenog bafera za smeštaj podataka.

2. **Slabi kriptografski algoritmi**-više slabih algoritama u kriptografskim sistemima su još jedan od sigurnosnih problema. U OS kriptografski algoritmi se već koriste za šifrovanje lozinki. Ako algoritam koji se koristi nije dovoljno jak, napadač može izvući „čistu“ lozinku.

3. **Slabi protokoli za autentifikaciju** - pre nego što korisnik dobije dozvolu za pristup resursima, on mora dokazati svoj identitet. Proces se naziva autentifikacija. Većina autentifacionih sistema se zasniva na zajedničkoj tajni uključenih strana. Autentifikacioni mehanizam je najčešće samo lozinka, tajna reč poznata samo sistemu i koriniku. Međutim, ostvarivanje bezbedne autentifikacione procedure je složen zadatak, posebno u distribuiranom okruženju.

9.2 – Slabosti OS

4. **Nesigurni „*Bootstraping*“** - sistem inicijalizacije je veliki sigurnosni problem u današnjim OS. Svi sistemi su ugroženi tokom podizanja (*bootstraping*). Na primer, mnogo hakera je otkrilo da je SunOS vrlo lako restartovati u jednokorisničkom režimu. Komande unete u tom modu rade sa *root* privilegijama, i moguće je proširiti ove privilegije na server. Windows NT sistemi koji se izvršavaju na PC-ma mogu biti restartovani iz drugih OS, kao što je MS-DOS. Kada je neki drugi sistem pokrenut na PC-u, NTFS volumen može biti montiran na njemu Pristup fajlovima na novo-montiranom volumenu će premostiti kontrolne mehanizme pristupa implementirane u Windows NT.
5. **Konfiguracione greške** - loše konfigurisanje OS može biti ozbiljan problem jer može da omogući napadaču laku dostupnost resursa

- **Prva četiri** su zasnovana na tehničkim ili sistemskim osnovama, dok je **peti** povezan sa organizacionim i upravljačkim problemima.
- Ne potiču sve slabosti iz samog OS, jer loše izabrane lozinke nisu slabost OS već neopreznost korisnika.

9.3 - Zaštitne mere kod OS

- Zaštitne mere se uvek primenjuju na višem nivou, jedan nivo je fizički i prva linija odbrane je fizička zaštita, ako hardver nije fizički dostupan neovlašćenim osobama smanjuje se rizik da bude oštećen.
- Zaštita na transportnom i mrežnom nivou je danas jako značajna, jer se stara o bezbednom i pouzdanom prenosu podataka
- OS mora da zaštiti sebe i sistem od slučajnog ili namernog oštećenja
- U mehanizme zaštite na ovom nivou spadaju:

1. ***Identifikacija korisnika operativnom sistemu*** - traži da svaki korisnik koji pristupa sistemu ima korisničko ime na sistemu i odgovarajuću lozinku. Tako operativni sistem zna da li se radi o pravom, ovlašćenom korisniku ili ne, i na taj način korisniku dozvoljava ili ne dozvoljava da se koristi uslugama operativnog sistema.
2. ***Kontrola pristupa na nivou sistema datoteka*** - kontrola pristupa je primenjena u sve operativne sisteme, u listama za kontrolu pristupa dato je ko može da pristupi datoteci ili direktorijumu i šta s tom datotekom ili direktorijumom može da radi.

9.3 - Zaštitne mere kod OS

3. ***Kriptografske mere zaštite*** - svaki podatak na računaru se štiti šifrovanjem, postoje programi koji šifruju kompletne diskove, prenosive medijume. Šifrovanje podataka na diskovima može se obaviti na nivou datoteka i drajvera.
4. ***Kontrola daljinskog pristupa*** - operativni sistem mora da obezbedi kontrolu daljinskog pristupa sistemu, i mora imati mrežnu barijeru koja filtrira podatke na mrežnom i transportnom sloju i obezbedi kontrolu pristupa mreži za različite procese.
5. ***Praćenje sigurnosnih događaja*** - sigurnosni događaji su akcije usmerene na resurse koji su zaštićeni nekom sigurnosnom merom, kao što je kontrola pristupa.
6. ***Izrada rezervnih kopija značajnih podataka*** - kopije treba praviti zbog gubljenja podataka koje nekada mogu biti veoma značajne, kada se nešto desi sa podacima koji se pamte na medijumima.
7. ***Izrada plana restauracije*** – ovde se navodi koji su podaci važni i daje zaštitna mera koju treba preduzeti u slučaju proboja sigurnosnih mera

