

# Informaciona bezbednost

## Kontrola pristupa

Goran Sladić

Katedra za informatiku

2025.



Fakultet tehničkih nauka  
Univerzitet u Novom Sadu

# Identifikacija ≠ autentifikacija ≠ autorizacija

- Identifikacija je proces pripisivanja ID-a čoveku ili drugom računaru ili mrežnoj komponenti
- Autentifikacija je proces provere identiteta
- Autorizacija je utvrđivanje prava koja korisnik ima nad resursima u sistemu

# Identifikacija ≠ autentifikacija ≠ autorizacija

- Autorizacija zahteva uspešnu autentifikaciju
  - Autentifikacija prethodi autorizaciji
- Autentifikacija podrazumeva identifikaciju
  - Identifikacija je sastavni deo postupka autentifikacije

# Kontrola pristupa

- "Ko može da uradi šta"
- Bezbednosni mehanizam koji je prisutan u svim delovima informacionih sistema
- Prvi bezbedan sistem za računanje - registar kasa (Dayton, Ohio, 1879)
  - Kupac vidi iznos koji je prodavac otkucao
  - Fioka se otvara samo prilikom unosa iznosa
  - Kasa zapisuje istoriju naplata

# Rizici po bezbednost informacija

- CIA klasifikacija
  - Confidentiality (poverljivost)
    - Čuvanje podataka od neovlašćenog čitanja
  - Integrity (integritet)
    - Čuvanje podataka od izmena
  - Availability (dostupnost)
    - Informacije su dostupne u trenutku kada su i potrebne
- Mehanizmi kontrole pristupa bave se poverljivošću i integritetom
  - Onaj ko neovlašćeno pristupi nekom sistemu može da utiče i na dostupnost

# Razvoj mehanizama kontrole pristupa

- Prvi radovi početkom 1970-tih
- Standardizacija početkom 1980-tih
- Role-based Access Control (RBAC) početkom 1990-tih

# Koncepti kontrole pristupa

- Korisnik (*user*)

- Čovek koji koristi informacioni sistem
- Ima svoj identifikator
- Može imati više identifikatora
- Sistem može povezati više identifikatora sa istim korisnikom
- Sesija je jedna instanca komunikacije korisnika sa sistemom

# Koncepti kontrole pristupa

- Subjekat (*subject*)

- Računarski proces (~ program) koji obavlja zadatke za korisnika
- Jedan korisnik, sa istim ID-jem, može imati više subjekata (email klijent, web klijent, ...)
- Kontrola pristupa sprovodi se za svaki subjekat posebno

# Koncepti kontrole pristupa

- **Objekat (*object*)**

- Bilo koji resurs informacionog sistema koji je dostupan korisniku
  - Fajl
  - Štampač
  - Baza podataka
  - Pojedini slogovi u bazi podataka
- Tipično se tretiraju kao pasivni entiteti koji sadrže ili primaju podatke
- Stari modeli kontrole pristupa uključivali su mogućnost da se programi, štampači i drugi aktivni entiteti posmatraju kao objekti

# Koncepti kontrole pristupa

- Operacija (*operation*)
  - Aktivan proces koga je pokrenuo subjekat
- Primer: bankomat
  - Korisnik se autentificuje karticom i PIN-om
  - Program koji opslužuje korisnika je subjekat
  - Subjekat može da pokrene više operacija
    - Upit stanja
    - Isplata
    - Uplata

# Koncepti kontrole pristupa

- Dozvola (*permission*)
  - Dopuštenje da se obavi određena operacija u okviru sistema
  - Kombinuje objekat i operaciju
- Dva objekta i ista operacija → različite dozvole
- Isti objekat i dve operacije → različite dozvole

# Koncepti kontrole pristupa

- Minimalne privilegije (*least privilege*)
  - Selektivno dodeljivanje dozvola korisnicima tako da nemaju više privilegija nego što je minimalno neophodno za obavljanje njihovog posla
- Ako korisnik ima mogućnost da izvrši nepotrebne ili štetne operacije → potencijalni problem
- Određivanje skupa minimalnih privilegija je zadatak administrativne prirode
  - Identifikacija funkcija vezanih za jedno radno mesto ili korisnika
  - Specifikacija dozvola potrebnih za obavljanje svake od funkcija
  - Restrikcija korisnika na neki domen uz dodeljene privilegije
- Striktno pridržavanje ovog principa → korisnik može imati različite dozvole u različitim trenucima
- Skup dozvola se menja tokom vremena (dinamička priroda)

# Elementi kontrole pristupa

- Tri apstrakcije kontrole se mogu razmatrati:
  - **Politike/polise** kontrole pristupa
  - **Mehanizmi** kontrole pristupa
  - **Model** kontrole pristupa

