Problem with access control matrix

On a typical system, number of subjects and objects very large.

Most entries either blank indicating no access or the same (because implementations have a default setting)

Creation and deletion of subjects and objects require matrix to be managed carefully adding to complexity of code.

ACL’s and capabilities – variants based on access control matrix that eliminates many problems.

ACL – long list for each object what every subject is allowed to do with object.

Top to down

Generalize across subjects.

acl (file 1) = { (process 1, { read, write, own} ), ( process2,{append}) }

acl (file 2) ={ (process 1,{read} ),(process2 ,{read, own } ) }

Each subject and object has an associated ACL. Thus process1 owns file1 and can read or write to it ;process2 can only append to file1.

If a subject is not named in the ACL ,it has no rights over associated object.

If many subjects have same rights one can define wildcard to match unnamed subjects.

Abbreviations of ACL’s

UNIX systems divide set of users into 3 categories – owner, group owner and all other users.

rwx – read,write and execute rights

rw r none

Permissions represented as three triplets

Revocation of rights

Revocation or prevention of subject accessing object requires subject’s rights be deleted from object’s ACL.

Entry for a subject is removed from object’s ACL.

If only specific rights are to be deleted they are removed from specific subject’s entry in the ACL.

Capabilities

Like a row of ACM.

Each subject has a set of pairs , each pair contains object and a set of rights.

Generalize over objects.

Cap(process1 )= {(file1,{read,write} ) }

Capabilities encapsulate object identity.

Location of object in memory encapsulated in capability.

When process uses capability to access object, OS examines capability to determine object and the access to which process is entitled.

Implementation of capabilities

Three mechanisms used to protect capability – tags, protected memory, cryptography.

Tagged architecture has set of bits associated with each hardware word.

Tag has 2 states – set and unset.

Tag set - ordinary process can read but can’t modify word.

Tag unset - ordinary process can read and modify word.

Each capability has a checksum , checksum digitally enciphered .

Revoking access to object easy in ACL -remove all entries in list of that object

In capabilities, all capabilities granting access to object should be revoked.

Expensive, use global table to implement.

DAC

Individual user can set access control mechanism to allow or deny access to an object.

Base access rights on identity of subject and identity of object involved.

Owner constrains who can access it by allowing only particular subjects to have access.

Child keeps a diary. Child controls who can read it.

MAC

When a system mechanism controls access to an object, individual user cannot alter that access, control is mandatory access control also called rule based access control.

OS enforces mandatory access controls.

A security policy – 2 types of access controls, alone or in combination.

When Concentrating on rights part

Our examples largely been about files.

Think about files

Is there good set of rights that meet our needs for access control.

Seen lot of these things

Read

Write

Execute

When looked at unix based access control system these seemed at first to be enough.

(own) – flashed by briefly

Append – not there in linux

Is it good to have longer list of privileges, more fine control over each situation.

Append different from write

Need place to save high score list from time to time

Need some place to save data ,in this case it is my game high score -mygamehs.txt

Called this a text file as it’s just a mass of characters

mygame

mygamehs.txt

What rights should mygamehs.txt file have?

One right bit apart from others represented in illustrations.

That is the own.

Some people take ownership for granted. It’s in my directory. Of course I own it.

Want to give own right special consideration

What does own mean.

Access control matrix – Description of system at an instant of time

Things can change. Can create a new file – file3

Or a new user can arrive.

All of a sudden matrix not good enough.

Want to change rights

How to go about changing things ?

DAC - Discretionary access control

MAC - Mandatory access control

Many MAC’s

In cryptography MAC was message authentication codes

Authentication MAC was media access control

Your security manager can decide which policy is suitable for your access control.

These things are deciding within which limitations things can change

DAC – If you own object, it’s at your discretion what to do with it.

own right is everything

If it is Mandatory AC, own gives me some ability to change things

But I am ultimately subject to rules of the system.

System has rules , so it is Rule Based

System we have seen are largely DAC systems

We have already seen examples of policies where system decides what I am allowed to do . Bell Lapadula is a MAC based system.

Discretionary access control over our notes.

In exam hall ,will be writing notes on a sheet of paper , cannot pass to others.

Got access control lists,capabilities

Role based access control

In large businesses rights on certain objects less to do with your name, more to do with your role in the company.

Marketing has different rights from economy department.

Principles of how we work out access control , broken into roles and subjects ,we are mapping actual subjects to the roles.

Capability based systems – I can now have myself as a subject . Hopefully mechanisms of propagating rights through cloud.

Give (preferably IT related) illustrative examples from your life and experience of when the privileges that are associated with a resource...

1. can be decided by you, the resource owner. (DAC)
2. are not decided by you even if you are the owner since that is controlled by the system that the resource is contained within. (MAC)
3. are decided by you to a limited fashion, but beyond that it are controlled by system rules. (MAC with some DAC)
4. are decided by the original creator of the resource more than whoever owns it. (something else not yet studied!...)

Answer text

DAC -a notes document written by yourself in microsoft word

MAC - a system OS executable file which you cannot change or modify on your system

MAC with some DAC - a file you are working on  which contains information about a product of the company you are working for and you

have signed a non-disclosure agreement for the company

Original creator - A private licensed software developed by a person