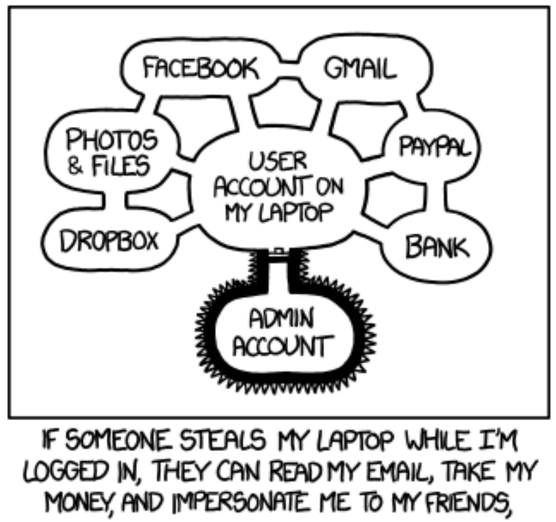
# FIT 1047

## Introduction to computer systems, networks and security

MONASH University




IF SOMEONE STEALS MY LAPTOP WHILE I'M LOGGED IN, THEY CAN READ MY EMAIL, TAKE MY MONEY, AND IMPERSONATE ME TO MY FRIENDS,

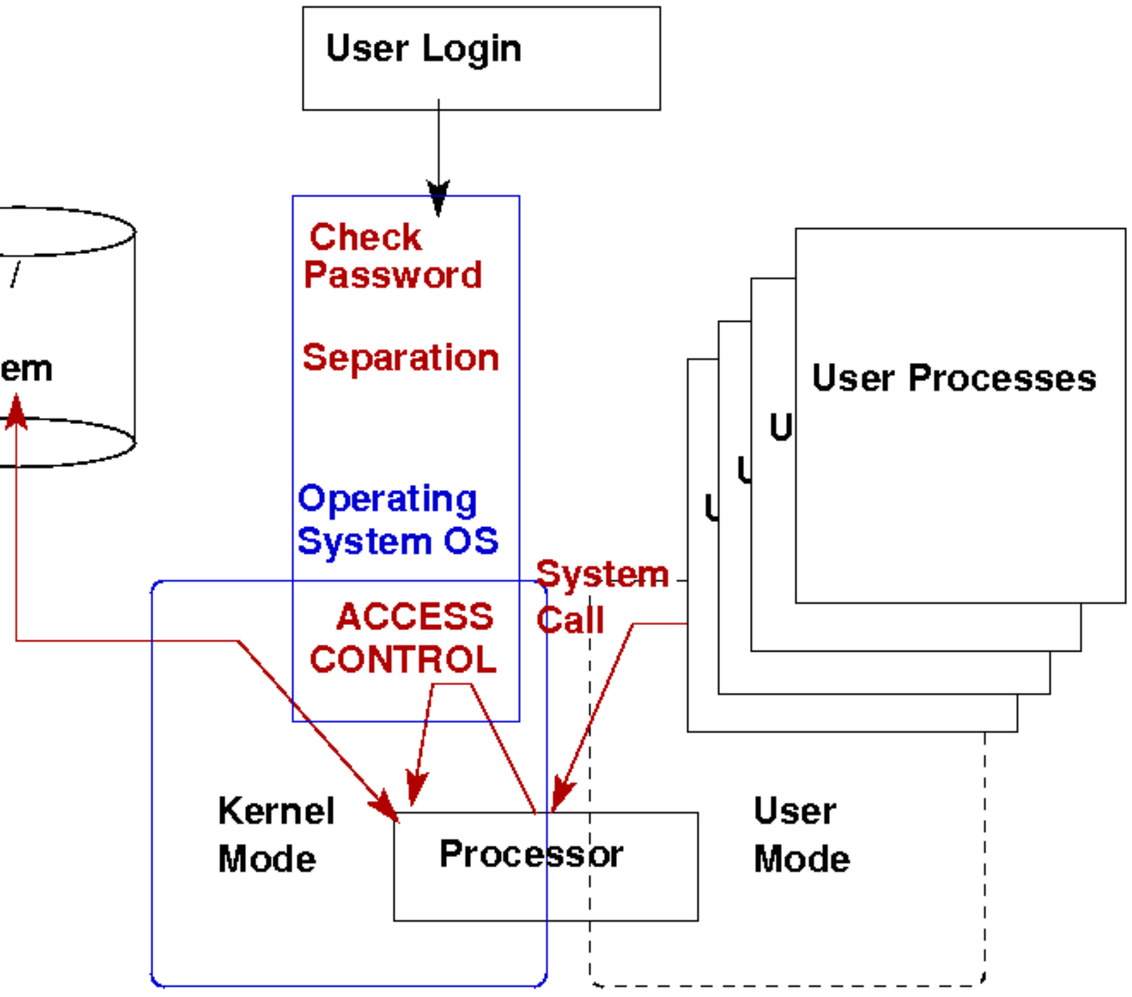
BUT AT LEAST THEY CAN'T INSTALL DRIVERS WITHOUT MY PERMISSION.