# Politika kontrole pristupa

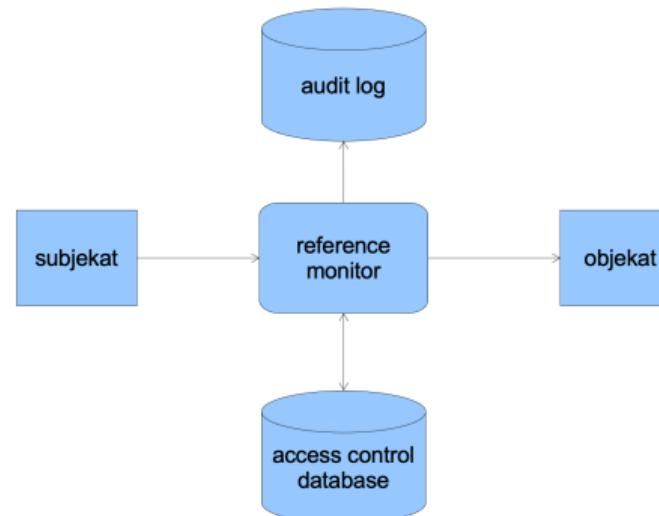
- Politika
  - Zahtevi visokog nivoa kojima se definiše ko može da pristupi čemu i pod kojim uslovima
- Politika se može definisati posebno za različite aplikacije ali često je definisana u okviru realnog sistema (njegove organizacione strukture)
  - Finansijska institucija
  - Vojna institucija
  - Zdravstvena institucija
- Politika se menja tokom vremena, jer odslikava promene u načinu rada organizacije
  - Pri tome ne moraju da se menjaju model ili mehanizmi kontrole pristupa

# Mehanizmi kontrole pristupa

- Sprovode politiku kontrole pristupa
- Zahtevaju da se bezbednosni atributi o korisnicima i resursima čuvaju
  - Atributi korisnika mogu biti ID, grupe, uloge kojima pripada, nivo poverenja koje mu je dato
  - Atributi resursa mogu biti nivoi osjetljivosti, tipovi, pristupne liste
- Mehanizam 1: poređenje vrednosti bezbednosnih atributa
  - Za *read* operaciju - *clearance level* korisnika  $\geq$  *classification level* resursa
- Mehanizam 2: poklapanje u bezbednosnim atributima
  - Za fajl je vezana lista parova (korisnik, pravo)
  - Provera obuhvata pretragu liste za korisnika koji traži pristup i operaciju koju zahteva

# Mehanizmi kontrole pristupa - reference monitor

- Apstraktni pogled na (pod)sistem za kontrolu pristupa
- Služi kao vodič za dizajn, implementaciju i analizu bezbednih IT sistema
- Predstavlja hardverske i softverske delove sistema koji su odgovorni za primenu bezbednosnih polisa



## Mehanizmi kontrole pristupa - reference monitor

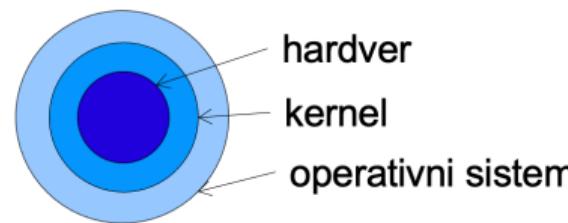
- Zahtevi za implementaciju reference monitora se sastoje iz tri principa:
  - Kompletност (*completeness*) – uvek se mora pozvati i nije ga moguće zaobići
  - Izolacija (*isolation*) – mora biti otporan na neovlašćene izmene (*tampering*)
  - Proverivost (*verifiability*) – njegova korektna implementacija mora biti proveriva/dokaziva

# Mehanizmi kontrole pristupa - reference monitor

- Kompletност
  - Subjekat može da pristupi objektu **isključivo** preko RM-a
- Problem 1: šta su objekti u sistemu?
  - Očigledne stvari: fajlovi, memorija, baferi, ...
  - Manje očigledne stvari: imena fajlova, poruke o greškama, ...
  - Kompletnost zahteva da se zaštite **svi** objekti, ne samo očigledni
- Problem 2: kako sprečiti zaobilaženje RM-a?
  - Kako sprečiti pristup fajlu ako se pristup vrši preko fizičke adrese na disku?
  - Kako SUBP da spreči pristup svojim fajlovima od strane operativnog sistema?