9.4 - Mehanizmi zaštite OS

- Koncept multiprogramiranja uvodi deljenje resursa između korisnika.
- Ovo uključuje deljenje memorije, ulazno/izlaznih uređaja i podataka.
- Mogućnost deljenja ovih resursa uvodi potrebu za zaštitom.
- OS može da obezbedi različite stepene zaštite za različite objekte
- OS treba da uravnoteži politiku omogućavanja deljenja, sa potrebom da zaštiti resurse od individualnih korisnika.
- Operativni sistem može ponuditi sledeće nivoe zaštite:

1. **Bez zaštite** (*No Protection*) - delovi koda sa kritičnim sekcijama se izvršavaju u različito vreme. Ovo je odgovarajući način kada se osetljive procedure izvršavaju u posebnim vremenskim intervalima;
2. **Izolacija** (*Isolation*) - kada se svaki proces izvršava nezavisno od drugih procesa bez deljenja resursa i bez međusobne komunikacije. Svaki proces ima svoj adresni prostor, datoteke i druge objekte. Ovaj pristup podrazumeva da svaki proces radi odvojeno od drugih procesa, bez zajedničkih resursa. Svaki proces ima sopstveni adresni prostor, fajlove, i ostale potrebne resurse.

9.4 – Mehanizmi zaštite OS

3. **Deliti sve ili ništa** (*Share all or Share Nothing*) - ovde vlasnik resursa određuje da li će resurs biti javan ili privatan. Ako je resurs javni svako mu može pristupiti, ako je privatni može mu pristupiti samo vlasnik;
4. **Deoba preko ograničenja pristupa** (*Share via Access Limitation*) - OS kod svakog pristupa datog korisnika nekom resursu proverava dozvolu pristupa. Na taj način OS obezbeđuje da samo autorizovan korisnik može da pristupi traženom resusu. OS radi kao međusloj između korisnika i resursa osiguravajući samo dozvoljene pristupe.
5. **Deoba preko dinamičkih mogućnosti** (*Share via Dynamic Capabilities*) - proširenje koncepta kontrole pristupa omogućavajući dinamičko kreiranje dozvola za deljene resurse.
6. **Ograničenje korišćenja objekta** (*Limit use of an object*) - pored ograničavanja pristupa objektu ograničavaju se i operacije koje se mogu vršiti nad objektom. **Višekorisnički OS** moraju da obezbede zaštitu od neautorizovanog pristupa jednog korisnika resursima drugog korisnika – postiže se putem korisničkih i računarskih naloga

9.4 - Zaštita memorije

- U multiprog. okruženju, zaštita memorije je osnovni faktor bezbednosti
- Pored bezbednosti problem je i ispravno funkcionisanje svih procesa
- Razdvajanje memorije između različitih procesa je lako ostvarljivo uz pomoć šeme virtuelne memorije.
- **Segmentacija** i ***Paging*** su efikasne metode za upravljanje memorijom.
- Ukoliko je potrebna potpuna izolacija, onda OS mora obezbediti da je svaki mem.segment/stranica dostupna samo procesu kome je dodeljena
- U tabeli stranica i/ili segmenta ne sme postojati duplikati zapisa ali isti segment/stranica se može pojaviti u više tabela ako je omoguć. deljenje
- Da bi se realizovalo deljenje, moguće je da segment bude referenciran u tabelama segmenata od strane više procesa.
- Segmentacija je pogodna za implementaciju polisa zaštite i deljenja, jer je za svaki segment definisana dužina kao i osnovna adresa u tabeli.
- Program ne može pristupiti memor.lokacijama izvan granica segmenta
- U *paging* sistemu, programer ne vidi strukturu stranica programa
- Mere za kontrolu pristupa u sistemima za obradu podataka se dele na: korisniku (***User-Oriented***) i podacima orijentisane (***Data-Oriented***).

9.4 – Korisnička kontrola pristupa

- Korisnička kontrola pristupa se naziva autentifikacija (*Authentication*) i predstavlja glavni problem zaštite kod svih OS.
- Većina autentifikacionih metoda su bazirane na:
  1. nečemu što korisnik zna - navođenje poverljivih informacija(lozinka)
  2. nečemu što korisnik ima - specijalan hardver (ključ ili ID kartica)
  3. nečemu što je korisnik - biološki atributi korisnika (otisak prsta, snimak mrežnjače oka, potpis, izgled lica, glas i td.
- Tradicionalan mehanizam za korisničku kontrolu pristupa deljenom resursu je prijavljivanje(***logging***): zahteva korisničko ime(ID) i lozinku
- **Tajnost lozinke** je način kojim se na ovaj način obezbeđuje zaštita.
- Od načina primene sistema lozinki najviše zavisi bezbednost OS.
- Zato većina današnjih sistema ne omogućava korisnicima unos onih lozinki koje nisu dovoljno sigurne i koje se nazivaju **slabe lozinke**: ime ili prezime, nadimak, ime člana porodice ili devojke, itd.
- Na Internetu postoji posebna metoda identifikacije koja podrazumeva korišćenje sertifikata (X.509)
- Lozinke predstavljaju najranjivije mesto pa su jedan od omiljenih objekata koje zlonamerni napadači koriste za narušavanje sigurnosti

