LUENTO 21

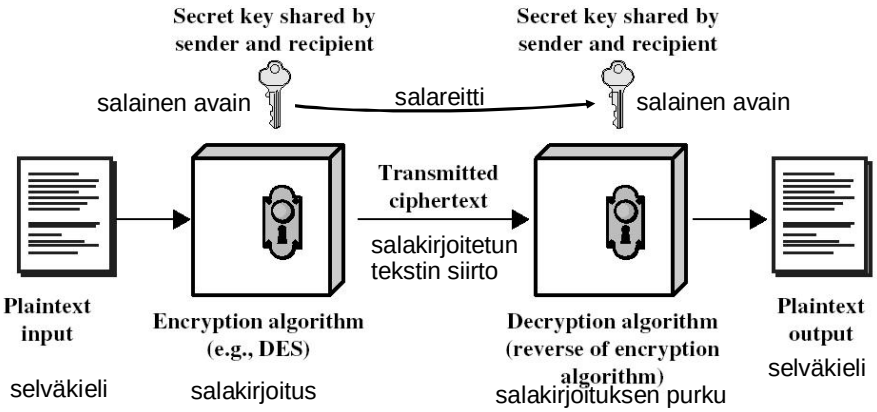
Security

Ch 16 [Stal 05]

1

Perinteinen, symmetrinen salaus

Sama avain molemmilla!

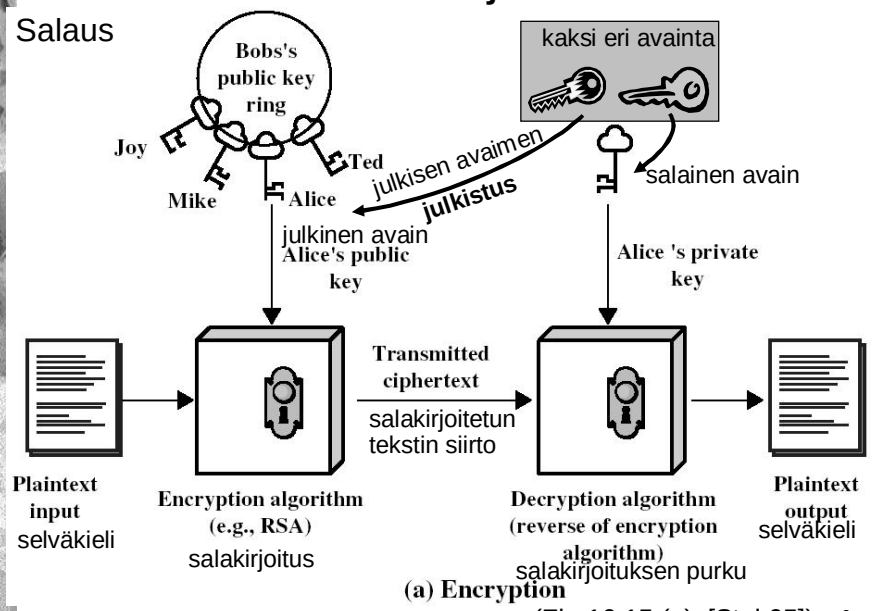


(Fig 16.14 [Stal 05])

