- Utilizarea Sistemelor de Operare

Securitatea în sisteme Linux

- Curs 10 -
8.12.2005

Universitatea POLITEHNICA Bucureşti

Securitate

- [08.12.2005] Securitatea în sisteme Linux
- [15.12.2005] Securitatea în sisteme Windows
- [22.12.2005] Securitatea aplicațiilor de rețea

Ce inseamna securitatea?

- protectia adecvata a sistemului (mecanisme de protectie a memoriei, a proceselor)
- protectia fata de mediul extern in care sistemul opereaza
- se doreste protectia atat a informatiilor din sistem, cat si a resurselor fizice a calculatorului
- ne vom referi la notiunile de securitate care se aplica asupra unui sistem de operare

Problematica securitatii

- multe companii detin informatie pretioasa pe care doresc sa o protejeze (stocata de obicei pe calculator)
- se spune ca un sistem este sigur daca resursele sale sunt utilizate si accesate in orice imprejurare asa cum se doreste
- **nu este posibila obtinerea securitatii totale**
- un sistem este total sigur numai daca nu este conectat la lumea exterioara

Niveluri de securitate

- la nivelul sistemului de operare: impiedicarea accesului neautorizat la o resursa (un fisier, un proces, o zona de memorie)
- la nivelul fizic: incaperile ce contin sistemel de calcul vor fi protejate de accesul persoanelor neautorizate
- la nivelul persoana: utilizatorii vor fi alesi cu grija pentru a reduce probabilitatea ca un utilizator sa permita accesul unei persoane neautorizate

Securitatea la nivelul sistemului de operare

- se bazeaza pe suportul de protectie oferit de hardware
- sistemele mai vechi (MS-DOS, Mac OS) nu furnizau securitate – hardware-ul utilizat nu permitea protectia memoriei sau a I/O
- sistemele moderne precum Windows NT, Linux au fost proiectate astfel incat sa poata asigura securitate (suportul hardware oferit permite acest lucru)

Autentificarea

- protectia sistemului depinde de capacitatea acestuia de identificare a programelor si proceselor care se executa
- aceasta se bazeaza pe capacitatea de identificarea a utilizatorilor sistemului
- cum stim ca un utilizator este autentic?
  - poseda o entitate de identificare (cheie sau card)
  - un nume (identificator) de utilizator si parola
  - atribut de utilizator (amprenta, retina, semnatura)

Parole

- cel mai utilizat mecanism de abordare
- utilizatorului i se cere introducerea unei parola la intrarea in sistem; se compara parola introdusa cu cea stocata de sistem
- de obicei introducerea unei parole se face in modul ECHO OFF pentru a impiedica “shoulder sourfing”
- principala problema: pastrarea secreta a unei parole – o parola poate fi ghicita, descoperita intamplator, sau transferata de la un utilizator autorizat la unul neautorizat

Ghicirea unei parole

- prin incercari, pe baza numelui utilizatorului sau a unor nume/cuvinte cu legatura (nume de animale preferate, numele copiilor, zile de nastere, etc.)
- in 1997, un sondaj efectuat in Londra -> 82 % din parole puteau fi ghicite usor
- utilizarea fortei brute – o parola cu 4 cifre are 10000 de posibilitati; daca s-ar incerca o parola la fiecare milisecunda, in 10 secunde s-ar putea ghici parola

Protectie la ghicirea parolelor

- un singur cont (identificator) pentru fiecare utilizator (nu vor exista mai multi utilizatori care sa imparta acelasi cont)
- utilizarea parolelor generate aleator de sistem – problema: greu de retinut
- verificarea periodica a parolelor utilizatorilor
- password aging: fortarea schimbarii parolei dupa o anumita perioada
- criptarea parolelor

Parolele in UNIX

- se utilizeaza un program de criptare
- se cripteaza foarte usor si rapid parola
- decriptarea este foarte grea, de dorit imposibila
- sistemul mentine un fisier numai cu parolele criptate
- se pot totusi genera parola criptate dupa un dictionar si sa se verifice cu cele din fisier
- de obiecei nu se va permite accesul la acest fisier

Parolele in UNIX (2)

- la inceput se pastrau criptat in **`/etc/passwd`**
- actualmente parolele criptate se pastreaza in **`/etc/shadow`**, fisier la care numai administratorul are acces
- intrare in **`/etc/passwd`**

```
guest:x:12754:0::/home/guest:/bin/bash
```