9.4 - Korisnička kontrola pristupa

- U cilju bolje zaštite, potrebno je da OS podrži jake lozinke, odnosno:
  - ✓ kontroliše da lozinke budu reči koje se ne mogu naći u rečniku,
  - ✓ kontroliše da lozinke budu reči najmanje 6-8 karaktera dužine,
  - ✓ kontroliše da lozinke budu od slova, brojeva i specijalnih znakova
  - ✓ obezbedi da lozinke imaju period važenja,
  - ✓ ograniči broj pokušaja prijavljivanja na OS sa pogrešnom lozinkom, što će nakon max. broja pokušaja automatski zabraniti korisnič.nalog.
- Najvažnija lozinka u datom sistemu je lozinka **sistem administratora**
- Upravo zbog toga najveći broj napada na sistem ima za cilj pronalaženje lozinke sistem administratora.
- Sistem zaštite koji se zasniva na lozinkama se može narušiti pogađanjem lozinki.
- Drugi način je metodom grube sile gde se korišćenjem današnjih računara veoma brzo može pretražiti kompletan skup mogućih lozinki čija je maksimalna dužina unapred poznata.

9.4 - Korisnička kontrola pristupa

- Neka od osnovnih pravila kojih se treba držati kod lozinki:
  - ✓ čuvati poverljivost lozinki,
  - ✓ ne beležiti lozinke na papir, u nekom fajlu, E-mail-u
  - ✓ lozinke se ne smeju odavati drugim korisnicima, čak ni administratorima, odgovornim osobama i sl.,
  - ✓ korisnici ne smeju menjati lozinke ukoliko sumnjaju na nepravilnosti u radu servisa
  - ✓ birati kvalitetne lozinke, duge minimalno 6 znakova, da nisu vezane uz imena, datume, telefonske brojeve i sl.,
  - ✓ lozinke moraju sadržati različite oznake: brojeve, mala i velika slova, i ako je moguće i specijalne znakove,
  - ✓ izbegavati ponovnu uptrebu starih lozinki,
  - ✓ izbegavati lozinke koje već koriste na drugim sistemima,
  - ✓ redovno periodično menjati lozinke itd.

9.4 – Korisnička kontrola pristupa

- Narušavanje sistema zaštite koji se zasniva na lozinkama se može narušiti kao rezultat vizuelnog ili elektronskog monitoringa.
- **Vizuelni monitoring** nastaje gledanjem u tastaturu prilikom unosa korisničkog imena i lozinke.
- **Elektronski monitoring** se može uraditi pomoću ***sniffing*** alata kojim se može snimiti identitet korisnika i njegova lozinka.
- Kriptovanje podataka koji sadrže lozinku rešava ovaj problem
- Za to se koriste jednosmerne **heš funkcije**
- Korisnička kontrola pristupa u distribuiranim okruženjima može biti:

1. **Centralizovana -** mreža pruža servis za prijavljivanje, određujući kome je dozvoljeno da koristi mrežu i kojim korisnicima je dozvoljeno povezivanje.
2. **Decentralizovana -** tretira mrežu kao transportnu komunikacionu vezu, a odredišni host obavlja uobičajne procedure prijavljivanja.

- U mnogim mrežama, koriste se dva nivoa kontrole pristupa

9.4 - Kontrola pristupa podacima

- Nakon uspešnog prijavljivanja, korisniku se može odobriti pristup jednom ili skupu hostova i aplikacija.
- Potrebne su informacije o kontroli pristupa podacima.
- U tom svetlu, OS nudi dva tipa zaštitnih modela:
  1. obavezna kontrola pristupa (MAC) i
  2. diskretna kontola pristupa (DAC).
- U terminologiji sigurnosti računara pasivni resursi se nazivaju **objekti** (***objects***), a aktivni entiteti koji koriste resurse su **subjekti** (***subjects***).
- Tipični **objekti** su: fajlovi, direktorijumi, memorija, štampači, itd., a u **subjekte** spadaju: korisnici, procesi, ...
- Uloge zavise od situacije: npr., proces može zahtevati pristup nekim resursima (subjekat), a kasnije biti meta zahteva za pristup (objekat).
- U obaveznoj kontroli pristupa, takođe nazvanoj ***multilevel*** kontrola pristupa, objekti (informacija) su klasifikovani po hijerarhijskim nivoima sigurnosne osetljivosti (najčešće:*top-secret,secret,confidential*) gde se subjektima (korisnicima) dodeljuju sigurnosna uverenja.