# Mehanizmi kontrole pristupa - reference monitor

- Izolacija
  - Mora biti nemoguće za napadača da pristupi/promeni RM tako da on više ne funkcioniše pravilno
  - Potrebna je podrška i u hardveru i u softveru
- Jedno rešenje: security kernel
  - Minimalna implementacija onih funkcija sistema koje su relevantne za bezbednost
  - Oslanja se na hardver
  - Pruža usluge delovima operativnog sistema na višem nivou
  - Koristi se i za razdvajanje koda i podataka operativnog sistema od aplikacija
  - Kernel softver je takođe podložan greškama u implementaciji



# Mehанизми контроле приступа - reference monitor

- Proverivost
  - Ispravnost security kernela je potrebno proveriti
- Testiranje security kernela može biti veoma komplikovano
- Napraviti kernel što manjim
  - Isključiti sve funkcije koje nisu potrebne za bezbednost sistema
  - Definisati mali i jednostavan skup interfejsa
- Olakšati testiranje kernela dobrom implementacijom
  - Apstrakcija
  - Skrivanje informacija
  - Modularnost
- Formalno modeliranje kernela, formalne metode provere korektnosti

# Mehanizmi kontrole pristupa - reference monitor

- RM je potreban uslov za sprovođenje kontrole pristupa
- RM nije dovoljan uslov
  - Najčešće se RM kupuje u okviru nekog većeg sistema
- Kako svoju politiku implementirati na kupljenom RM-u?
- Tri dodatna uslova za sistem za kontrolu pristupa
  - Fleksibilnost – sistem mora da podrži politiku kontrole pristupa u organizaciji
  - Upravljivost – sistem mora biti jednostavan za korišćenje i upravljanje
  - Skalabilnost – funkcije sistema moraju raditi na isti način i za realan (veliki) broj korisnika i resursa u realnom (velikom) sistemu

# Modeli kontrole pristupa

- Zasnivaju se na konceptima kontrole pristupa
- Predstavljaju apstraktni pogled na mehanizme za sprovođenje kontrole pristupa
  - Jednostavnija analiza
  - Mogućnost izbora različitih implementacija
- Različiti modeli kontrole pristupa postoje
  - Lampson
  - Bell-LaPadula
  - US DoD standardi
    - Discretionary Access Control (DAC)
    - Mandatory Access Control (MAC)
  - Clark-Wilson
  - Role-based Access Control (RBAC)

# Lampson model

- „Matrica pristupa“
  - Jedan red po subjektu
  - Jedna kolona po objektu
- Koristi koncepte subjekta i objekta
  - Subjekti: procesi koje je pokrenuo korisnik
  - Objekti: resursi sistema (fajlovi, ...)
  - Operacije: operacije nad resursima (čitanje, pisanje)
- Elementi matrice su skupovi dozvoljenih operacija

# Lampson model

- „Matrica pristupa“

<del>Objekat</del> <del>Subjekat</del>	Fajl_1	Fajl_2	Fajl_3	Proces_1
Pera	Read, Write	-	Write	-
Mika	-	Execute	-	Suspend
Žika	-	Read	Read	-
Gaja	Read	-	-	-

# Lampson model

- U realnim sistemima matrice su velike (puno korisnika, puno objekata)
- I retko popunjene (*sparse*)
- Dva uobičajena mehanizma za implementaciju:
  - Liste sposobnosti (*capability lists*)
    - Za svaki subjekat čuva se lista njegovih „sposobnosti“ (objekat, pravo)
    - Kako dobiti sve subjekte koji mogu da pristupe određenom objektu? Samo prolazom kroz sve liste
  - Liste kontrole pristupa (*access control lists, ACLs*)
    - Za svaki objekat čuva čuva se lista parova (subjekat, pravo)
    - Ne mora biti puno ACL listi u sistemu ako se korisnici raspodele u grupe

# Lampson model

- Capability List

Subjekat		
Pera	Fajl_1: Read, Write	Fajl_3: Write
Mika	Fajl_2: Execute	Proces_1: Suspend
Žika	Fajl_2: Read	Fajl_3: Read
Gaja	Fajl_1: Read	

- Access Control List

Objekat		
Fajl_1	Pera: Read, Write	Gaja: Read
Fajl_2	Mika: Execute	Žika: Read
Fajl_3	Pera: Write	Žika: Read
Proces_1	Mika: Suspend	

## Bell-LaPadula model

- Formalizacija vojnih pravila za kontrolu pristupa
- Objekti = dokumenti
- Objekti imaju svoj nivo poverljivosti (classification level)
  - Poverljivo
  - Strogo poverljivo
  - Državna tajna
- Korisnici imaju svoj nivo pristupa (clearance level)

# Bell-LaPadula model

- Osnovna pravila:
  - ① *No read up* - subjekat ima pristupa samo onim dokumentima čiji nivo je manji ili jednak njegovom
  - ② *No write down* - subjekat može da piše samo u dokumente čiji nivo je veći ili jednak njegovom (ali ne može da čita)
- Dodatni koncept: kategorija
  - Svaki dokument spada u jednu ili više kategorija
  - Korisnik mora imati odgovarajući nivo pristupa za svaku kategoriju

# Bell-LaPadula model

- Problem ovog modela:

- Sistem nije odlučiv (*undecidable*) - ne može se znati da li će konfiguracija za koju se smatra da je ispravna ostati ispravna
- Harrison, Ruzzo, Ullman 1976. - formalan dokaz
- Korisnici mogu (čak i bez namere) dodeliti prava pristupa kroz mehanizme za delegiranje prava

