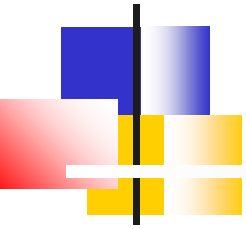
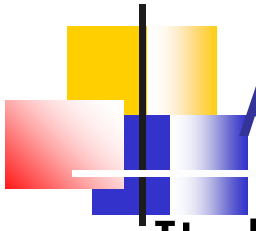


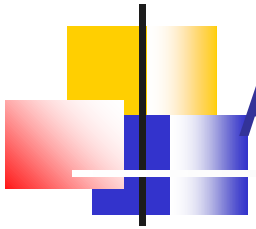
Asset Analysis



Asset Analysis -I

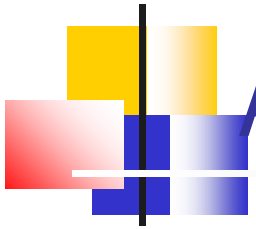
- It discovers the assets that result in an impact (a loss for the organization) if successfully attacked
- This implies to discover which ICT resources an organization needs to work in an efficient way

1. Discover the fundamental business processes
2. Critical ICT resources for these processes
3. The impact for the organization if
   - A business process is stopped (resource integrity or availability)
   - The resource has to be rebuilt ex novo (integrity)
   - The attacker discovers the information in the resource (confidentiality)



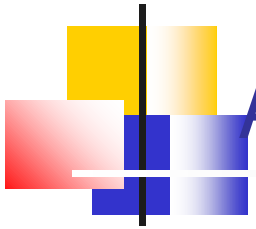
Asset Analysis -II

- Physical and Logical ICT Resources
  - Databases
  - Applications to access the database and compute the outputs of interest (may be even more important than the database i.e. application using pubblic data)
  - Computational power
  - Communication bandwidth



Asset Analysis -III

- Physical Resources
  - With the IoT and Industrial Control Systems, ICSs, a cyber attack can affect resources controlled by ICT networks
  - A production line can be stopped by a cyber attack
  - A large number of assets



Asset Analysis -IV

- Approximating the value of a resource is not trivial
- A possible heuristics consider the cost of rebuilding the resource if it disappears
- An asset analysis is useful non only for security reasons but because it returns an inventory with the existing resources that should be protected
- The first of any set of principles to evaluate and manage ICT risk always requires to build an inventory of all the resources in the system to be protected (they include ICT resources)

The cost of malicious cyberactivies in USA (Feb. 2018)

Attackers

a) **Nation-states**: Russia, China, Iran, and North Korea. These groups are well funded and often engage in sophisticated, targeted attacks.

b) **Corporate competitors**: Firms that seek illicit access to proprietary IP, including financial, strategic, and workforce-related information on their competitors.

c) **Hacktivists**: Private individuals or groups with a political agenda and seek to carry out high-profile attacks. Attacks help hacktivists distribute propaganda or to cause damage to opposition organizations for ideological reasons.

d) **Organized criminal groups**: These are criminal collectives that engage in targeted attacks motivated by profit seeking.

e) **Opportunists:** Usually amateur hackers driven by a desire for notoriety.

f) **Company insiders**: These are typically disgruntled employees or ex-employees

The cost of malicious cyberactivies in USA (Feb. 2018)

- Malicious cyber activity cost between $57 and $109 billion in 2016.
- These activities target private and public entities and manifest as denial of service attacks, data and property destruction, business disruption for collecting ransoms, theft of data, intellectual property, financial and strategic information.
- Damages from cyberattacks and cyber theft may spill over from the initial target to economically linked firms, magnifying the damage to the economy
- Firms share common cyber vulnerabilities, causing cyber threats to be correlated across firms. The limited understanding of the correlation among these common vulnerabilities impedes the development of the cyber insurance market.
- Scarce data and insufficient information sharing impede cybersecurity efforts and slow down the development of the cyber insurance market and prevent risk transfer
- Lax cybersecurity imposes negative externalities on other economic entities and on private citizens. Failure to account for these externalities results in underinvestment in cybersecurity by the private sector relative to the socially optimal level of investment.

Externalities

- An externality is a cost or benefit incurred or received by a third party that has no control over the creation of that cost or benefit.
- It can be both positive or negative. Costs and benefits can be both private—to an individual or an organization—or social, as it can affect society as a whole.
- Security Externality:
  - Unprotected computers may be used to attack other computers. There is a lack of incentive for each user to adequately protect against viruses, since the cost of virus is borne by others. This is a positive "externality." Such settings lead to a classic free-rider problem, individuals will choose less security than the social optimal. If I increase the protection of my computer, I enhance the security of other users as well as my own.
  - Network Effects may contribute to security problems. Large networks are more vulnerable to security breaches, because of the success of the network. In part because of its large installed base, more people are searching vulnerabilities in Microsoft's Internet Explorer, hence it is more vulnerable than a less popular browser.