9.4 - Kontrola pristupa podacima

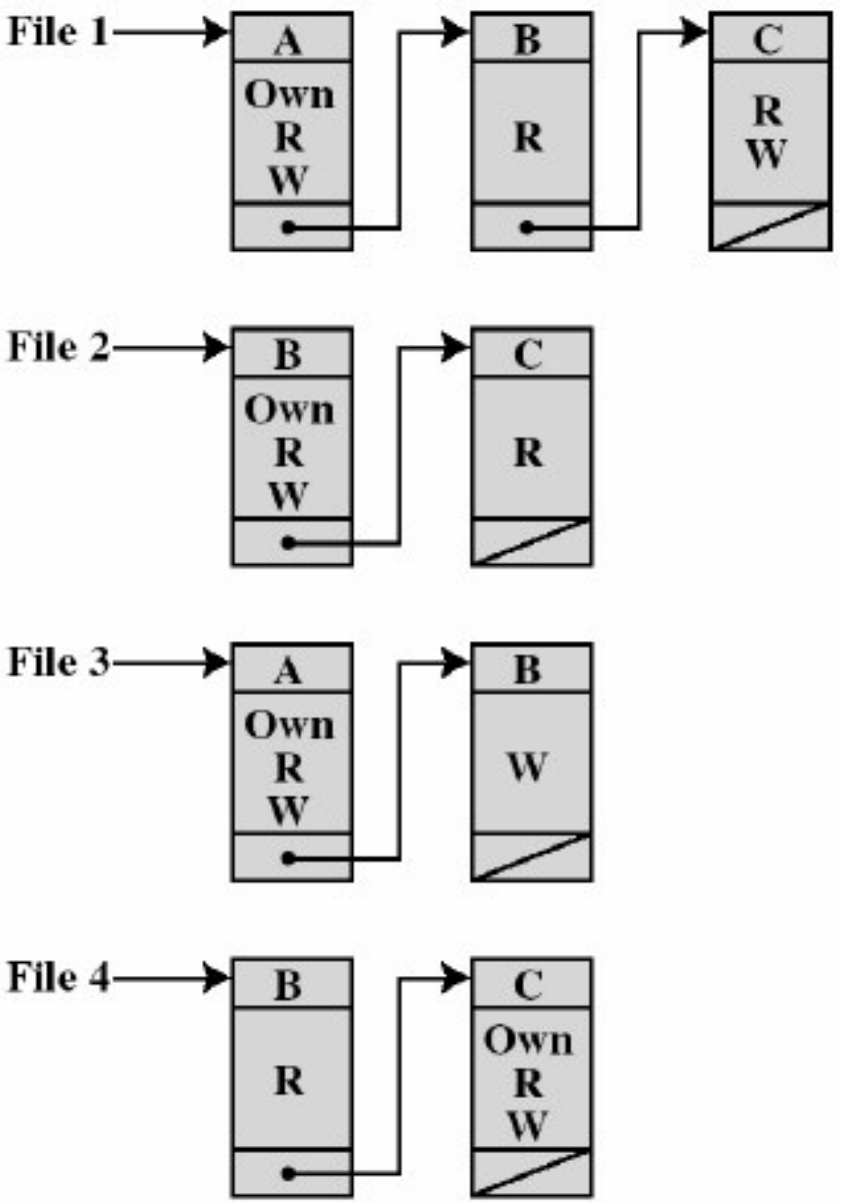
- Pristup određenog subjekta nekom objektu se odobrava ili odbija u zavisnosti od relacije između dozvola koje ima subjekat i bezbednosnih klasifikacija objekta.
- **Lattice** i **Bell-LaPadula** model su bazirani na MAC.
- U diskretnoj kontroli pristupa svaki objekat ima jedinstvenog vlasnika
- Vlasnik ostvaruje svoje diskreciono pravo preko dodeljivanja dozvola za pristup.
- **Lampson** je uveo model matrice pristupa za DAC.
- Ovde redovi predstavljaju objekte a kolone subjekte

