

Goal of Protection -

Protection can improve reliability by differentiating unauthorised users from authorised users.

The role of protection in a computer system is to provide a mechanism that enforces policies when using resources.

Mechanism - determines how something will be done.

Policies - decides what will be done.

Principles of protection - is the

"Principle of least privilege"

It dictates that programs, users, and even systems be given just enough privileges to perform their task.

It helps to produce a more secure environment.

Ex - Read only mode is required then it will never give R & W both modes to the user.

Domain of protection -

A computer system is a collection of processes and objects.

objects are -

H/w - CPU, Printer etc

S/w - files, programs etc

① A ~~process~~ process should be allowed to access only those objects for which it has authorization.

and

② A process should be able to access only those resources that it currently requires to complete its task.

This is also known as "need-to-know" principle.

Domain Structure - A process operates within a protection domain, which specifies the resources that a process may access.

Domain - (Set of objects + type of operations)

A domain is a collection of access rights that defines ability to execute an operation on an object.

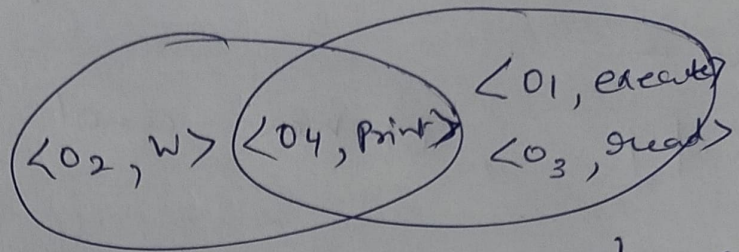
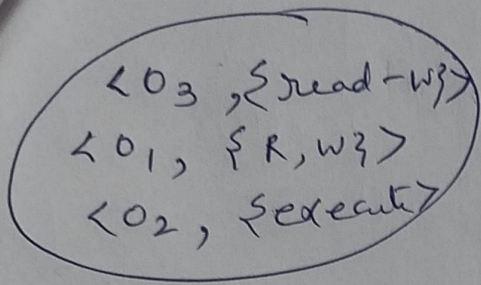
Access Right - $\langle \text{Object-Name, Right-Set} \rangle$

D1

D2

D3

(2)



Domains may share their access rights

Access Matrix - Protection Model can be represented by a matrix called as Access matrix.

rows - Domains

columns - objects

each entry - access right

$Access(i, j) \rightarrow$ Process in D_i can invoke an operation on O_j

Obj Domain	f_1	f_2	f_3	Printer	D_1
D_1	R		R		
D_2				P	Switch
D_3		R	E		
D_4	R w		R w		

$Access(1, 1) = R$

Process in D_1 can Read (operation) file f_1 .

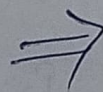
A process in D_2 can switch to Domain D_1

To allow controlled changes in the access matrix entries, Three operations required are -

Copy, owner and control.

Access Matrix with copy right -

obj / Domain	F1	F2	F3
D1	Execute		Write*
D2	E	Read*	E
D3	E		

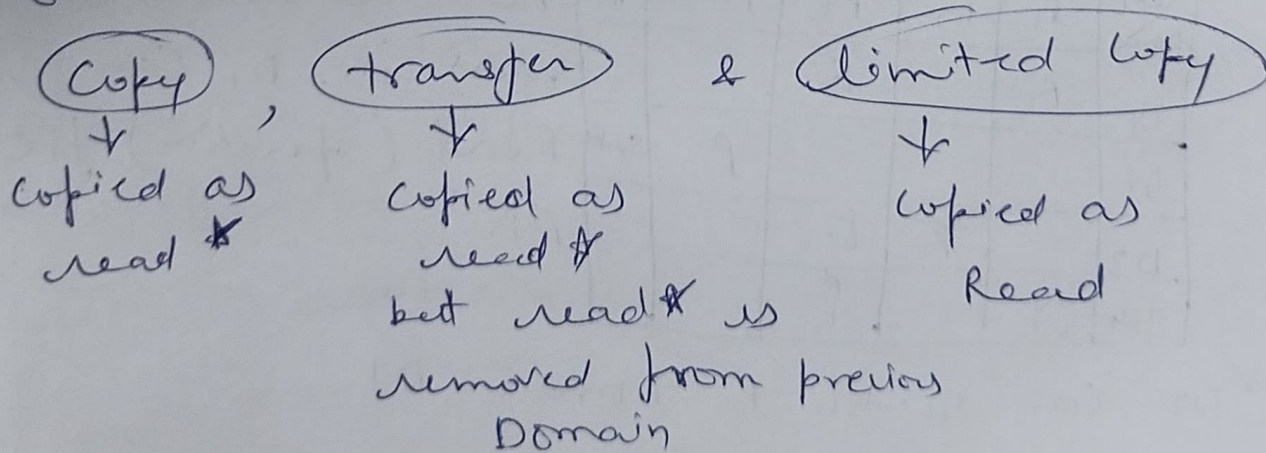


	F1	F2	F3
D1	E		Write*
D2	E	Read*	E
D3	E	Read	

Ability to copy an access right from one domain to another (in the same column) is copy right represented by *.

Propagation of copy right is limited means

now process running in Domain D3 can not copy access right Read to some other row as * is not there.



OWNER - It allows addition of new rights and removal of rights from access matrix.

If access (i, j) includes the owner right then a process executing in Domain D_i can add and remove any right in any entry of object column j .

$D \backslash O$	F1	F2	F3
D1	OWNER E		W
D2		Read write	R* OWNER
D3	E		

\Rightarrow

$D \backslash O$	F1	F2	F3
D1	OWNER E		W
D2		OWNER R* W*	R* W OWNER
D3		Write	

D is owner of an object, so can change any right

CONTROL - A mechanism is needed to change the entries in a row.

The control right is applicable only to domain objects.

\backslash	F1	F2	F3	D1	D2
D1	R			S	control
D2		W			write

D1 can change row of D2.

Implementation of Access matrix -

① Global Table -

$\langle \text{domain, object, right-set} \rangle$, search all these entries, if this triple found operation is allowed to continue.

(i) large table

② Access list of objects -

Each column in the access matrix can be implemented as an access list for the object.

operation π on an object o_j is attempted
we search the access list of o_j for
entry $\langle D_i, \pi \rangle$

③ Capability list for Domains -

A capability list for a domain is a list of objects together with the operations allowed on those objects.

Domain $\rightarrow \{ \text{set of capabilities} \dots \}$
Capability is (object + operation)

Capability list is associated with domain & a process can not access it directly. Capability list is accessible only by the operating system.

④. A Lock-key Mechanism.—

Each object has a list of unique bit patterns, called locks. Similarly each domain has a list of unique bit patterns called keys.

A process executing in a domain can access an object only if that domain has a key that matches one of the locks of the object.

list of keys of domain are managed by OS.