(xkcd.org)

A central question in cyber security is about who (persons, processes, devices, etc.) has access to which resources in the system.

Resources: read files, execute programs, change data-base content, share data with other principals, etc.

ACCESS CONTROL


User Login
Check Password
Separation
Operating System OS
ACCESS CONTROL
Disk / File System
System Call
User Processes
Kernel Mode
Processor
User Mode

# Access control on Operating System level

- Distinguish users, groups of users
- Controls access to files, ports, devices, and other resources
- User authentication (e.g. password, fingerprint)
- Allocate processes to users and enforce separation
- OSs can support complex policies for individual programs (e.g SELinux)

# Access control on application level

- This is what user usually can see (and also configure)
- Often complex security policies
- Enterprise applications: Staff with various roles, and fine-grained access to transactions
- Social networks: Rather complex rules on who can see, copy, forward, search what data.

# Facebook privacy settings

## Privacy Settings and Tools

| | | | |
|---|---|---|---|
| **Who can see my stuff?** | Who can see your future posts? | **Only Me** | Edit |
| | Review all your posts and things you're tagged in | | Use Activity Log |
| | Limit the audience for posts you've shared with friends of friends or Public? | | Limit Past Posts |
| **Who can contact me?** | Who can send you friend requests? | **Friends of Friends** | Edit |
| **Who can look me up?** | Who can look you up using the email address you provided? | **Everyone** | Edit |
| | Who can look you up using the phone number you provided? | **Everyone** | Edit |
| | Do you want search engines outside of Facebook to link to your profile? | **Yes** | Edit |

# Facebook page roles

| | Admin | Editor | Moderator | Advertiser | Analyst |
|---|---|---|---|---|---|
| Manage Page roles and settings | ✔ | | | | |
| Edit the Page and add apps | ✔ | ✔ | | | |
| Create and delete posts as the Page | ✔ | ✔ | | | |
| Send messages as the Page | ✔ | ✔ | ✔ | | |
| Respond to and delete comments and posts to the Page | ✔ | ✔ | ✔ | | |
| Remove and ban people from the Page | ✔ | ✔ | ✔ | | |
| Create ads | ✔ | ✔ | ✔ | ✔ | |
| View insights | ✔ | ✔ | ✔ | ✔ | ✔ |
| See who published as the Page | ✔ | ✔ | ✔ | ✔ | ✔ |

# Access control in enterprise applications

- Can enforce protection properties.
- Controls access to resources, data-bases, transactions, etc.
- Can be role-based (not just user-based)

# Example for access control policies

Lets look at a very simplified bookkeeping system.

It consists of:

- Operating System
- Accounts Program
- Accounting Data
- Audit Trail

Access rights:

- r permission to read
- w permission to write
- x permission to execute
- - no permission at all

# Naive approach

| User | OS | Account Program | Accounting Data | Audit Trail |
|---|---|---|---|---|
| Sam | rwx | rwx | rw | r |
| Alice | x | x | rw | - |
| Bob | rx | r | r | r |

# Refined approach

| User | OS | Accounts Program | Accounting Data | Audit Trail |
|---|---|---|---|---|
| Sam | rwx | rwx | r | r |
| Alice | rx | x | - | - |
| Acc Program | rx | r | rw | w |
| Bob | rx | r | r | r |

# Access control matrix

- Columns are access control lists ACLs
- Rows are capabilities (also tickets)

# ACL for Accounting Data

| User | OS | Accounts Program | Accounting Data | Audit Trail |
|---|---|---|---|---|
| Sam | rwx | rwx | r | r |
| Alice | rx | x | - | - |
| Acc Program | rx | r | rw | w |
| Bob | rx | r | r | r |

# Access control matrix

- Columns are access control lists ACLs
- Rows are capabilities (also tickets)

# Capabilities for Alice

| User | OS | Accounts Program | Accounting Data | Audit Trail |
|---|---|---|---|---|
| Sam | rwx | rwx | r | r |
| Alice | rx | x | - | - |
| Acc Program | rx | r | rw | w |
| Bob | rx | r | r | r |

