Bezbednost računarskih sistema

Operativni sistemi 2

Institut za matematiku i informatiku

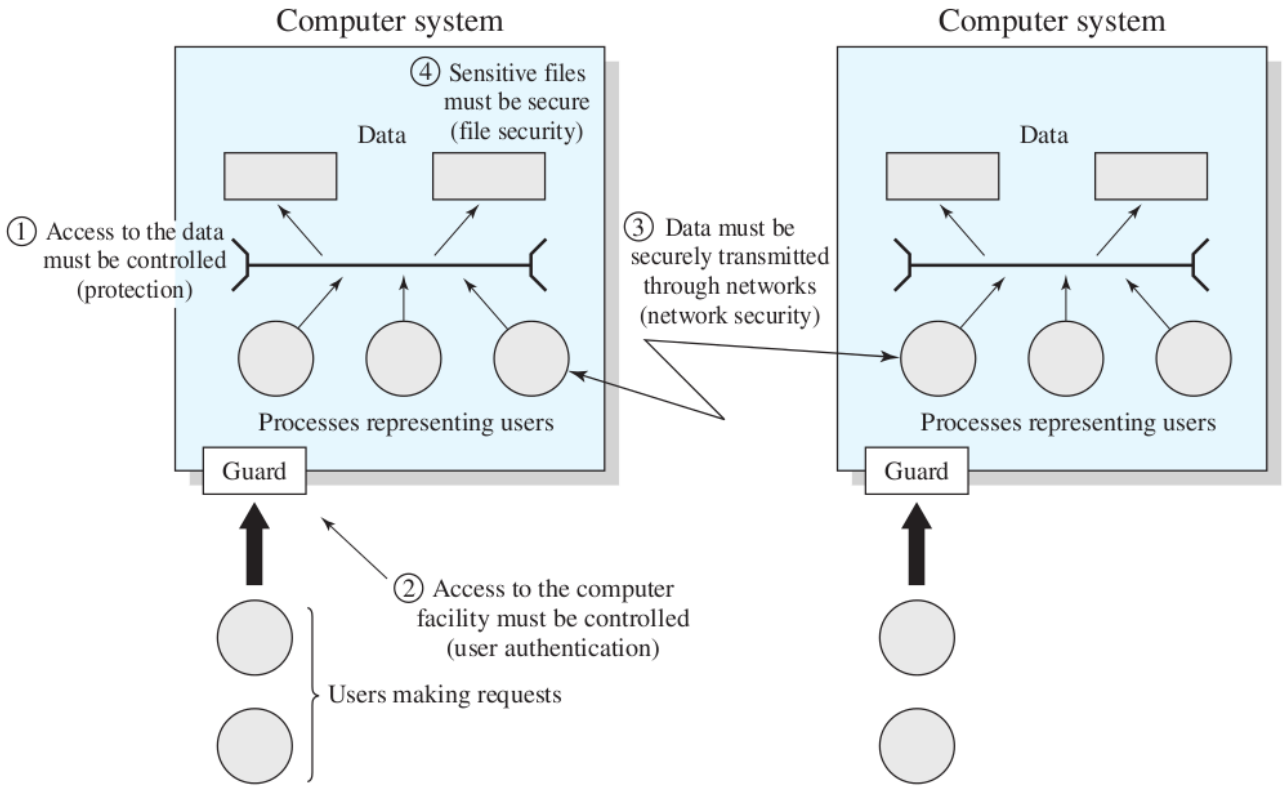
Institut za matematiku i informatiku
Prirodno-matematički fakultet, Kragujevac

Decembar 2013. god.

O čemu će biti reči?

Zahtevi bezbednosti

- Poverljivost
- Integritet
- Raspoloživost
- Autentičnost



Vrste pretnji

1. Prekid - napad na **raspoloživost**
2. Presretanje - napad na **poverljivost**
3. Menjanje - napad na **integritet**
4. Fabrikacija - napad na **autentičnost**

Objekti koji se napadaju

	Availability	Confidentiality	Integrity
Hardware	Equipment is stolen or disabled, thus denying service.		
Software	Programs are deleted, denying access to users.	An unauthorized copy of software is made.	A working program is modified, either to cause it to fail during execution or to cause it to do some unintended task.
Data	Files are deleted, denying access to users.	An unauthorized read of data is performed. An analysis of statistical data reveals underlying data.	Existing files are modified or new files are fabricated.
Communication Lines	Messages are destroyed or deleted. Communication lines or networks are rendered unavailable.	Messages are read. The traffic pattern of messages is observed.	Messages are modified, delayed, reordered, or duplicated. False messages are fabricated.