Free riding

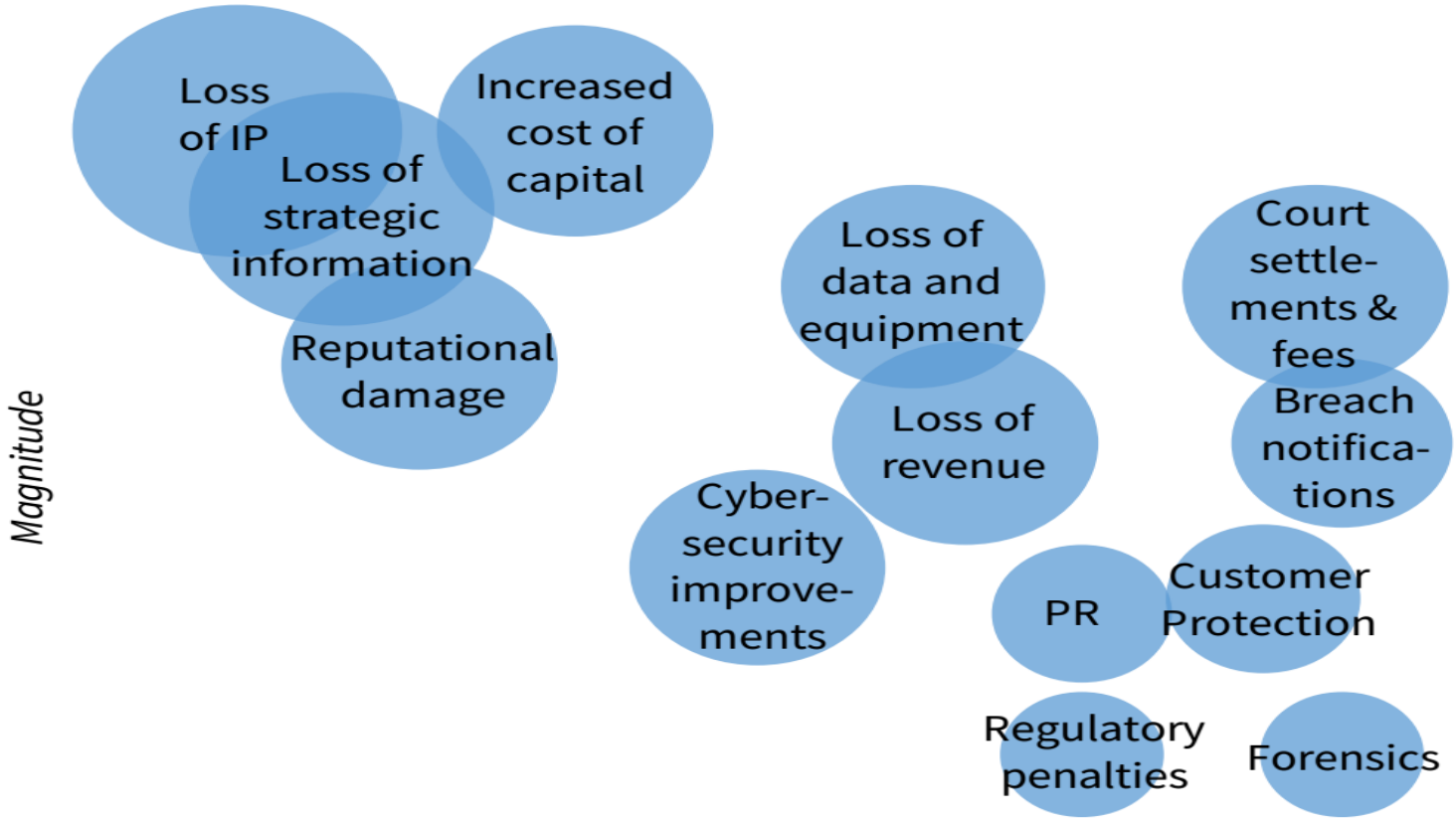
- Security is a pubblic good depending on the effort of many individuals. Voluntary provision of public goods may result in free riders: individuals may tend to shirk, resulting in an inefficient level of the public good.
- How much effort each individual exerts depend on his own benefits and costs, the efforts by the other individuals, and the technology that relates individual effort to outcomes.
- We distinguish three prototypical cases.
  - Total effort. Security depends on the sum of individual efforts.
  - Weakest link. Security depends on the minimum effort.
  - Best shot. Security depends on the maximum effort.

Free riding: results

- Agents with a benefit from security and that pay a cost, success is probabilistic
- total effort: security is determined by the agent with the highest benefit-cost ratio. Other free ride on this agent.
- weakest-link case, security is determined by the agent with the lowest benefit-cost ratio
- Systems will become increasingly secure as the number of agents increases in the total efforts case, but they are increasingly unreliable as the number increases in the weakest link case (random agents)

The cost of malicious cyberactivies in USA (Feb. 2018)