# TCSEC standard

- Standardizacija modela kontrole pristupa u okviru US Department of Defense
- *Trusted Computer System Evaluation Criteria* („Orange Book“) 1983.
- Definiše dva modela kontrole pristupa
  - Discretionary Access Control (DAC)
    - Vlasnici objekata dodeljuju prava pristupa
    - Subjekti imaju posebno pravo delegiranja sopstvenog prava drugim subjektima
    - Model nije odlučiv
  - Mandatory Access Control (MAC)
    - Model jeste odlučiv jer subjekti ne mogu da daju prava drugima
    - Višenivojski model, na osnovu Bell-LaPadula

# Discretionary Access Control

- Ograničavanje pristupa objektima na osnovu identiteta korisnika ili grupe korisnika
- „Diskrecija“ - korisnik koji ima odgovarajuće pravo može delegirati pravo pristupa objektu drugim korisnicima
- Uvodi koncept „vlasništva“ nad objektom
  - Vlasnik objekta ima pravo da dodeljuje prava pristupa drugim korisnicima
- Najčešći mehanizam za implementaciju DAC modela su ACLs

# Discretionary Access Control

- Dve osnovne slabosti:

- Dodeljivanje prava čitanja je tranzitivno
  - Alice dozvoli Bobu da čita određeni fajl (Alice je vlasnik fajla)
  - Bob iskopira sadržaj fajla u svoj fajl (Bob je vlasnik novog fajla)
  - Bob dozvoli Carol da čita novi fajl
  - Alice ne zna da Carol ima pristup podacima iz njenog fajla!
- Ranjivost na napade trojanskim konjem
  - Programi nasleđuju identitet korisnika koji ih je pokrenuo
  - Bob napiše program za Alice koji će sadržaj Alicinog fajla iskopirati na neko mesto koje je dostupno i Bobu i Alice
  - Alice će pokrenuti program, ne znajući šta sve on radi
  - Bob će dobijeni fajl skloniti na svoje privatno mesto
  - Bobov program može čak i obrisati Alicine fajlove
  - U audit logu piše da je Alice pokrenula program koji je obrisao fajlove

# Discretionary Access Control

- Primer implementacije - *protection bits*:
  - Kontrola pristupa fajl-sistemu na UNIX/Linux operativnim sistemima
  - Svaki objekat u fajl-sistemu ima dodeljen atribut koji definiše pravila za kontrolu pristupa
- Tri kategorije korisnika:
  - *Self* – vlasnik fajla
  - *Group* – kolekcija korisnika koji dele zajednički pristup fajlu
  - *Other* – svi ostali osim vlasnika ili članova grupe

# Discretionary Access Control

- Primer implementacije - *protection bits*:
  - Kontrola pristupa fajl-sistemu na UNIX/Linux operativnim sistemima
  - Svaki objekat u fajl-sistemu ima dodeljen atribut koji definiše pravila za kontrolu pristupa
- Za svaku kategoriju definisana su po tri bita
  - r - dozvola čitanja
  - w - dozvola pisanja
  - x - dozvola izvršavanja (za direktorijume: dozvola listanja)

# Discretionary Access Control

- Npr. (r w x)(r - x)(-- x)
  - Vlasnik ima prava čitanja, pisanja i izvršavanja
  - Članovi grupe imaju prava čitanja i izvršavanja
  - Ostali korisnici imaju prava izvršavanja
- Vlasnik i super korisnik mogu da modifikuju zaštitne bite nad fajlom
- Samo jedna grupa je dostupna za fajl i administrator upravlja članstvom
- Zaštitni biti ne odgovaraju u potpunosti "matrici pristupa" zbog čega sistem ne može precizno da da pristup objektu na individualnom nivou
- Zbog ovoga mnoge novije verzije UNIX-like operativnog sistema uključuju ACL mehanizam

# Discretionary Access Control

- Primer: relacione baze podataka
- Objekti su:
  - Tabele
  - Pogledi
- Objekat ima vlasnika
- Operacije su:
  - SELECT
  - INSERT
  - UPDATE
  - DELETE

# Discretionary Access Control

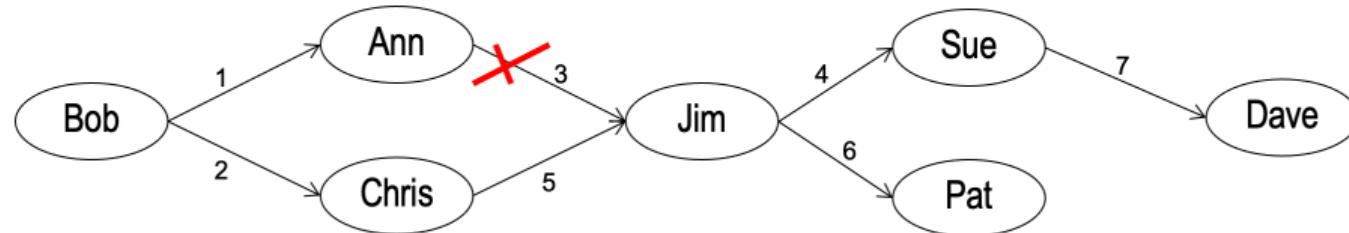
- Kako organizovati administraciju?
- Centralizovana administracija
  - Samo određeni privilegovani korisnici mogu da daju i oduzimaju prava pristupa
- „Vlasnička“ administracija
  - Davanje i oduzimanje prava pristupa može da izvrši samo vlasnik objekta
  - Delegiranje vlasničkih prava – vlasnik objekta može i drugim korisnicima dodeliti pravo da daju i oduzimaju prava pristupa
  - GRANT Select ON Employee TO Tim **WITH GRANT OPTION;**