2

Julkisen avaimen salakirjoitusmenetelmä

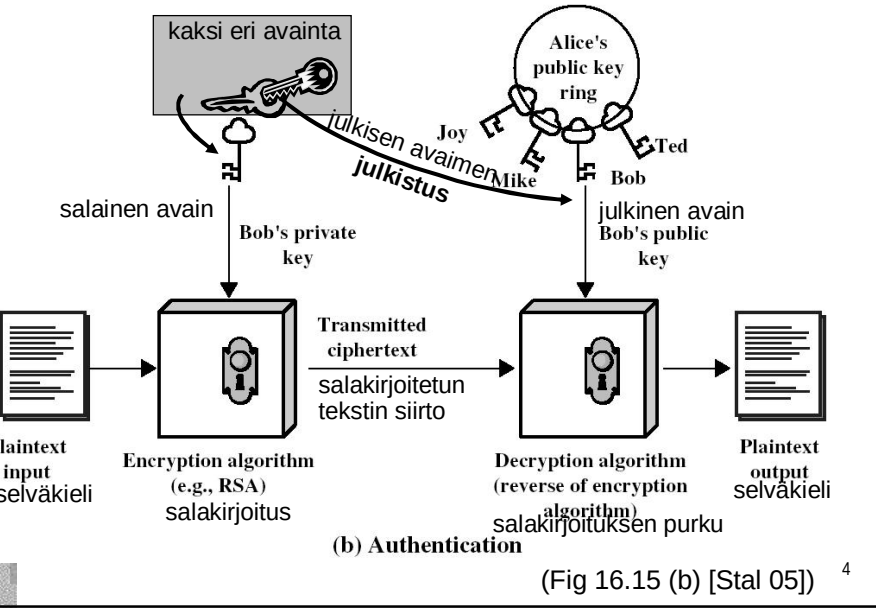
Salaus



(Fig 16.15 (a) [Stal 05])

3

Tunnistus julkisen avaimen menetelmällä



(Fig 16.15 (b) [Stal 05])

4

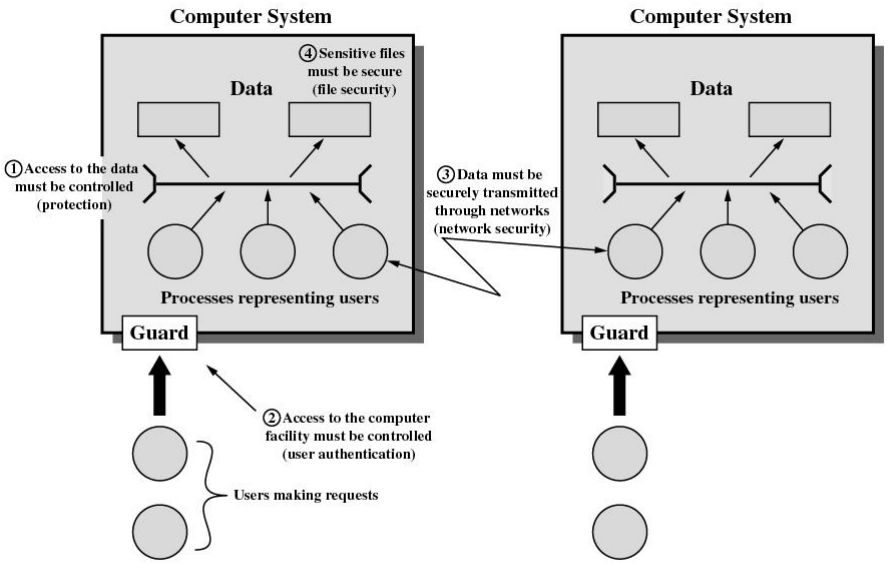


Figure 16.1 Scope of System Security [MAEK87]

Requirements

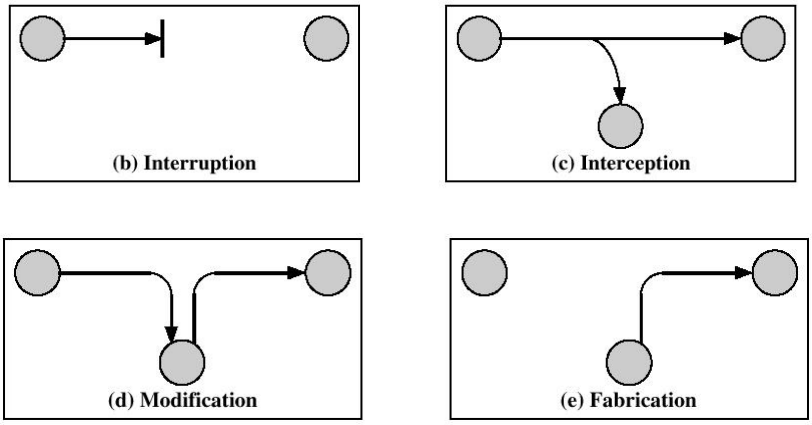
- confidentiality, secrecy
  - Information accessible for reading only by authorized parties
- integrity
  - Information (assets) can be modified only by authorized parties
- availability
- authenticity
  - System must be able to verify the identity of the user

6

Threats

DoS – denial of service

(Fig 16.2 [Stal 05])