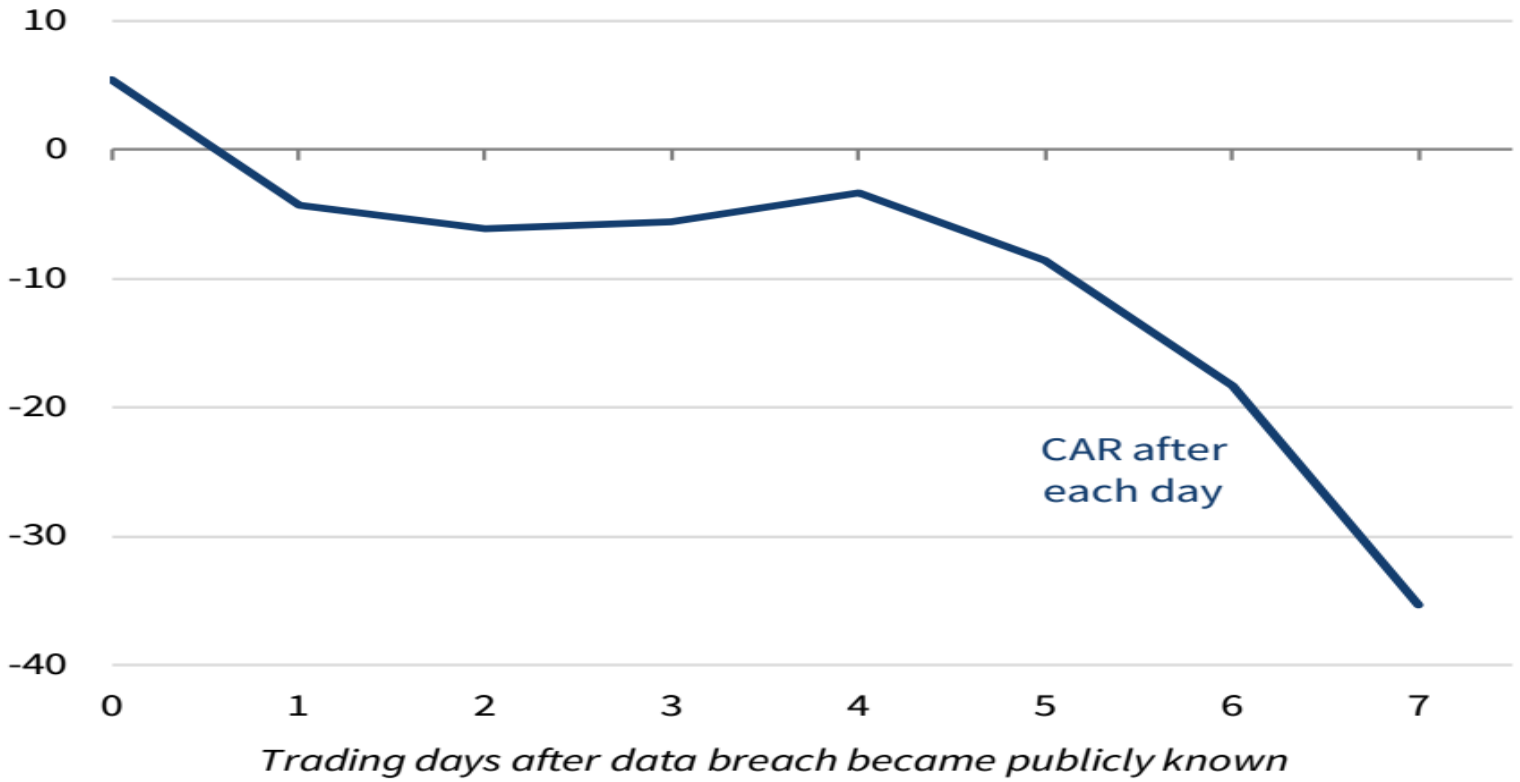
Figure 1. Cost Components of an Adverse Cyber Event

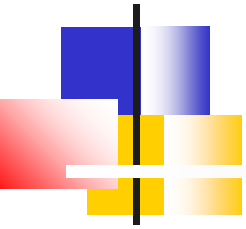


The cost of malicious cyberactivies in USA (Feb. 2018)

Figure 5. SolarWorld's Cumulative Abnormal Return After Its Data Breach Became Publicly Known

(CAR, percent)



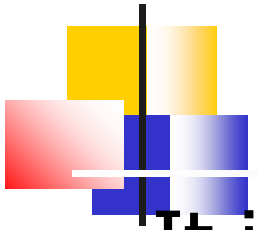


Security Policy

Security Policy

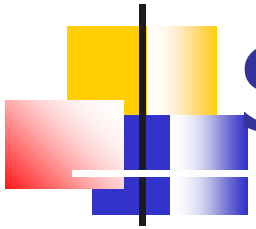
A set of rules that an organization adopts both to minimize cyber risk and to define the goals of security

- Defining the goal of security = the assets and the resources to protect in order to protect the assets
- Defining the correct behavior of all the users
- Forbidding dangerous behaviors and components
- It implies the definition of
  - System architecture
  - Catalogue (inventory) of components and of applications
  - Users (rights and constrains)
  - Administrators (rights and constrains)
  - Legal use of the resources
  - Who has to verify that the policy is applied
  - What happens if the policy is violated



Security Policy

- It is critical because it defines
  - The goals and the assets of an organization
  - Legal behaviour for each class of users
  - Whether components can still have some vulnerabilities and how they should be used
  - Rules to manage both human and ICT resources
  - Roles and responsibility
- The security policy cannot violate the legislation that concerns ICT systems



Subject and object

- A more abstract definition of a policy represents user and resources in an abstract way in terms of objects to define which operations that users can apply
- A subject is any entity that can invoke the operations an object defines
- An object that invokes some operations defined by other objects is both a subject and an object
- The implementation of subjects and objects depends upon the implementation level (e.g. the VM) of interest