# Discretionary Access Control

- SQL GRANT naredba: dodeljivanje prava pristupa
  - (Bob): GRANT Select ON Employee TO Ann **WITH GRANT OPTION**;
  - (Ann): GRANT Select ON Employee TO Jim;
  - (Bob): GRANT Update, Insert ON Employee TO Jim;
- SQL REVOKE naredba: uklanjanje prava pristupa
  - (Bob): REVOKE Select ON Employee TO Jim;

# Discretionary Access Control

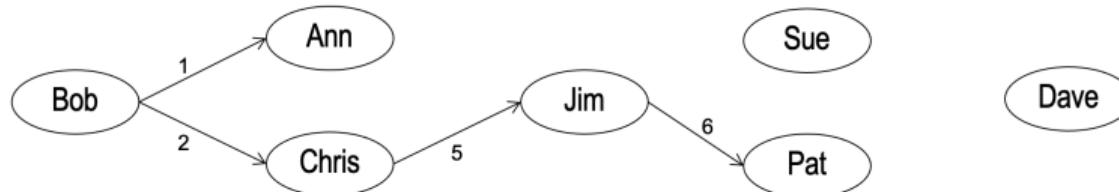
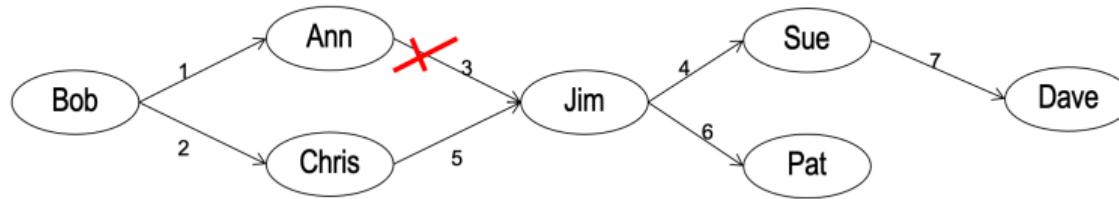
- Ukidanje prava može biti
  - Kaskadno
  - Kaskadno bez vremenske odrednice
  - Nekaskadno
- Primer:
  - Ann ukida pravo koje je dodelila Jimu
  - (brojevi predstavljaju hronologiju dodeljivanja prava)



# Discretionary Access Control

- Kaskadno ukidanje prava

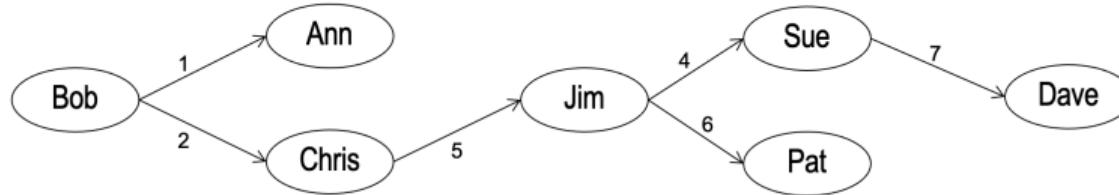
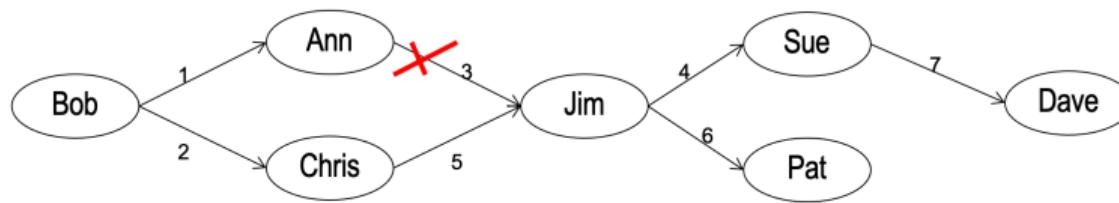
- Prava se ukidaju tako da novo stanje odgovara situaciji kada se sprovede ista sekvenca dodela izuzimajući ukinutu
- Mora se voditi računa o redosledu dodeljivanja prava, odnosno o trenutku (timestamp) kada je neko pravo dodeljeno