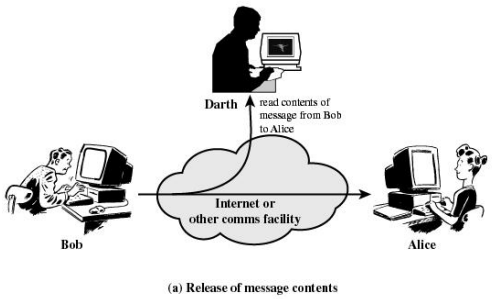
See also Tbl 16.1 [Stal 05]

7

Passive attacks (sniffing, listening)

Fig 16.3 [Stal 05]

- Breaks confidentiality, but not integrity
- release of contents, traffic analysis



(a) Release of message contents

8

Active attacks

- Breaks integrity
- masquerade
- Replay
- modification of msg contents
- DoS = denial of service
  - overload, sabotage

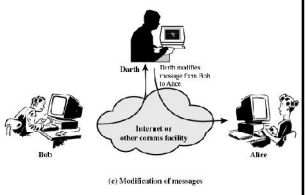


Fig 16.4 (a) [Stal 05]

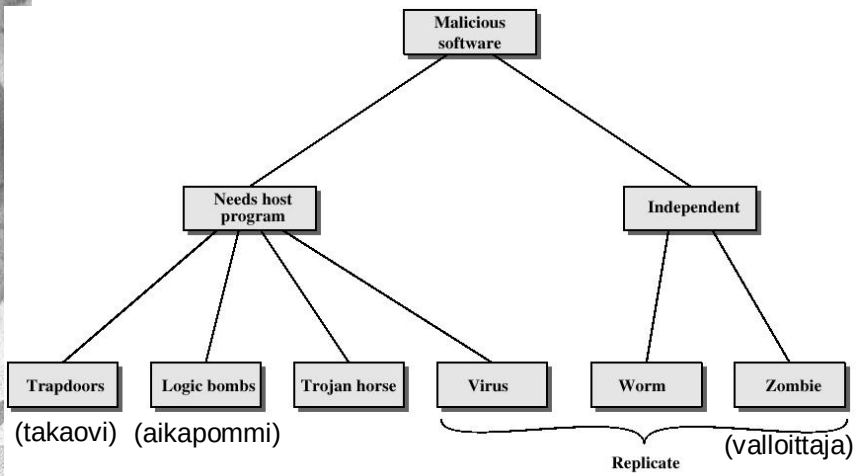
Fig 16.4 (b) [Stal 05]

Fig 16.4 (c) [Stal 05]

Fig 16.4 (d) [Stal 05]

9

Malicious software - classification



(Fig 16.8 [Stal 05])

10

```
while (TRUE) {                          while (TRUE) {
    printf("login: ");                      printf("login: ");
    get_string(name);                       get_string(name);
    disable_echoing( );                     disable_echoing( );
    printf("password: ");                   printf("password: ");
    get_string(password);                   get_string(password);
    enable_echoing( );                      enable_echoing( );
    v = check_validity(name, password);v = check_validity(name, password);
    if (v) break;                           if (v || strcmp(name, "zzzzz") == 0) break;
}                                       }
execute_shell(name);                    execute_shell(name);
```

(a) (b)

Fig. 9-10. (a) Normal code. (b) Code with a trap door inserted. [Tane 01]

(a) (b)

Fig. 9-9. (a) Correct login screen. (b) Phony login screen.

11

Buffer overflow

[Tane 01]

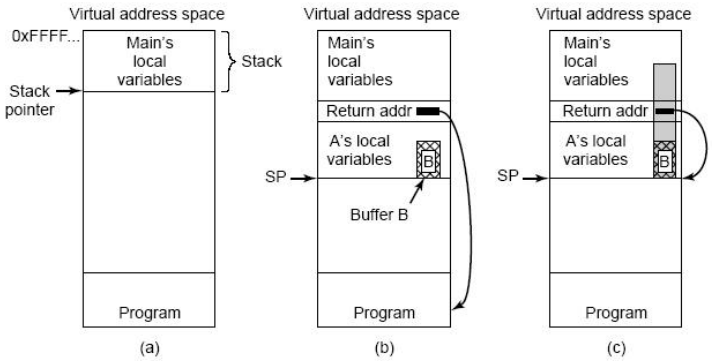


Fig. 9-11. (a) Situation when the main program is running. (b) After the procedure *A* has been called. (c) Buffer overflow shown in gray.