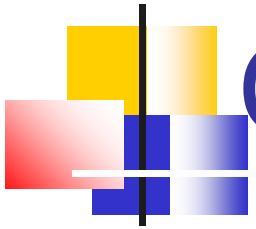
Subject = user, application, program, process, thread, instruction …

Object = instance of an abstract data type, procedure or function, variable, logical or physical resources

Rights

- A subject entitled to invoke an operation of an object owns a right on this object
- Rights are directly or indirectly deduced from the security policy
  - Direct = S can read the file F then *S owns a read right on F*
  - Indirect = since S can read F then then any program P that S execute can read the memory segment MS that stores a record of F then *P owns a read rights on MS*

    = the right of P on MS is deduced from those of S on F



Objects, operations and types

- The specification of an object with its operations defines (implements) a data type
- A type system can allow only those invocations of an operation on an object that are entitled by the policy
- However *dynamic controls cannot be avoided* due to vulnerabilities in the compiler or in the run time support that result in run time behavior that differs from the expected one according to the specifications

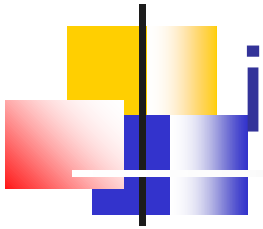
Reflections on Trusting Trust

To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.

KEN THOMPSON

The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.) No amount of source-level verification or scrutiny will protect you from using untrusted code. In demonstrating the possibility of this kind of attack, I picked on the C compiler. I could have picked on any program-handling program such as an assembler, a loader, or even hardware microcode. As the level of program gets lower, these bugs will be harder and harder to detect. A well-installed microcode bug will be almost impossible to detect. After trying to convince you that I cannot be trusted, I wish to moralize. I would like to criticize the press in its handling of the "hackers," the 414 gang, the Dalton gang, etc. The acts performed by these kids are vandalism at best and probably trespass and theft at worst.

Security Policies: a first important classification

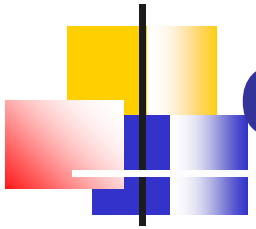


- Default allow = it defines forbidden behaviours and *allows anything that it does not define* = enumerating badness
- Default deny = it defines legal behaviors and *forbids anything it does not define* eg anything else
- Default allow is very dangerous =anytime we forget to enumerate a bad behavior enumerating badness does not work

An analogy

- Default allow = defines a set S by describing those elements that do not belong to S = the complement of S

- Default deny = defines a set by describing the elements that belong to S



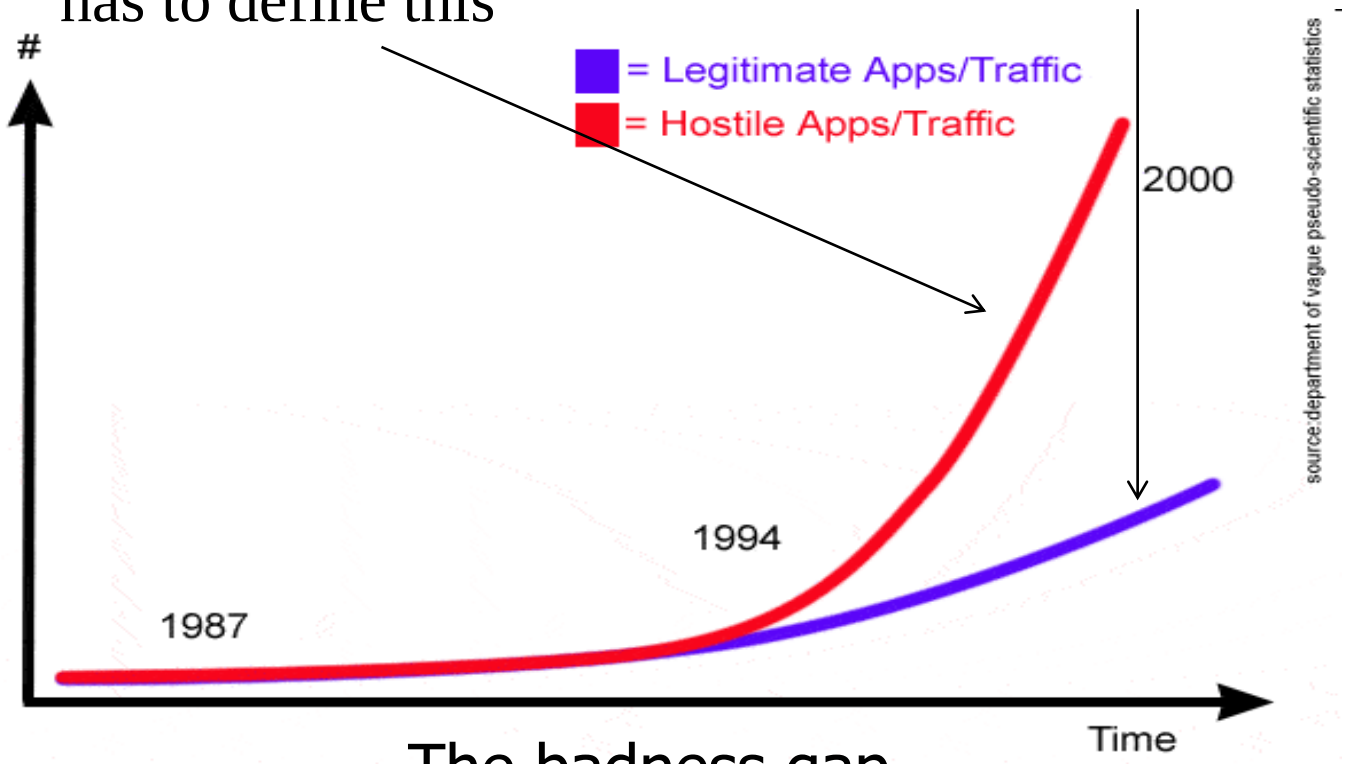
The Six Dumbest Ideas in Computer Security (M.Ranum)