# Discretionary Access Control

- Kaskadno bez vremenske odrednice

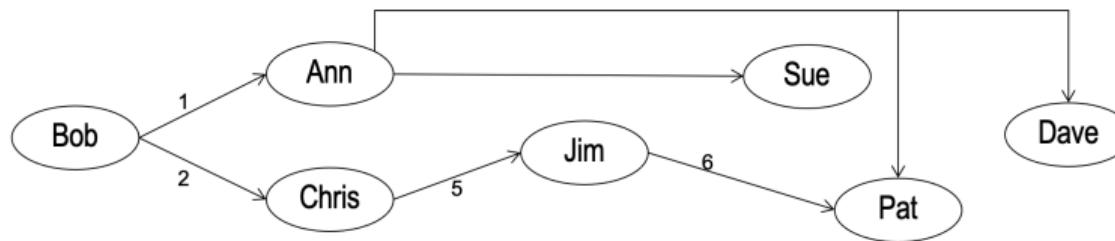
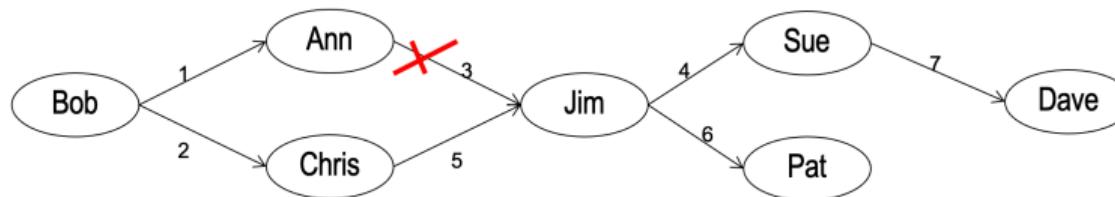
- Prava se ukidaju tako da se ne vodi računa o trenutku kada je pravo i dodeljeno



# Discretionary Access Control

- Nekaskadno

- Uklanjanje prava ne sme utiče na prava koja je dodelio korisnik koji ih je upravo izgubio (Jim)



# Mandatory Access Control

- Baziran na Bell-LaPadula modelu
- Bezbednosni nivoi se dodeljuju korisnicima i objektima gde subjekti u ime korisnika rade akcije
- Bezbednosni atributi korisnika i objekata imaju
  - Higerarhijsku komponentu - bezbednosni nivo (clearance level)
    - Unclassified (U)
    - Confidential (C)
    - Secret (S)
    - Top secret (TS)
  - Nehigerarhijsku komponentu - dodeljene kategorije, npr.
    - NATO
    - NUCLEAR
    - ...

# Mandatory Access Control

- Primer:
  - $TS \geq S \geq C \geq U$
  - $S(NATO, NUCLEAR) \geq S(NUCLEAR) \geq S$
- Pravila *no read up i no write down*
- Korisnik sa atributima  $S(NUCLEAR)$  može da pristupa objektima sa atributima  $S(NUCLEAR), S, C, U$

# Mandatory Access Control

- Model je odlučiv
- Važno je razlikovati korisnika i subjekta (program)
  - Alice ima TS nivo: trebalo bi da može da čita i piše sve dokumente
  - Alice (korisnik) neće odavati TS informacije na nižim nivoima, ali možda program koji ona koristi (subjekat) hoće
  - Alice mora da promeni ("spusti") svoju sesiju na S nivo da bi pisala u fajlove na S nivou
- *No write down* pravilo obezbeđuje od napada trojanskim konjem
  - Korisnik ne može da piše u objekat koji je dostupan korisnicima sa nižim nivoima od njegovog
  - Alice ima nivo S(NUCLEAR)
  - Bob ima nivo S
  - Bob podmeće trojanskog konja Alice
  - Program će moći da čita podatke sa nivoa S(NUCLEAR) kada ga pokrene Alice
  - Ali neće moći da ih piše u fajl nivoa S (koga Bob može da čita)
  - Zaštita od brisanja trojanskim konjem nije rešena
  - Alice može Bobovim trojanskim konjem da pobriše sve svoje fajlove

# Mandatory Access Control

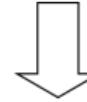
- Primer: baze podataka
- Korisnici X, Y i Z sa nivoima poverljivosti:
  - $\text{clearance}(X) = \text{TS}$
  - $\text{clearance}(Y) = \text{S}$
  - $\text{clearance}(Z) = \text{U}$
- Podaci u tabeli baze podataka su sledeći

Project Name	Topic	Location	
Black, TS	Databases, TS	Los Angeles, TS	
Silver, S	Supply Chain, S	New York, S	
Gold, U	Inventories, S	Atlanta, S	
Indigo, U	Telecommunication, U	Austin, U	