12

Viruses

- Virus dropper inserts virus onto the system
- Many categories of viruses,
  - Parasitic (in a File)
  - Memory resident
  - Boot
  - Macro
  - Polymorphic
  - Stealth

13

A Boot-sector Computer Virus

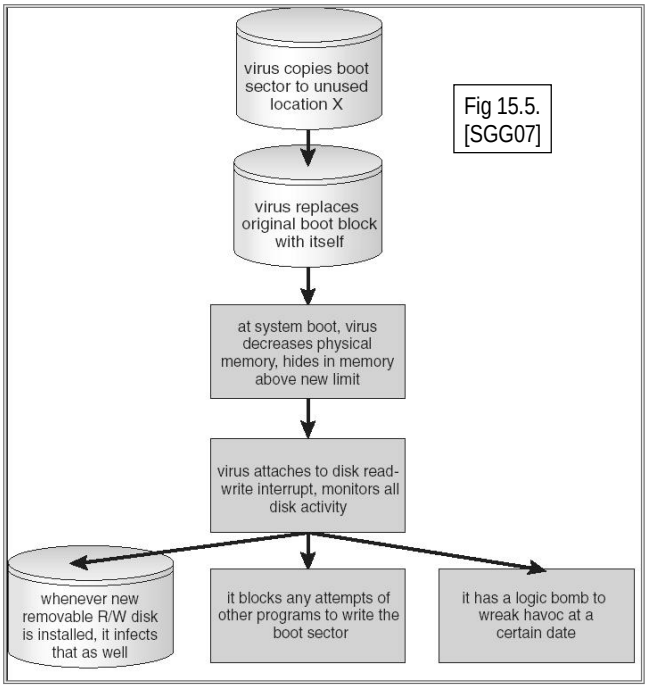


Fig 15.5. [SGG07]

14

```
MOV A,R1        MOV A,R1        MOV A,R1        MOV A,R1        MOV A,R1
ADD B,R1        NOP             ADD #0,R1       OR R1,R1        TST R1
ADD C,R1        ADD B,R1        ADD B,R1        ADD B,R1        ADD C,R1
SUB #4,R1       NOP             OR R1,R1        MOV R1,R5       MOV R1,R5
MOV R1,X        ADD C,R1        ADD C,R1        ADD C,R1        ADD B,R1
                NOP             SHL #0,R1       SHL R1,0        CMP R2,R5
                SUB #4,R1       SUB #4,R1       SUB #4,R1       SUB #4,R1
                NOP             JMP .+1         ADD R5,R5       JMP .+1
                MOV R1,X        MOV R1,X        MOV R1,X        MOV R1,X
                                                MOV R5,Y        MOV R5,Y
(a)             (b)             (c)             (d)             (e)
```

Fig. 9-17. Examples of a polymorphic virus.

[Tane 01]

15

Automatic virus detection/ analysis: Digital Immune System

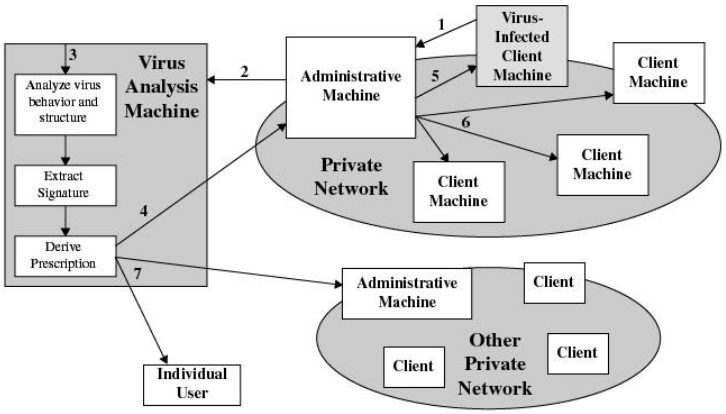


Figure 16.9 Digital Immune System

16

Intruders

- Growing problem
  - Using unauthorise account — masquerader
  - Misusing own account — misfeasor
  - Hiding the usage — clandestine user
    - get root access, hide tracks
- How
  - Guess / try / crack passwds
  - Trojan horse
  - Sniffing
    - Clear text username /passwd in a message
  - Phishing
  - Man in the browser
    - Malprogram that activates only when browser access bank pages, then collects data and send it somewhere

