Computer Security (COM-301)

Discretionary Access Control

Carmela Troncoso
SPRING Lab
carmela.troncoso@epfl.ch

Simon says

Simon is the owner of Color OS, a simple OS that has 4 files: `red`, `yellow`, `green`, and `blue`.

Simon says that a user Harry can read the file red, can write yellow, and can read and execute blue. What is the capability that Simon should give to Harry?

(a)
`red`: {(Harry, read, no write/execute)}
`yellow`: {(Harry, write, no read/execute)}
`blue`: {(Harry, read/execute, no write)}.

(b)
`red`: {(Harry, read)}
`yellow`: {(Harry, write)}
`blue`: {(Harry, read/execute)}.

(c)
Harry: {(`red`, read), (`yellow`, write), (green, -), (`blue`, read/execute)}

(d)
Harry: {(`red`, read), (`yellow`, write), (`blue`, read/execute)}

The correct answer is (d).
The options (a) and (b) are not capabilities, and (c) has an empty slot. Recall that negative permissions are not included in capabilities to not waste space.

Confusion

Which of the following security violations is **NOT** caused by a confused deputy?

(a) A hacker gains access to a user's social network account by getting the user's browser to send the hacker this user's credential

(b) A virus infects an email client to send spam

(c) A journalist tricks a banker into revealing the bank statements of a famous singer

(d) A detective leaks information to a criminal using a covert channel

(D) The detective is no tricking any privileged process into doing something against the security policy

To ACL or not ACL

We are back in COVID times, and contact tracing is needed.

You are setting up a new Bar in Lausanne, and have one computer for people to give their phone numbers. To make it easy you let them register (create a row with their phone number) and then only add the date where they visit the bar.

Is ACL a good solution to manage access to the database? (to avoid that users can influence the contact tracing process)

And if instead of one computer you have one computer per table but they are not connected to the internet?

Principals: Customers (N principals of type customer), Admin (bar owner)
Objects: rows in the database. Rows contain the telephone number, and the dates
[You could also justify rows contain other personal information, though it does not appear in the question]

Customers must have access to their own row to write / append the days of the visit.

1 computer for the whole Bar: ACLs and capabilities would be ok. One could even say ACL is more robust as it avoids delegation (Customer A adding a date on Customer B's row if Customer B gives A their capability). This would depend on the threat model under consideration.

M computers non connected: ACLs become hard to manage, as now the permissions are distributed. To revoke/add a permission you need to go to all computers. A capability would allow you to centralize the permission control in that capability. (of course, there is still the delegation problem, that in a real case you would have to weight against the complexity of maintaining the ACL).

Proposition 1: Bob cannot write in Alice's file!
Proposition 2: Bob can overwrite Alice's messages!

Solution: "setuid" mechanism:
The solution is to let Bob execute `msg` as if he was Alice. For this we can set the **suid** bit **s**

```
-rws--x---      Alice Alice+Bob msg

-rwx------      Alice Alice+Bob msgfile
```

When the suid bit is set, when the program is executed it runs with the permissions of the owner. When the program ends, the permissions are returned to normal.

Now, when Bob executes `msg`, it will run with Alice's permission. Thus, it can open `msgfile` and write the message. When the message end, Bob cannot write on the file so he cannot overwrite Alice's messages.

Like with any other program, it is very hard to know if the program is doing only what it is meant to do. Thus, setting suid on root programs is very dangerous, as they run on the TCB! If something goes wrong there are no more protections