# Mandatory Access Control

- Primer: podacima pristupa Y, clearance(Y) = S

Project Name	Topic	Location
Black, TS	Databases, TS	Los Angeles, TS
Silver, S	Supply Chain, S	New York, S
Gold, U	Inventories, S	Atlanta, S
Indigo, U	Telecommunication, U	Austin, U

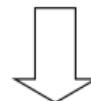


Project Name	Topic	Location
Silver, S	Supply Chain, S	New York, S
Gold, U	Inventories, S	Atlanta, S
Indigo, U	Telecommunication, U	Austin, U

# Mandatory Access Control

- Primer: podacima pristupa Z, clearance(Z) = U

Project Name	Topic	Location	
Black, TS	Databases, TS	Los Angeles, TS	
Silver, S	Supply Chain, S	New York, S	
Gold, U	Inventories, S	Atlanta, S	
Indigo, U	Telecommunication, U	Austin, U	



Project Name	Topic	Location	
Gold, U	-, U	-, U	
Indigo, U	Telecommunication, U	Austin, U	

# Mandatory Access Control

- Primer: Z hoće da doda novi red (Silver, Linear Programming, Omaha)

Project Name	Topic	Location	
Black, TS	Databases, TS	Los Angeles, TS	
Silver, S	Supply Chain, S	New York, S	
Gold, U	Inventories, S	Atlanta, S	
Indigo, U	Telecommunication, U	Austin, U	
Silver, U	Linear Programming, U	Omaha, U	

- Problem: ponavljanje podataka sa istim ključem!

# Mandatory Access Control

- Primer: Z hoće da zameni NULL vrednosti konkretnim podacima (Markov Chain, New Jersey)

Project Name	Topic	Location	
Gold, U	-, U	-, U	
Indigo, U	Telecommunication, U	Austin, U	



Project Name	Topic	Location	
Black, TS	Databases, TS	Los Angeles, TS	
Silver, S	Supply Chain, S	New York, S	
Gold, U	Inventories, S	Atlanta, S	
Indigo, U	Telecommunication, U	Austin, U	
Gold, U	Markov Chain, U	New Jersey, U	

# Biba model

- MAC model se fokusira na poverljivost podataka, a zapostavlja integritet
- Biba model (1977) fokusira se na integritet a zapostavlja poverljivost
- Uvodi read-write ograničenja bazirana na nivoima integriteta (sa hijerarhijskom i kategorizacijskom komponentom)
  - Za korisnika: indikacija nivoa poverenja u korisnika u pogledu menjanja podataka na datom nivou
  - Za objekat: osetljivost objekta na izmene
- Primer nivoa:
  - Critical (C)
  - Important (I)
  - Ordinary (O)
- Pravila za kontrolu pristupa su inverzna u odnosu na Bell-LaPadula model
  - Subjekat S može da čita objekat O ako je  $\text{clearance}(S) \leq \text{classification}(O)$
  - Subjekat S može da piše u objekat O ako je  $\text{clearance}(S) \geq \text{classification}(O)$

# Clark-Wilson model

- Clark-Wilson 1987: u komercijalnim (ne-vojnim) primenama daleko je važniji integritet nego poverljivost
  - Integritet: podaci se menjaju samo na ispravan način od strane autorizovanih korisnika
- Dva centralna koncepta modela:
  - Dobro formirana transakcija (*well-formed transaction, WFT*)
    - Sistem ograničava korisnika na promene podataka samo pomoću odgovarajućih transakcija
    - Podaci iz jednog validnog stanja mogu preći u drugo validno stanje
    - Npr. bankarski službenik ne može da modifikuje proizvajni deo podataka o klijentu već samo onaj deo koji je deo transakcije (kao što je skidanje novca sa računa ili uplata)
  - Razdvajanje zaduženja (*separation of duty, SoD*)
    - Osigurava konzistentnost izmena u podacima
    - Npr. nabavku zahteva korisnik A, odobrava korisnik B, kontroliše (nadgleda) korisnik C

# Clark-Wilson model

- Osnovna jedinica kontrole pristupa je uređena trojka
  - Korisnik (*user*)
  - Transformaciona procedura (*transformation procedure*, TP)
  - Podatak sa ograničenim pristupom (*constrained data item*, CDI)
- Pored toga, postoji i:
  - Podatak bez ograničenja pristupa (*unconstrained data item*, UDI)
  - Procedura za proveru integriteta (*integrity verification procedure*, IVP) - utvrđuje da li je podatak u validnom stanju