17

Access rights

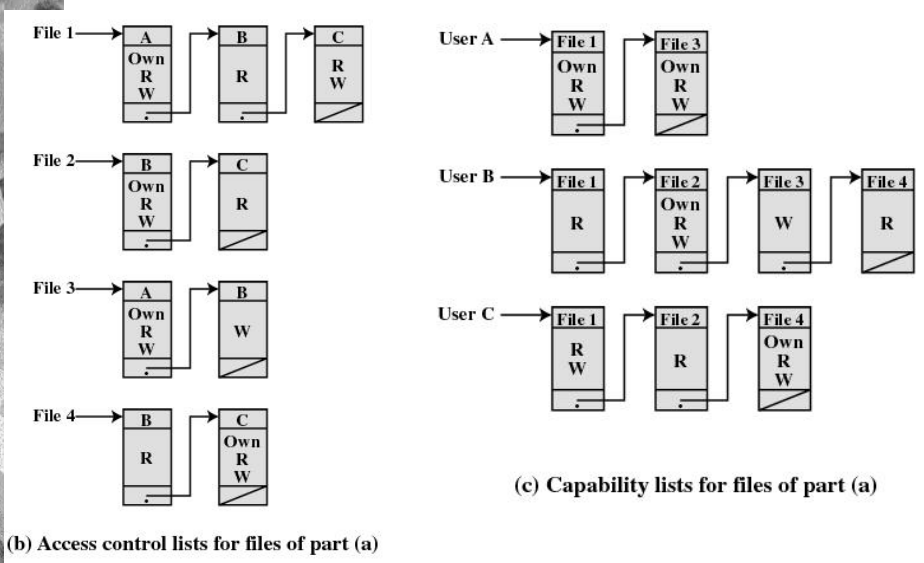
- Identified user (rows in access matrix)
- Domains: groups of resourses (and users) that have certain access restrictions (rows and columns)
- Resources, files: (columns in the access matrix)
- Access rights checked by OS (at every access, or at open)

[Stal 05]

	File 1	File 2	File 3	File 4	Account 1	Account 2
User A	Own R W		Own R W		Inquiry Credit	
User B	R	Own R W	W	R	Inquiry Debit	Inquiry Credit
User C	R W	R		Own R W		Inquiry Debit

(a) Access matrix

18



(b) Access control lists for files of part (a)

(c) Capability lists for files of part (a)

[Stal 05] Fig 16.5

19

Access control policies

- DAC – discretionary access control
  - Owner decides
  - User can dynamically change access rights
  - User in with normal user data

Choise, Option, Select

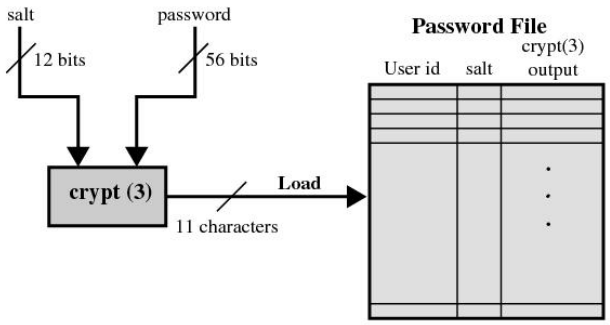
Remove read?