1. Default Permit (default allow)
2. Enumerating Badness
3. Penetrate and Patch
4. Hacking is Cool
5. Educating Users
6. Action is Better Than Inaction

Enumerating Badness

A default deny policy defines this

A default allow policy has to define this



The badness gap

A modular security policy

A security policies includes the followings 9 policies:

1. Acceptable Use Policy (AUP)

   It stipulates the constraints and practices that an employee using organizational assets must agree to in order to access to the network or the internet.

2. Access Control Policies (ACP)

   It outlines the access available to employees in regards to an organization's data and information systems. It covers standards for user access, network controls, operating system controls, passwors complexity, methods for monitoring how corporate systems are accessed and used, how access is removed when an employee leaves the organization.

A modular security policy

3. Change Management Policy

It refers to a formal process for making changes to IT, software development and security services/operations. It increases the awareness and understanding of proposed changes across an organization, and to ensure that all changes are conducted methodically to minimize any adverse impact on services and customers.

4. Information security policies

The critical one, defined in the following

A modular security policy

5. Incident Response Policy

The incident response policy is an organized approach to how to manage an incident and remediate the impact to operations. The goal of this policy is to describe the process of handling an incident with respect to limiting the damage to business operations, customers and reducing recovery time and costs.

6. Remote Access Policy

It defines acceptable methods of remotely connecting to an organization's internal networks. It includes addendums with rules for BYOD assets. It is a requirement for organizations that have dispersed networks that extend into insecure network locations, such as coffee house or unmanaged home networks.

A modular security policy

7. Email/Communication Policy

It formally outlines how employees can use the business' chosen electronic communication medium. It covers email, blogs, social media and chat. It provides guidelines on what is considered the acceptable use of any communication technology.

8. Disaster Recovery Policy

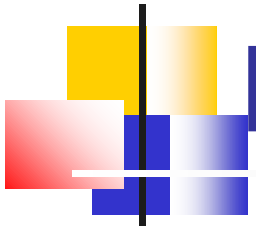
It is a part of the business continuity plan. If an event has a significant business impact, the BCP will be activated.

9. Business Continuity Plan (BCP)

The BCP will coordinate efforts across the organization and will use the disaster recovery plan to restore hardware, applications and data deemed essential for business continuity.

Classes of ACPs

- Discretionary access control
  - An owner exists for each object (info, record, app)
  - The owner defines (has the right and the burden of defining)
    - The subjects can operate on the object (need to access)
    - The rights for each subject = the operations it can invoke
- Mandatory access control
  - There is an owner but there are some system wide rules it has to satisfy = it cannot violate