Napadi na komunikacione linije i mreže

- Pasivni napadi - Teško se otkrivaju jer ne menjaju podatke. Najefikasnija je odbrana šifrovanjem.
  1. Preuzimanje sadržaja poruke
  2. Analiza saobraćaja
- Aktivni napadi - Uključuju bar neku izmenu sadržaja odn. toka podataka.
  1. Maskiranje
  2. Ponavljanje
  3. Izmena poruke
  4. Odbijanje usluge (*Denial of service*)

Zaštita

Šta se deli u multiprogramiranom sistemu?

1. Memorija
2. UI uređaji
3. Programi
4. Podaci

Vrste deljenja objekata

1. Bez zaštite
2. Izolacija
3. Deljenje svega ili deljenje ničega
4. Deljenje putem ograničenog pristupa
5. Dinamičko deljenje
6. Ograničena upotreba objekata

Zaštita memorije

- Razdvajanje memorijskog prostora različitih procesa implementira primenom virtualne memorije
- Straničenje i/ili segmentacija
- Nema dvostrukih unosa u tabele straničenja ili segmentacije!
- IBM zSeries z/OS

Kontrola pristupa orijentisana na korisnika

- Procedure bezbednosti se dele na one orijentisane prema korisnicima i one orijentisane prema podacima
- Uglavnom se koristi korisnički ID / lozinka
- U distribuiranom okruženju, kontrola može biti:
  1. centralizovana (`ActiveDirectory`, `LDAP/Kerberos`)
  2. decentralizovana
  3. u dva nivoa

Kontrola pristupa orijentisana na podatke

- Posle uspešnog prijavljivanja, korisniku se dodeljuje pristup do jednog ili više računara ili aplikacija
- Svaki korisnik mora da ima profil sa dozvolama
- Odluke se donose ne samo na osnovu identiteta korisnika
- Opšti model je matrica pristupa po dve dimenzije: subjekat i pravo pristupa

Izvedbe matrice pristupa

1. **Liste za kontrolu pristupa** - za svaki objekat nabrajaju se korisnici i prava pristupa
2. **Kartice mogućnosti** - za koje je objekte i postupke korisnik ovlašćen i na koji način. Veći bezbednosni rizik jer kartice mogu da se razmenjuju po sistemu.

Vrste uljeza

1. Maskirač - pojedinac koji probija sistemsku kontrolu kako bi iskoristio validan korisnički nalog
2. Insajder - zvanični zlonamerni korisnik
3. Prikriveni korisnik - uzima kontrolu koju ima supervizor sistema (`root`)

Napadi se mogu podeliti na benigne i maligne.

Tehnike napada

Fajl sa lozinkama, npr. `/etc/shadow`, može se zaštititi na dva načina:

1. Jednosmerno šifrovanje - lozinka se čuva nekim od nepovratnih algoritama
2. Kontrola pristupa (-r———)

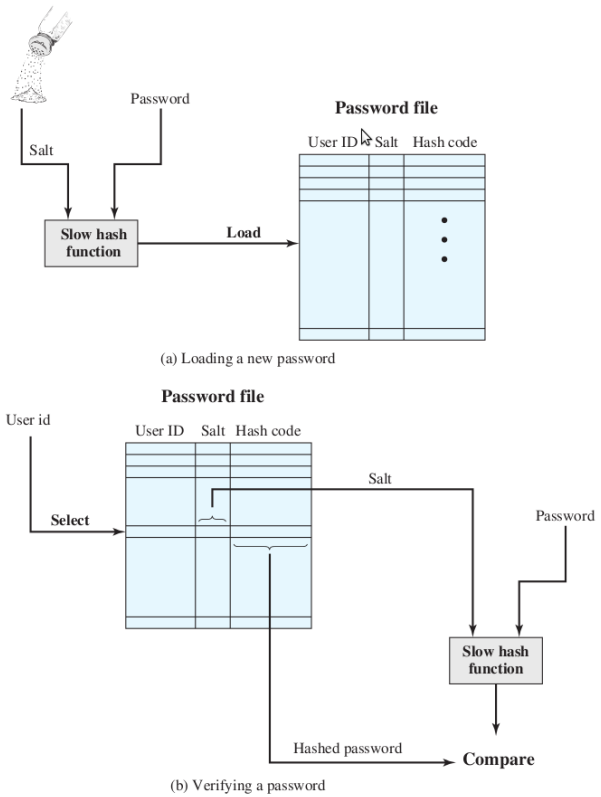
Tehnike napada provaljivanjem lozinke:

1. podrazumevana lozinka
2. kratke lozinke 1-3 znaka
3. reči iz rečnika
4. informacije o korisnicima, npr. imena dece itd.
5. brojevi telefona, soc. osiguranja...
6. iskorišćenje trojanca za zaobilaženje ograničenja
7. osluškivati vezu

UNIX *Data Encryption Standard* - DES

Uloga *salt*-a:

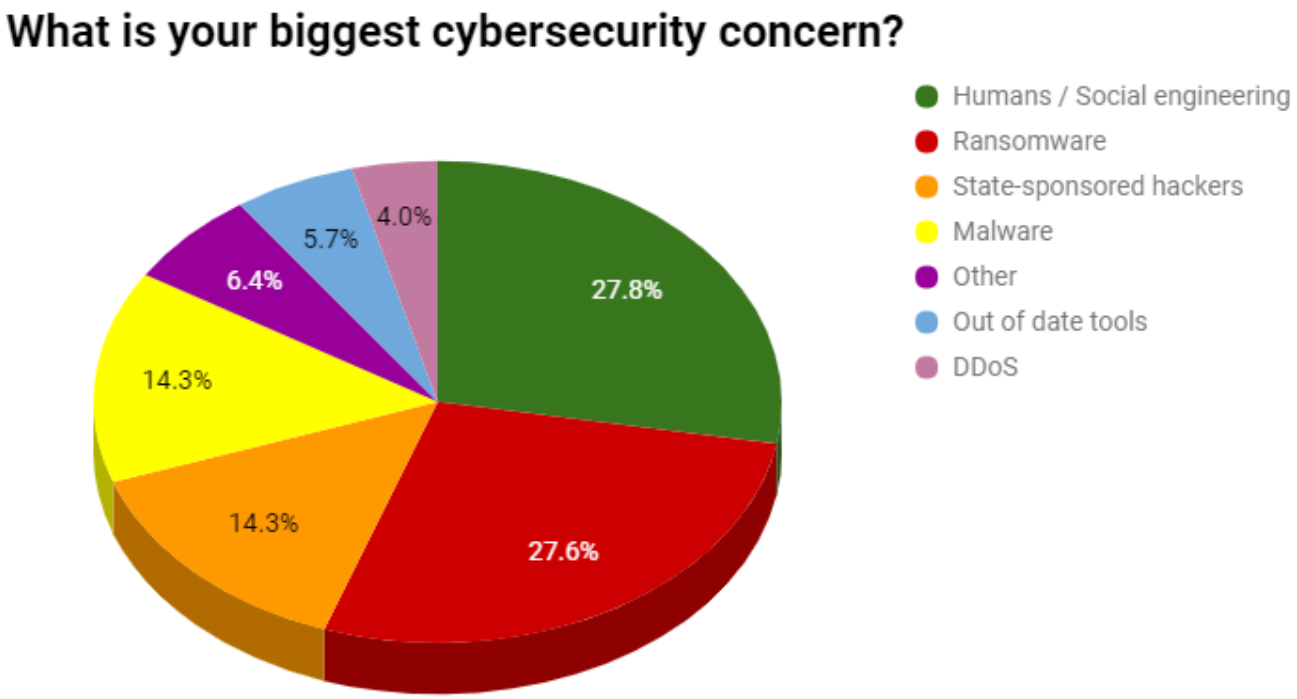
- Sprečava da se u fajlu pojave duplikati lozinki
- Produžava lozinku
- Sprečava korišćenje hardverske implementacije DES



Strategije za biranje lozinki

1. **Obuka korisnika** - ograničeni domet
2. **Lozinke koje generiše računar** - teško za pamćenje, pa postoji rizik da se negde zapišu
3. **Provera lozinke unapred** - ravnoteža sa restriktivnim pravilima, da korisnik izabere približno šta želi, a što je dovoljno bezbedno
4. **Provera lozinke unazad** - sistemski program za razbijanje