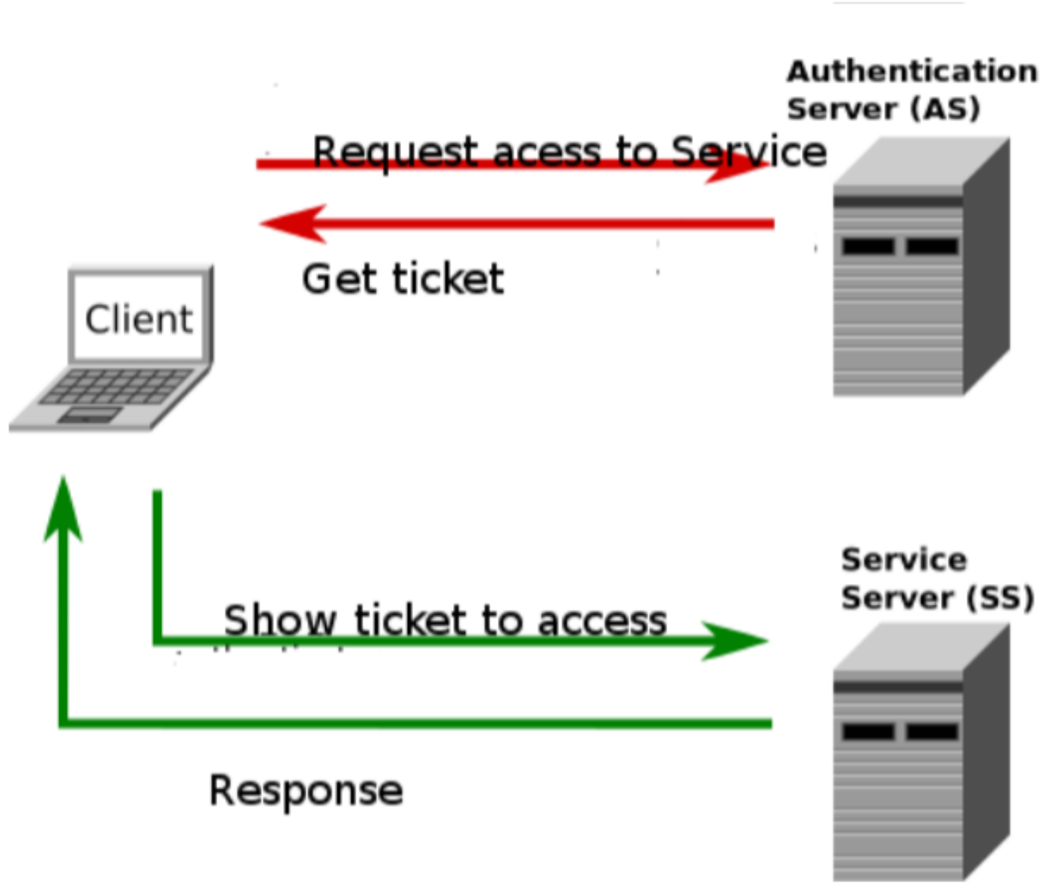
- Discretionary access control DAC: Depending on their rights, users can change ACLs and revoke or give rights to other users
- Mandatory access control MAC: A system (OS, Database management system) enforces pre-defined access policies

- Access control matrices / ACLs don't scale well.
- 1000 staff and 200 applications means 2 million entries to be managed.

- Use groups or roles to manage privileges of large sets of users.
- Role-based access control RBAC
- Example Healthcare: Doctor, Nurse, Administration, etc.

- ACLs are easy if users own their files and can manage access rights for these files.
- Difficult in distributed systems. Large and dynamic sets of users difficult to manage.
- Central authorities need to keep track of all resources that users have access to.
- Another option: use capabilities (rows in the matrix)

- Ticket or token-based access control.
- A central server controls access and issues tickets.
- Ticket contains capabilities (i.e. what is the user allowed to do)
- Example: Kerberos, Microsoft Active Directory


Authentication
Server (AS)
Request acess to Service
Get ticket
Client
Service
Server (SS)
Show ticket to access
Response

# Single sign-on

- Just log in once and access many services (e.g. Monash University authcate)
- Very convenient. High usability
- Single point of failure. Needs secure implementation and high level of control.

# Other types of access control

- Sandboxing encapsulates a process and restrict access to system outside of sandbox
- Example for sandboxing: Java Virtual Machine
- Virtualization can restrict access to shared resources. Mainly used for efficient use of resources, but also introduces level of access control.
- Examples: Xen, KVM, VMware

- Main goal of access control: limit the damage that can be done by users, groups of users.
- Privilege escalation is a goal for attacks
- Many ways how access control can go wrong

# What can go wrong?

- Weaknesses in software, interfaces, protocols
- Physical attacks
- Race conditions, feature interaction problems
- Connect devices (USB)
- Social engineering

# How to authenticate a person?

- Identify at login
- Authenticate particular transactions

- The Password is the most commonly used way
- Multi-factor authentication combines different ways of authentication

(Wikimedia Commons)

# Biometrics

- Fingerprints
- Voice recognition
- Iris scans
- Others

- Biometrics have high usability
- Not really secret information
- Cannot be revoked/replaced
- No pseudonymous/anonymous access

# Hardware Tokens

- Separate device / additional security
- Computer can still be attacked

# Authentication of Transactions

- E.g. for money transfer in banking
- Tansaction numbers (TANs) are not linked to actual transaction
- SMS TAN can show info on transaction. Two devices might need to be manipulated.
- TAN generator reads barcode from screen and generates TAN linked to transaction.

# Additional security mechanisms

- Hard disk encryption
- Virus protection
- Backups
- Security updates
- Trusted Computing (special security hardware)