	Text.doc	password	Slika.bmp
Mirko	rw	r	x
Vesna	w	r	-
Miloš	rw	rw	rwx

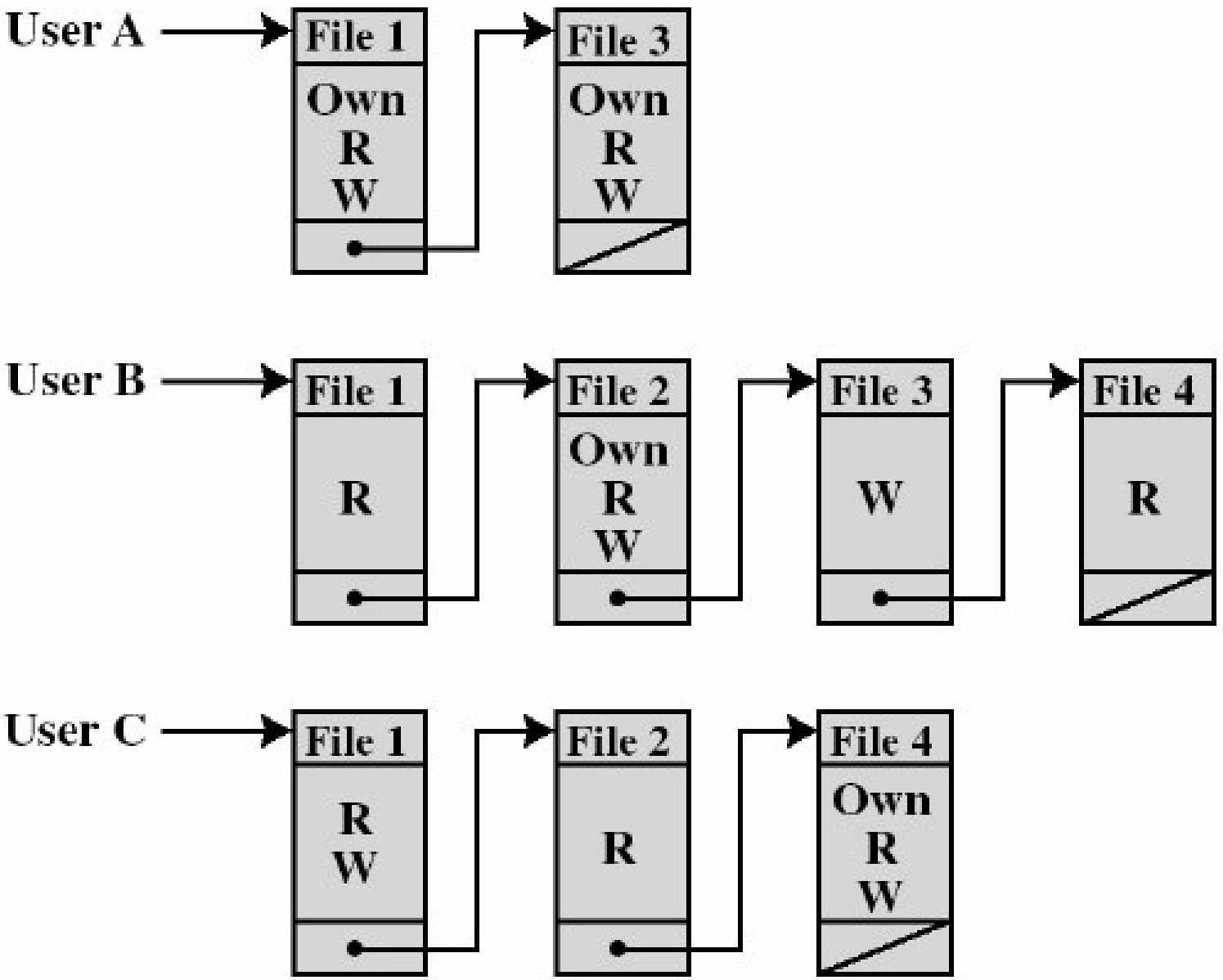
9.4 - Kontrola pristupa podacima

- U realnim sistemima, međutim, matrice za kontrolu pristupa nisu praktične, jer su obično raštrkane, postoji značajna redundansa, novi subjekti i objekti se mogu lako dodavati ili uklanjati, pa centralizovana matrica može postati usko grlo (***bootleneck***).
- Matrica može da se razloži po kolonama, formirajući **Access Control List - ACL** koja nudi detalje u vezi prava pristupa za korisnike.
- **ACL** može sadržati podrazumevane ili javne unose.
- Dekompozicija po redovima nudi **karte mogućnosti**
- Ove karte sadrže ovlašćenja korisnika nad objektima i operacijama.
- Svaki korisnik ima više karata i može ih pozajmiti ili dati drugima.
- Pošto se ove karte mogu raširiti po sistemu, one predstavljaju veći sigurnosni problem u odnosu na ACL.
- OS zadržava sve karte na ime korisnika i to u delovima memorije koji su nedostupni korisnicima kako bi rešio ovaj problem.
- Većina procesora podržava najmanje dva režima rada: korisnički i kernel mod, što omogućava zaštitu ključnih podataka (tabele kakva je blok za kontrolu procesa) od mešanja korisničkih programa