- intrare in **`/etc/shadow`**

```
guest:jZGM7QoMqWGOA:12754:0:99999:7:::
```

Exemple de parole sigure

- minim 7 caractere
- atat caractere upper case si lower case
- cel putin un caracter special sau numeric
- nu trebuie sa fie nume de oameni sau cuvinte de dictionar
- usor de retinut

- Numele câinelui meu este Brusc. → NcmeB.!
- I check my e-mail every 3 hours → Icme-me3h
- Muraturile din 3 borcane sunt acre, dar inca bune → Md3bsa,dib

Contul de root

- cel mai cautat cont; detine controlul absolut al masinii si, posibil, al altor calculatoare din retea
- trebuie folosit **numai** atunci cand este nevoie (in rest se vor utiliza numai conturi obisnuite)
- folosirea necorespunzatoare poate duce la pierderea de informatii in mod involuntar
- comanda sudo permite unui alt utilizator (cineva foarte de incredere) rularea unui set restrans de comenzi cu privigii de root
- nu trebuie folosit pentru un program care permite lucrul cu un shell → utilizatorul va putea face orice ca si root

Securitatea fisierelor

- stabilirea parametrului **umask** de creare a fisierelor cat mai restrictiv
- setari tipice pentru umask: 022, 027, 077
- permisiunile de creare pentru un fisier se obtin prin realizarea unui SI logic intre permisiunile implicite (666 pentru fisiere si 777 pentru directoare) si masca inversata
- exemple
  - director: implicit 777; umask: 022 → 755 (rwxr-xr-x)
  - fisier: implicit 666; umask: 027 → 640 (rw-r-----)
- se foloseste comanda umask: $ umask 033

Securitatea fisierelor (2)

- unele aplicatii (precum ping, traceroute) necesita prezenta unui bit special (SUID/SGID); prezenta acestui bit pe un executabil permite executia acestuia cu drepturile celui ce detine executabilul (de obicei root) – unul din tipurile de atacuri preferate de atacatori.
- numai aplicatiile sigure trebuie sa aiba setat bit-ul respectiv
- bitul SUID este prezent pe pozitia de user-execute:

```
razvan@valhalla:~$ ls -l /bin/ping
-rwsr-xr-x  1 root root 30764 2003-12-22 17:18 /bin/ping
```

Securitatea retelei

- packet sniffers: atacatorul asculta pe portul Ethernet pentru siruri precum “passwd” sau “login” si capteaza traficul ce urmeaza – solutie: folosirea ssh sau a parolelor criptate
- DoS – Denial of Service – atacatorul incearca sa faca unele resurse prea ocupate pentru a face fata la cereri
  - Ping Flooding (smurfing)
  - Ping o' Death
  - contramasura: monitorizarea traficului – tcpdump – si utilizarea unui firewall

Limitarea drepturilor utilizatorilor

- un utilizator din cadrul sistemului poate genera DoS (crearea unui numar foarte mare de procese, alocarea de memorie peste limitele sistemului, umplerea spatiului de pe disc)
- solutii:
  - **quota** – fiecare utilizator primeste o cota (o limita de spatiu in cadrul sistemului de fisiere)
  - **/etc/login.defs**
  - **utilizarea PAM** (Pluggable Authentication Module) -> /etc/pam.d/limits.conf

chroot jail

- **int chroot (char *path);**
- se schimba directorul radacina pentru aplicatia curenta la directorul curent
- in felul acesta aplicatia nu poate sa schimbe directorul de lucru in cadrul unui director aflat mai sus in cadrul ierarhiei
- pentru siguranta maxima, orice fisier adaugat in “jail” trebuia analizat pentru a elimina posibilitatea “evadarii”

setrlimit

- stabileste limite de utilizare a resurselor
- exista o limita soft si o limita hard (ca si la **quota)**
- se pot stabili limite de procese, de fisiere, de memorie, de dimensiune a stivei, a timpului petrecut de procesor, etc.
- getrlimit si getrusage pot fi utilizate pentru a determina limitele stabilite sau procentul de utilizare a unei resurse

ulimit

- asemanatoare cu setrlimit dar poate fi apelata din interpretorul de comenzi; limitarea se realizeaza asupra shell-ului si asupra proceselor pornite de acesta
- sintaxa: `ulimit optiuni limit`
- se poate stabili o limita hard/soft sau nelimitata
- optiunile folosite sunt cele care selecteaza resursa dorita (`-f` – dimensiunea maxima a fisierelor; (`-s` – dimensiunea maxima a stivei, etc.)