- MAC – mandatory access control
  - Central policy defines te rights
  - Users cannot change access rights
  - Used with classified information

Obligatory

20

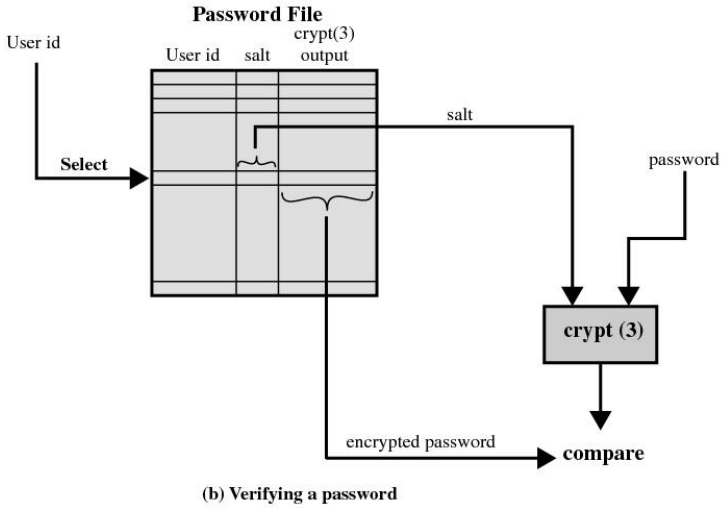
UNIX password model



(a) Loading a new password

21

UNIX password model



(b) Verifying a password

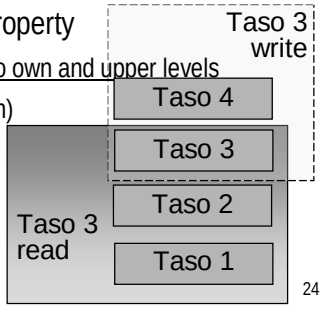
22

Luotettu järjestelmä (trusted system)

23

Multilevel Security

- Information classification
  - unclassified, confidential, secret, top secret
- Users have trust level
- No-read-up eli simple security property
  - Everyone can see (read) only own level or below classified data
- No-write-down eli *-property eli star-property
  - Everyone can produce (write) data only to own and upper levels
  - No leaks down (unless explicit permission)
- Needs Mandatory Access Control
- Example: reference monitor
  - Mediation, isolation, verifiability



24

Reference Monitor

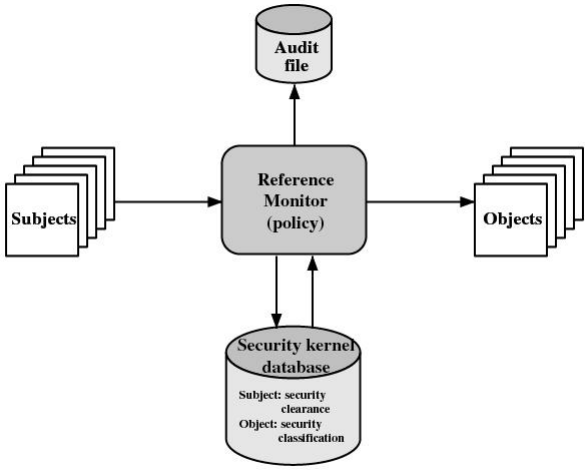


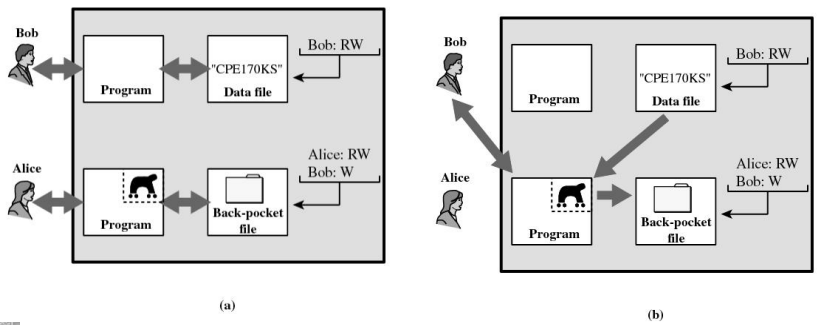
Figure 16.10 Reference Monitor Concept

25

Trojan example

Fig 16.11 (a, b) [Stal 05]

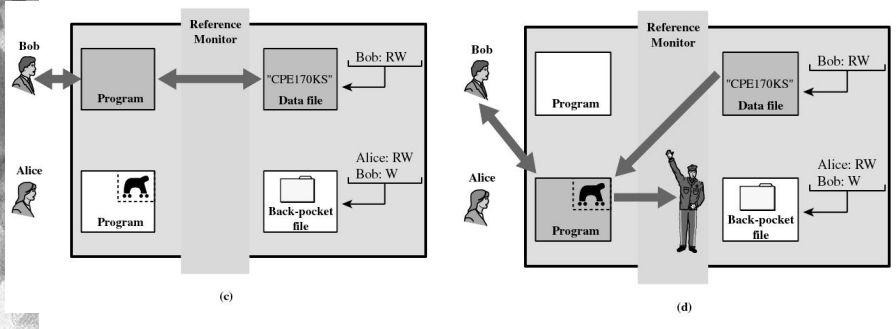
- Without reference monitor, only using access rights
- (Leak happens using trojan program)