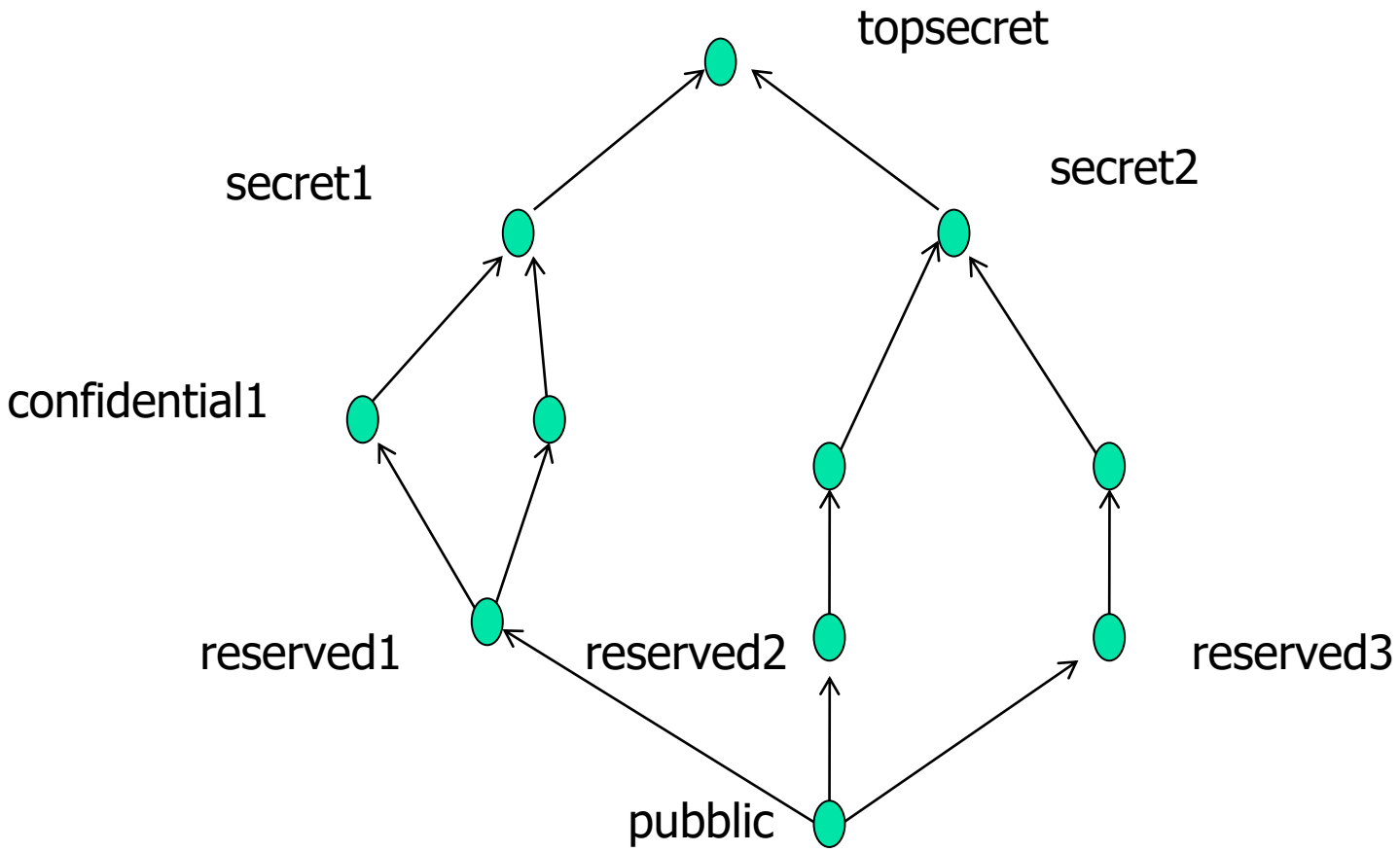


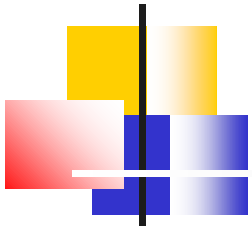
Mandatory Access Control

- All the objects are partitioned into classes
- All the subjects are partitioned into classes
- The same classes for object and subjects

  (not strictly required but it simplifies everything)
- All the classes are partially ordered
- A subject may be granted the right to invoke an operation only if the classes of the subject and of the object satisfy a predefined condition