Linux capabilities

- o noua modalitate de implementare a privilegiilor
- se trece de implementarea traditionala in care procese erau fie privilegiate (efective user ID = 0 – root) sau neprivilegiate
- capabilitatile divid privilegiile asociate traditional root-ului in unitati distincte ce pot fi activate/dezactivate independent
- implementate de la versiunea 2.2 a kernel-ului
- sunt pastrate la procesele fiu create cu ajutorul apelului fork

Linux capabilities (2)

- capabilitati sunt descrise in POSIX 1003.1e
- Linux adauga capabilitati specifice
- se utilizeaza biblioteca libcap: **capget, capset**
- exemple:
  - CAP_CHOWN – se poate schimba posesorul
  - CAP_KILL – se pot trasmite semnele proceselor altor utilizatori
  - CAP_SETUID – se poate schimba bitul SETUID
  - CAP_NET_RAW – socket-i raw (pentru **ping**)
- avantaj folosire capabilitati: vulnerabilitati ce conduc la obtinerea contului de root vor fi mai putin frecvente

Studiu de caz: vsftpd

- vsftpd – Very Secure FTP Daemon
- proiectat pentru sisteme UNIX cu accentul pe securitate
- implementarea principiului celui mai mic privilegiu (principle of least privilege)
  - **principle of distrust**: fiecare proces al aplicatiei implementeaza doar ce este nevoie; restul se realizeaza de alte procese si se folosesc mecanimse IPC
  - **chroot:** schimbarea directorului radacina la unul vid unde nu se pot produce pagube

Studiu de caz: vsftpd (2)

- lucrul cu reteaua se realizeaza de catre un proces ce lucreaza intr-un “chroot jail”
- operatiile privilegiate sunt realizate de un proces parinte (codul este cat mai mic posibil)
- “neincredere” fata de apelurile de biblioteca
  - implementarea proprie a unui utilitar de listare a continutului unui director (/bin/ls)
  - apelurile de biblioteca externe sunt incadrate intr-un fisier propriu pentru detectarea mai usoara a problemelor de securitate

Security Enhanced Linux (SELinux)

- o versiune a kernel-ului de Linux si a utilitarelor asociate
- isi propune sa demonstreze utilitatea controlului accesului obligatoriu (mandatory access controls)
- caracteristica esentiala este ca utilizatorul nu are control absolut asupra resurselor pe care le creeaza
- se doreste ca fiecare program/server sa utilizeze cantitatea minima de resurse necesara pentru a rula
- dispare conceptul de utilizator privilegiat (root)
- SELinux s-a unit cu seria 2.6 a kernel-ului (merging)

Cresterea sigurantei unui sistem

- backups
- security updates
- acordarea drepturilor minime pentru utilizatori/resurse
- verificarea periodica a starii sistemului
- keep it simple
- monitorizarea traficului
- go here:
  http://www.cert.org/tech_tips/usc20_full.html

- paranoia este o virtute?

Este posibila crearea unui sistem sigur?

- **da**
- **de ce nu construiesc sisteme sigure?**
  - **utilizatorii nu sunt dispusi sa inlocuiasca un sistem foarte usor (Microsoft lanseaza un nou produs pe piata – SecureOS, dar nu ruleaza aplicatii Windows)**
  - **un sistem sigur trebuie sa fie simplu; adaugarea de noi caracteristici inseamna expunerea la noi riscuri de securitate**

Cat de sigur este un sistem?

- gradul de securitate al unui sistem este dat de gradul de securitate al celei mai slabe componente (weakest link)
- prezenta unui server cu un grad foarte inalt de securitate in cadrul unei retele de statii nesigure nu ajuta la cresterea securitatii

Sisteme de operare orientate spre securitate

- OpenBSD – complet orientat spre securitate
- TrustedBSD – subproiect al FreeBSD
- Linux
  - Adamantix
  - Hardened Gentoo
  - Immunix
- Solaris
  - Trusted Solaris