26

Example using reference monitor

Fig 16.11 (c, d) [Stal 05]



Æ Security level is more important than access rights !

27

UNIX tietoturva

Ch 10.7 [Tane 01]

28

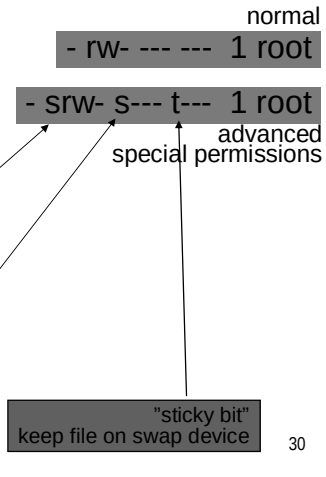
UNIX access rights

- File attributes (i-node)
  - uid, gid
  - Access rights for u,g,o (modes rwx)
- User (process) uid and gid from
  - **`/etc/passwd`** uid and primary effective gid
  - **`/etc/group`** other gids
- uid and gid passed to child processes and new files
  - Files: Umask in process control block

29

UNIX access rights

- File attributes (i-node)
  - uid, gid
  - Access rights for u,g,o (modes rwx)
- User (process) uid and gid from
  - **`/etc/passwd`** uid and primary effective gid
  - **`/etc/group`** other gids
- uid and gid passed to child processes and files
  - Files: Umask in process control block
- Temporary change of rights, example:
  - only root has w-right to */etc/passwd* file
  - *Passwd* program file has set SETUID bit
    - effective userid is the ownerid of the file passwd
  - User process gets root access rights for the execution duration of *passwd*-program (because root is the owner), thus, it can change its own password
  - Similar SETGID bit for groups
    - effective groupid