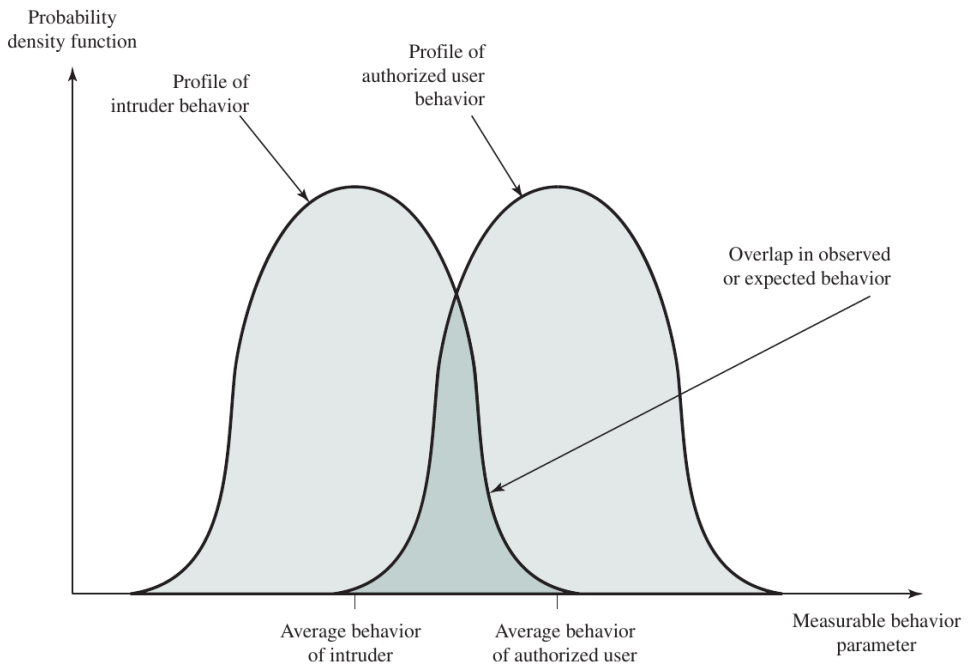
Nisu lozinke jedina meta napada



Čak sasvim suprotno!

Otkrivanje napada - ponašanje uljeza

1. **Brzina otkrivanja** - ako se napad otkrije na vreme, sprečena je veća šteta
2. **Sredstvo odvraćanja**
3. **Prikupljanje baze informacija** o tehnikama napada



Otkrivanje napada - ponašanje uljeza

1. **Otkrivanje statističke nepravilnosti**
   - Otkrivanje praga
   - Otkrivanje na osnovu osobina korisnika
   - Mašinsko učenje
2. **Otkrivanje na osnovu pravila** - ekspertni sistem
   - Otkrivanje nepravilnosti
   - Prepoznavanje upada (ekspertni sistem)

Poređenje dva pristupa

Statistička rešenja definišu **očekivano ponašanje**, dok rešenja zasnovana na pravilima definišu **ispravno ponašanje**.

Osnovno oruđe - nadzorni logovi

1. Subjekat - pokretač akcije
2. Akcija - operacija koju vrši subjekat na nekom objektu
3. Objekat - primalac akcije
4. Uslov izuzetka
5. Upotreba resursa - merljivi resursi sistema
6. Vremenska oznaka

Sigurnosni problemi

Buffer overflow (buffer overrun)

```
int main (int argc, char *argv[ ])
{
  int valid=FALSE;
  char str1[8];
  char str2[8];
  strcpy(str1,"START");
  gets(str2);
  if (strncmp(str1,str2,8)==0)
    valid=TRUE;
  printf("buffer: str1(%s),str2(%s),valid(%d) \n", str1, str2, valid);
}
```

```
$ gcc -o primer primer.c -m32
$ ./primer
START
buffer: str1(START),str2(START),valid(1)
$ ./primer
STARTSTART
buffer: str1(RT),str2(STARTSTART),valid(0)
```

Praktični primeri

- SSH passwordless autentifikacija (asimetricno šifrovanje)
- SSH hostbased autentifikacija
- PAM moduli
- Sertifikati za korišćenje Grida