Partial Order



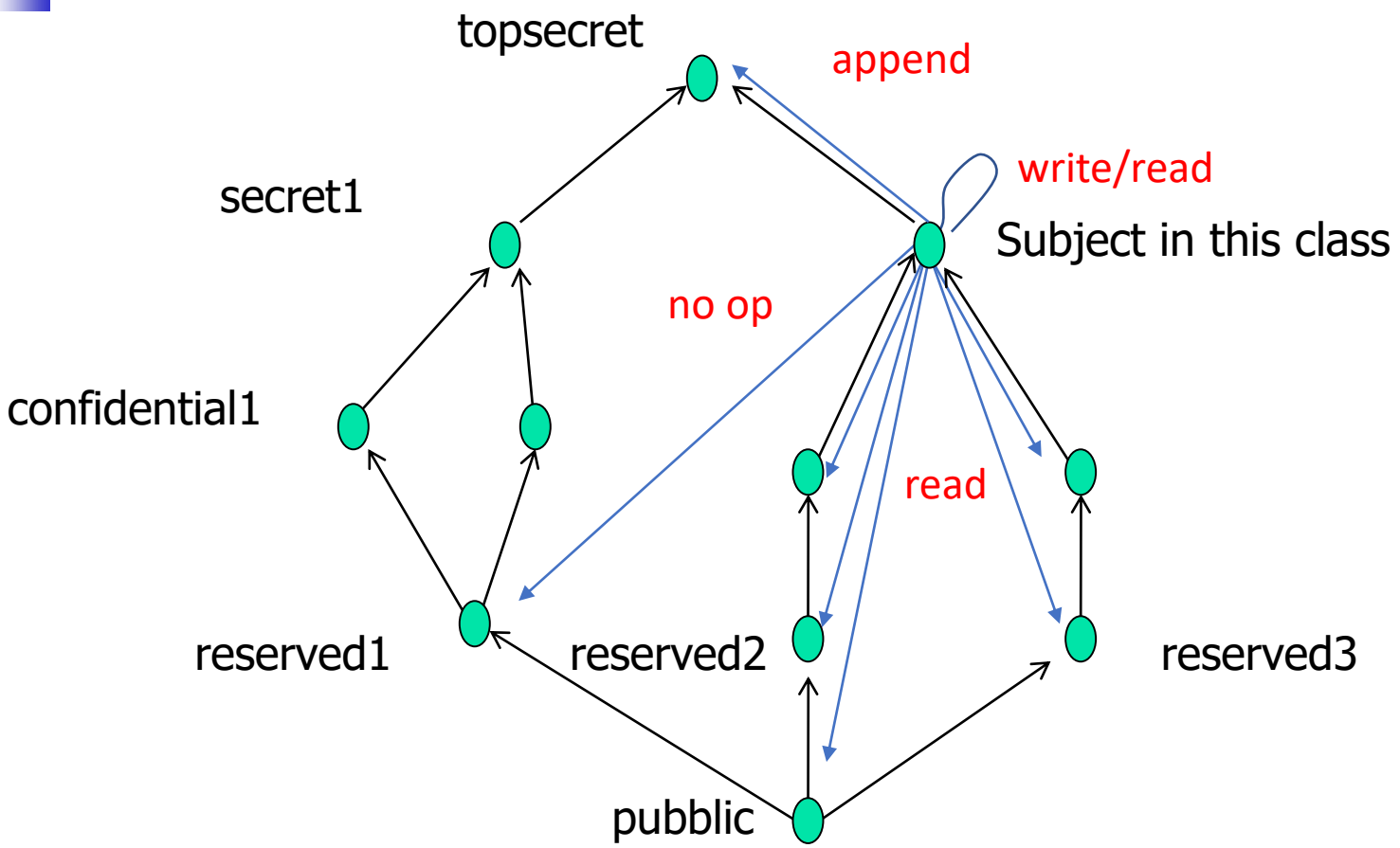


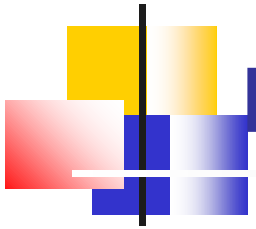
MAC information flow - I

- Object = file
- Operations = read/write/append
- A subject in a class C may be enabled to
  - Read any file with a class lower than or equal to C
  - Write any file with a class equal to C
  - Append a record to a file with a class larger than C
  - The owner of the file can grant the rights provided that the three previous rules are satisfied

This policy prevents loss (leaks) of information
(No write down)

MAC information flow confidentiality



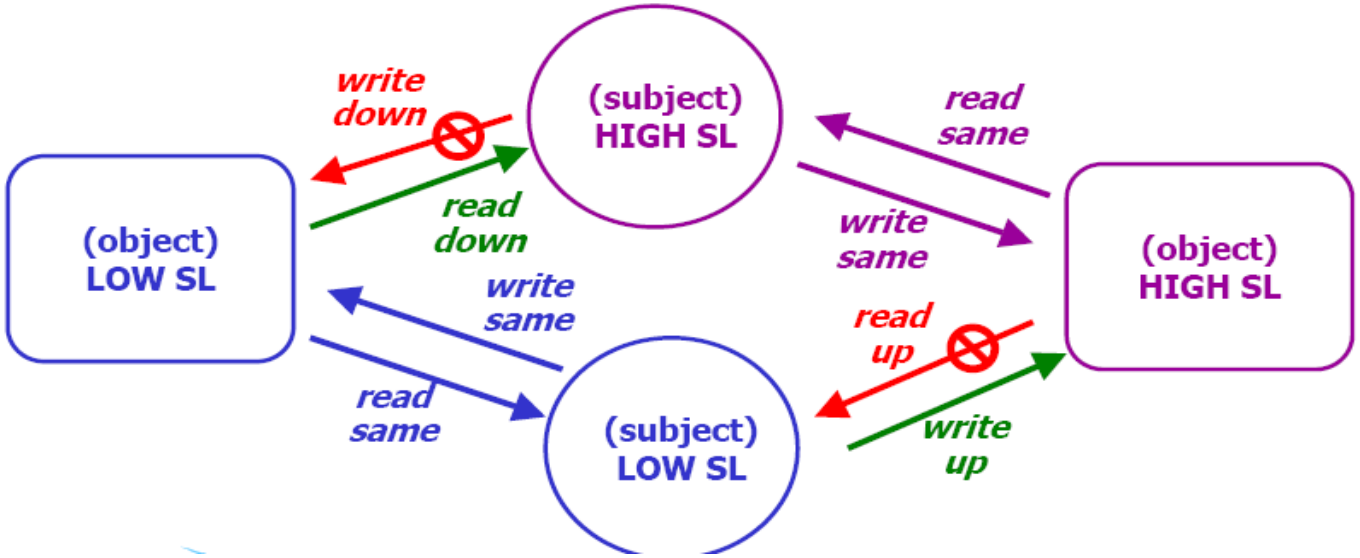


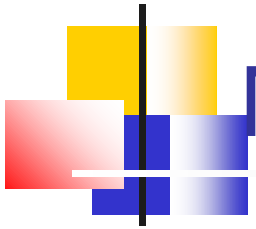
no write down

- Prevents an information flow (leakage) from an high level object to those with a lower level
- Guarantee confidentiality of information
- As a counterpart, the amount of information with a higher level increases because the information level cannot decrease and updates can occur at not lower levels
- A further operation is introduced to periodically desecretate information to the lower levels
- This operation is the ideal target for an attacker