9.4 - Access Control List



9.4 – Karte mogućnosti

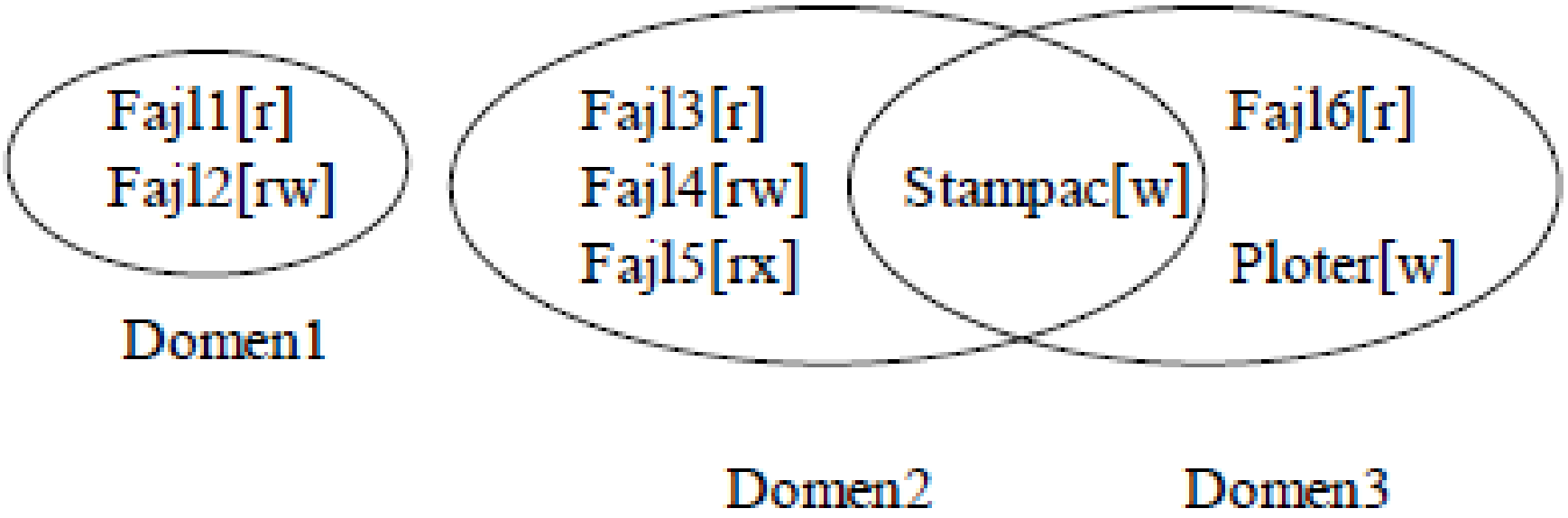


9.5 - Domeni zaštite OS

- OS upravlja raznim objektima koji mogu biti hardverski i softverski.
- Svaki objekat ima jedinstveno ime i može mu se pristupati preko precizno definisanog skup operacija.
- Zaštita se odnosi na kontrolu pristupa programa, procesa i korisnika
- Objektima mogu pristupati samo oni korisnici koji na to imaju pravo, to jest koji su ovlašćeni, i nad objektom mogu izvršiti samo operacije koje pripadaju dozvoljenom skupu operacija.
- Svaki domen definiše skup objekata i sve operacije koje se mogu obaviti nad tim objektom.
- Mogućnost da se izvrši operacija nad objektom zovemo pravo pristupa
- Domen je kolekcija prava pristupa koja su definisana parovima.
- Alokacija procesa u domene može biti statička ili dinamička, a sam domen može da se realizuje na različite načine:
    1. svaki korisnik može biti domen,
    2. svaki proces može biti domen
    3. svaka procedura može biti domen.
- Sistem koji ima dva režima rada ima 2 domena: **korisnički i sistemski**.

9.5 - Domen kao skup parova

- Svakom objektu se dodele tri slova koja označavaju prava pristupa.
- Na primer (**objekat, r w x**).
- Domen je skup parova (objekat,prava).
- Kada proces počne da se izvršava dodeli mu se domen i tada on može da pristupa samo objektima iz tog domena i to kako su prioriteti zadati.
- Oni koji dele domene sa vremena na vreme mogu menjati domen.



