Contents

1 Computer security 2

2 Confidentiality 2

3 Integrity 2

3.1 Integrity vs Authenticity . . . 2

4 Availability 2

5 Accountability 2

6 Data vs information 3

7 Security Design principles 3

7.1 Focus of Control . . . 3

1 Computer security

- Security is about the protection of assets. Assets might be physical, but they could also be data, information, even ideas
- Protection **measures**: prevention, detection, recovery manual or automatic
- Usually defined as three key areas (CIA):
  - **Confidentiality**: prevention of unauthorised disclosure of information
  - **Integrity**: prevention of unauthorised modification of information
  - **Availability**: prevention of unauthorised witholding of information or resources

2 Confidentiality

- The prevention of **unauthorised** users reading private or secret information.
- Privacy The protection of personal data
- Examples: medical records, transfer or credit card details

3 Integrity

- The prevention of unauthorised **modification** of data, and the assurance that data **remains unmodified**.
- Examples: distributed bank transactions, database records

3.1 Integrity vs Authenticity

- Just because we have *integrity*, doesnt mean we have *authenticity*.
- Can we verify the sender? Does it have *freshness*?
- Authenticity is integrity and freshness combined

4 Availability

- The property of being accessible an useable **upon demand** by an authorised entity
- In other words, we want to *prevent denial of service* (DoS)
- Examples: redundant power supplies, firewall packet filtering

5 Accountability

- Users should be held responsible for **their actions**
- The system should **identify and authenticate** users and ensure compliance
- Audit trails must be kept
- Non-repudiation: a situation where a statement's author cannot **successfully dispute its authorship** or the validity of an associated contract. Provides un-forgeable evidence that someone did something

6 Data vs information

- Security can be seen as controlling access to information
- This is hard, we usually control access to data instead
  - Data A means to represent information
  - Information An **interpretation** of that data
- Focusing on data can still leave information vulnerable:
  - Mikes criminal record not found in the database.
  - You do not have permission to access Mikes criminal record.
  - With the second message it can be known that Mike has a criminal record.

7 Security Design principles

7.1 Focus of Control

In a given application, should the focus of protection mechanisms be

- Data: Permitted manipulation of data e.g. consistency check
- Operations: Permitted invocations e.g. `transfermoney()`
- Users: Permissions for specific users e.g. /home/name/

7.2 Complexity vs. Assurance

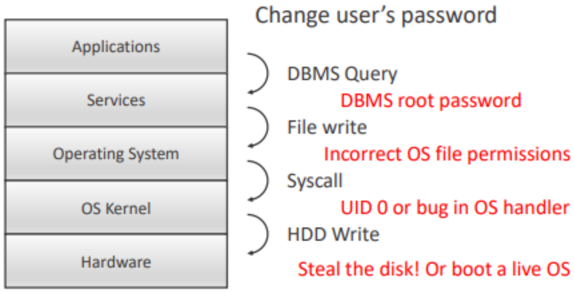
Would we prefer a simple approach with high assurance, or a feature-rich environment? Feature-rich security systems and high assurance do not match easily. E.g. Linux vs Windows permissions. To achieve a high degree of assurance, the security system has to be examined in close details and exhaustively as possible. Hence, there is a trade-off between complexity and assurance.

7.3 Decentralised Controls

- Should defining and enforcing security be performed by a central entity, or be left to individual components in a system?
- Central Entity Easy to achieve uniformity, but a possible bottleneck
- Distributed Solution More efficient, but harder to manage

7.4 Layered Security

- We can visualise our security model in layers
- Each layer protects a boundary, and relies on the security of the layers below
- Consider an application using a database:



Reference section

freshness

Implies that the sensed data are recent, and it ensures that no adversary replayed old messages