# Clark-Wilson model

- Devet pravila za obezbeđivanje integriteta podataka
  - 1 Za svaki CDI mora postojati IVP koja proverava da li je CDI u validnom stanju
  - 2 Svaka TP koja menja CDI mora biti sertifikovana da ga menja isključivo na validan način
  - 3 CDI može da menja samo sertifikovana TP
  - 4 Svaka TP mora da vodi dnevnik promena koje sprovodi nad CDI
  - 5 Svaka TP koja kao ulaz ima UDI mora da transformiše UDI u CDI isključivo na validan način
  - 6 Samo sertifikovane TP mogu da menjaju UDI
  - 7 Korisnik može da pristupi CDI samo kroz TP za koju je autorizovan
  - 8 Svaki korisnik mora biti autentifikovan pre pozivanja TP
  - 9 Samo bezbednosni administrator može da autorizuje korisnika da poziva TP
- TP - transformaciona procedura (*transformation procedure*)
- CDI - podatak sa ograničenim pristupom (*constrained data item*)
- UDI - podatak bez ograničenja pristupa (*unconstrained data item*)
- IVP - procedura za proveru integriteta (*integrity verification procedure*)

## Clark-Wilson model

- Bell-LaPadula model kontroliše tok podataka pomoću kontrole operacija čitanja i pisanja niskog nivoa
- Clark-Wilson model teži da se podaci menjaju samo na autorizovan način od strane autorizovanih korisnika
  - Ovo se ne može implementirati samo na nivou kernela
  - Primer: baza podataka gde tabele nisu neposredno dostupne korisnicima, nego samo uskladištene procedure

## Clark-Wilson model

- Razdvajanje zaduženja (SoD) - način da sprečimo da autorizovani korisnici naprave pogrešne izmene u podacima
- Npr. kupovina robe
  - Formiranje i slanje porudžbine (korisnik A)
  - Evidentiranje prispeća robe (korisnik B)
  - Odobravanje plaćanja (korisnik C)
- Ako bi sve ove podoperacije radila ista osoba moguće su prevare
- Ako ove podoperacije rade različite osobe, moguća je prevara, uz „zločinačko udruživanje“

## Politika kineskog zida

- Osnovna namera: sprečiti tok podataka koji mogu izazvati konflikt interesa
- Primer:
  - Finansijski konsultanti dobijaju privatne podatke svojih klijenata (banaka)
  - Ako konsultant poznaje privatne podatke ovih banaka može to da zloupotrebi (za privatni profit, za dobrobit jedne banke a na štetu druge)
- Privatni podaci o organizaciji se nalaze u jednoj od (međusobno disjunktnih) kategorija za konflikt interesa (COI)
- Svaka organizacija pripada u tačno jednu COI
- Svaki COI sadrži bar dve organizacije, koje se bave istom ili sličnom delatnošću
- Konsultant ne može da pristupi podacima više od jedne organizacije iz istog COI
  - Ako pristupi privatnim podacima jedne organizacije iz datog COI, ne može da pristupa podacima drugih organizacija iz istog COI
- Rešenje za kontrolu operacije čitanja, ne i pisanja

# Brewer-Nash model

- Svaka organizacija je predstavljena skupom podataka
- Skupovi podataka su smešteni u COI
- Pravilo za čitanje podataka: subjekat S ima pravo da čita objekat O ako je zadovoljeno nešto od sledećeg
  - O je u istom skupu podataka kao i neki drugi objekat koga je S već čitao
  - O pripada COI iz koga S još nije čitao ništa
- Pravilo za pisanje podataka: subjekat S ima pravo da piše u objekat O ako je zadovoljeno sve od sledećeg
  - S može da čita O prema pravilu za čitanje
  - Nijedan objekat iz drugih skupova u odnosu na skup kome pripada O nije dostupan za čitanje

# Brewer-Nash model

- Pravilo za pisanje je zamišljeno kao zaštita od trojanskih konja
- Alice ima pravo
  - Čitanja podataka energetske kompanije A i
  - Čitanja i pisanja podataka banke A
- Bob ima pravo
  - Čitanja podataka energetske kompanije B i
  - Čitanja podataka banke B
- Trojanski konj koji je pokrenula Alice (sa svojim privilegijama) bi mogao da pokuša da
  - Čita podatke o banci A i
  - Piše te podatke u banku B (ne može)
  - Piše te podatke u energetsku kompaniju B (ne može)

# Domain Type Enforcement model

- Subjekti = aktivni entiteti (procesi, programi)
- Subjektu se dodeljuje **domen**
- Objekti = pasivni entiteti (fajlovi, uređaji, delovi memorije)
- Objektu se dodeljuje **tip**
- Dozvole se vezuju za domene i tipove
- Domen-domen dozvole
  - Izražene tabelarno: domain-domain access control table (DDACT)
- Domen-tip dozvole
  - Izražene tabelarno: domain-type acces control table (DTACT)
  - U ćelijama tabele nalazi se skup dodeljenih prava

# Domain Type Enforcement model

- Primer: fajl sistem
  - Domen-domen dozvole: create (C) i kill (K)
  - Domen-tip dozvole: read (R), write (W), execute (E), browse directory (T)
  - Proces A može da pokrene proces B samo ako postoji pravo C u ćeliji tabele koja povezuje A i B
- Slično kao i Lampson model (matrica pristupa) ali je matrica znatno manja zbog grupisanja procesa u domene i objekata u tipove