9.6 – Matrice pristupa

- Zaštita se može prikazati kao matrica pristupa u kojoj vrste predstavljaju domene, a kolone predstavljaju objekte.
- Element matrice predstavlja skup operacija koje proces iz domena **Di** može da izvrši nad objektom **Oj**.
- Ako je potrebno promeniti sadržaj matrice pristupa, dodeliti ili oduzeti pravo nad određenim objektom, uvode se sledeće operacije:

1. **Operacija *Copy*** - kopira se pravo nad objektom, odnosno procesima iz drugog domena daje se neko pravo pristupa tom objektu. Zvezdicom(*) označavamo pravo kopiranja, što znači mogućnost da proces iz odgovarajućeg domena kopira pravo u drugi domen, to jest u drugo polje iste kolone. Postoje tri vrste kopiranja:

1. ***Kopiranje prava*** - proces u drugom domenu dobija kopiju prava i kopiju prava kopiranja, dato pravo se ne oduzima matičnom procesu
2. ***Prenošenje prava*** - proces u drugom domenu dobija kopiju prava i kopiju prava kopiranja, kopirano pravo se oduzima matičnom procesu
3. ***Ograničeno kopiranje*** - proces u drugom domenu dobija kopiju prava, ali ne dobija pravo kopiranja.

9.6 – Matrice pristupa

2. **Pravo vlasništva** - u matricu je potrebno uvesti mehanizam koji omogućava dodavanje novih prava ili ukidanje postojećih. Ove operacije nad objektom mogu izvesti procesi iz domena koji ima pravo vlasništva nad tim objektom (owner).

3. **Pravo upravljanja u domenu** - operacije kopiranja, dodele i oduzimanja prava modifikuju sadržaj određene kolone u matrici. U matricu se uvodi i pravo upravljanja u domenu kojim je omogućena promena prava po vrsti. Pravo upravljanja se može dodeliti samo objektima koji predstavljaju domene.

	Objekat			
Domen	Datoteka F_1	Datoteka F_2	Datoteka F_3	Štampač
D1	read		read	
D2				print
D3		read	execute	
D4	read, write		read, write	

9.6 – Domen predstavljen matricom

- Matrica pristupa reguliše kako procesi koji pripadaju različitim domenima pristupaju objektima u sistemu.
- Po ovako definisanoj matrici, procesi mogu preći iz jednog domena zaštite u drugi i tako ostvariti veća prava nad objektom, zbog čega se uvodi kontrola prelaska procesa iz jednog domena zaštite u drugi.
- Prebacivanje procesa iz jednog domena u drugi predstavljamo operacijom ***switch***.
- Matrica se proširuje kolonama koje predstavljaju domene kako bi se definisale moguće operacije prebacivanja iz jednog domena u drugi.

	Fajl 1	Fajl 2	Fajl 4	Fajl 4	Fajl 5	Stampac	Fajl 6	Ploter	D1	D2	D3
Dom 1	R	RW								Enter	
Dom 2			R	RW	RX	W					
Dom 3						W	R	W			

9.6 – Matrice pristupa

- Matrica pristupa može se na sistemu implementirati na četiri načina, u zavisnosti od skupa domena/objekata koji su u matrici:

1. ***Globalna tabela***, prvi i najprostiji slučaj realizacije matrice pristupa pomoću globalne tabele koja se sastoji od skupa (domen, objekat, skup prava),
2. ***Listu za kontrolu pristupa objektima***, uz pomoć liste za kontrolu pristupa (***access list***) formira se matrica pristupa.
3. ***Lista mogućnosti domena -*** ovaj način implementacije matrice pristupa predstavlja korišćenje liste mogućnosti domena. Lista mogućnosti (***capability list***) formira se za svaki domen i odgovara jednoj vrsti matrice prava pristupa.
4. ***Mehanizam ključeva,*** predstavlja kompromis između predhodna dva načina implementacije matrice pristupa. Svakom objektu se dodeli lista bravica (*lock*), a svakom domenu lista ključeva (*key*).

9.6 – Matrice pristupa

- U **UNIX**-u su svakom fajlu pridruženi **9 bitova** kojima se određuju prioriteti.
- Prva tri se odnose na vlasnika, druga tri na grupu kojoj vlasnik pripada, a poslednja tri na sve ostale korisnike.
- **r** bit označava pravo čitanja, **w** bit označava pravo pisanja, a **x** bit označava pravo izvršavanja fajla.