30

Windows 2000 Tietoturva

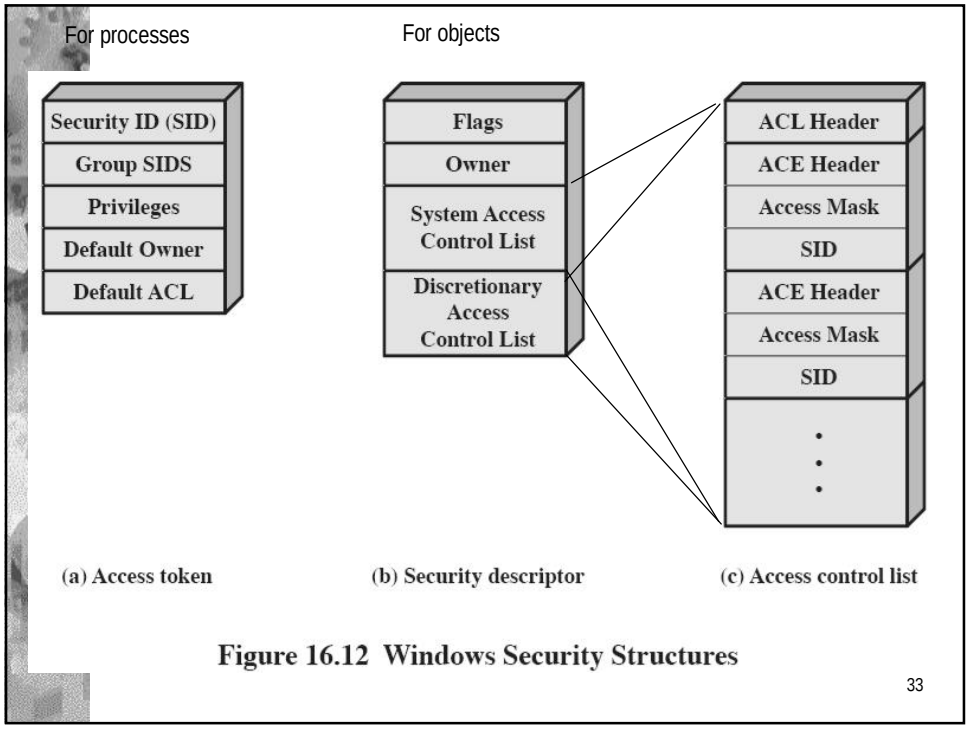
31

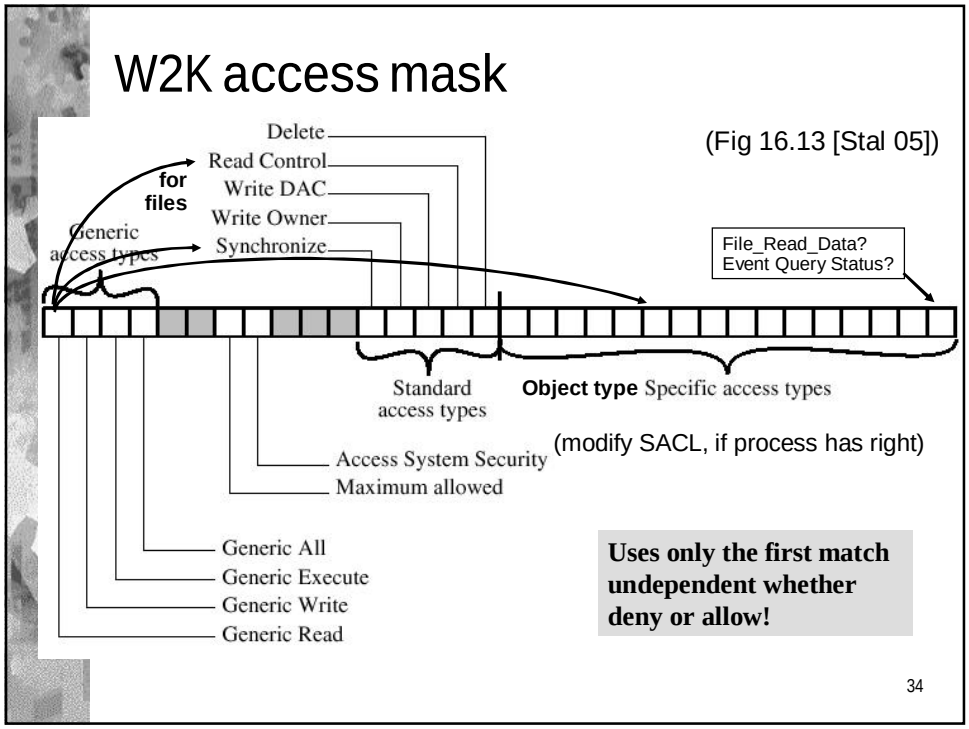
Computer Security Classifications

http://www.dynamoo.com/orange/summary.htm

- U.S. Department of Defense outlines four divisions of computer security: A, B, C, and D.
  - D – Minimal security.
  - C – Provides discretionary protection through auditing. Divided into C1 and C2. C1 identifies cooperating users with the same level of protection. C2 allows user-level access control.
  - B – All the properties of C, however each object may have unique sensitivity labels. Divided into B1, B2, and B3.
  - A – Uses formal design and verification techniques to ensure security.
- Windows on level C2: personal login, some restricted access to files

32


For processes
For objects
Security ID (SID)
Group SIDS
Privileges
Default Owner
Default ACL
Flags
Owner
System Access Control List
Discretionary Access Control List
ACL Header
ACE Header
Access Mask
SID
ACE Header
Access Mask
SID
(a) Access token
(b) Security descriptor
(c) Access control list
Figure 16.12 Windows Security Structures
33


W2K access mask
(Fig 16.13 [Stal 05])
Delete
Read Control
Write DAC
Write Owner
Synchronize
for files
Generic access types
File_Read_Data?
Event Query Status?
Standard access types
Object type Specific access types
(modify SACL, if process has right)
Access System Security
Maximum allowed
Generic All
Generic Execute
Generic Write
Generic Read
Uses only the first match undependent whether deny or allow!
34

W2K DACL Example (NTFS)