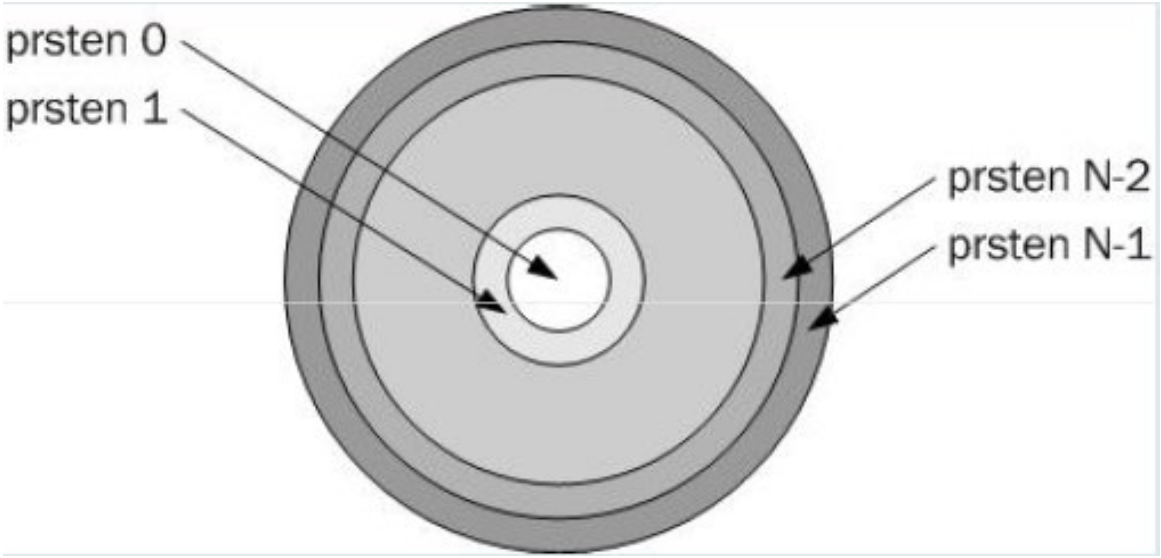
rwx	rwx	rwx
vlasnik	grupa	ostali

Primer: podatak **111101100** koji je pridružen nekom fajlu znači da:

- vlasnik može da čita i piše u fajl i da ga izvršava (**111**)
- grupa kojoj vlasnik pripada može da čita i izvršava fajl (**101**)
- svi ostali mogu samo da čitaju fajl. (**100**)

9.6 – Prstenasta zaštita OS

- U MULITCS sistemima, domeni zaštite su organizovani hijerarhijski u kružne *prstenove*.
- Svaki prsten predstavlja jedan domen.
- **D0** je najprivilegovaniji domen, to je režim rada jezgra.
- Prava iz višeg domena uključena su u skup prava nižih domena, dok obrnuto ne važi.
- Kada neka procedura iz većeg (spoljašnijeg) prstena želi da pozove neki servis ili proceduru iz nekog manjeg (unutrašnijeg) prstena to će uraditi pomoću poziva koji je sličan sistemskim pozivima.



9.7 – Rangovi sigurnosti

- Nacionalni centar za sigurnost računara (*National Computer Security Center*) kako bi pomogao pri zaštiti svojine i ličnih podataka u računarskim sistemima vlade, korporacija i kućnih korisnika definisao je nekoliko rangova, odnosno nivoa sigurnosti koji su:
  1. A1 - Verified Design (proverena arhitektura),
  2. B3 - Security Domains (domeni sigurnosti),
  3. B2 - Structured Protection (struktuirana zaštita),
  4. B1 - Labeled Security Protection (označena sigurnosna zaštita),
  5. **C2** - Controlled Access Protection(zaštita kontrolisanim pristupom),
  6. C1 - Discretionary Access Protection (diskreciona zaštita pristupa),
  7. D - Minimal Protection (minimalna zaštita).

Ključni zahteve koje OS mora da ispuni kako bi dobio rang C2:

- ✓procedura sigurnog prijavljivanja na sistem – jedinstvena identifikacija
- ✓diskreciona kontrola pristupa – vlasnik određuje prava na svom resursu
- ✓praćenje sigurnosnih događaja – njihovo otkrivanje i snimanje
- ✓zaštita od ponovne upotrebe objekata koja sprečava korisnike da vide podatke koje je drugi korisnik već obrisao ili ne dozvoljava pristup memoriji koju je drugi korisnik upotrebio i oslobodio.

Hvala na pažnji !!!

Pitanja

? ? ?