Mandatory Access Control - I

- Bell-LaPadula Policy (multilevel security)
  - access control attributes:
    - hierarchical security level
    - set of non hierarchical categories
  - fixed rules: "no read up, no write down"

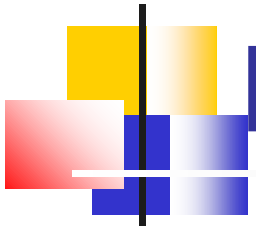




MAC information flow - II

- Object = file
- Operation= read/write
- A subject in class C may be enabled to
  - Write any file with a class lower than or equal to C
  - Read any file with a class larger than or equal to C

Integrity is privileged (No write up)



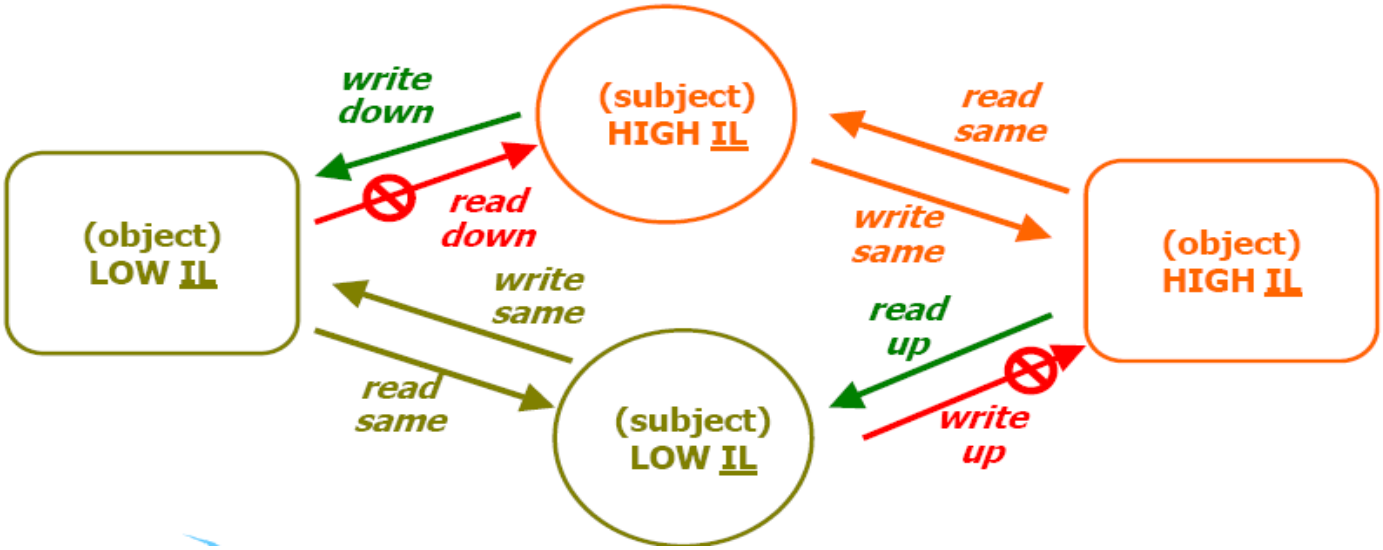
No write up

- A low integrity subject cannot update an highly integrity object to prevent loss of integrity
- Integrity is privileged at the expence of confidentiality
- You can know but not change the parameters to control a nuclear plant

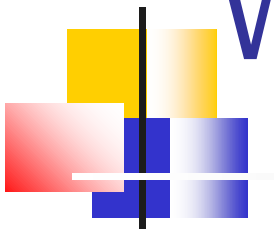
Mandatory Access Control - II

(multilevel security)

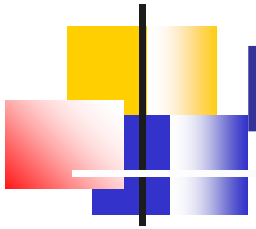
- Biba Integrity model
  - access control attributes:
    - hierarchical integrity level
    - set of non hierarchical integrity categories
  - fixed rules: "no write up, no read down"
    - exact opposite of BLP/multilevel security



Watermark



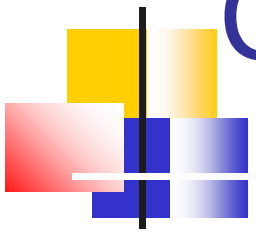
- The level of a subject has maximum but is not fixed, it is the highest one of the objects it has worked on
- To protect confidentiality, the level increases has the subject reads critical information
- Monotonic increase of the level, after a given level has been reached no decrease is possible
- Time dependent MAC policy
- Introduced to minimize the flowing of information at higher levels



No interference property

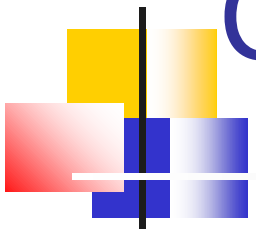
- Each object and each subject is paired with a label that defines the corresponding level
- An object label is updated at run time according to both
  - the operations that are invoked
  - the level of the subject invoking the operations
- A system satisfies the *no interference principle* if the labels paired with an object do not change even after removing subjects with different, i.e. a lower or an higher, level from the system (Bell-LaPadula/Biba)
- No information
  - leakes from the higher levels
  - can affect objects with a higher level

Clark -Wilson -1



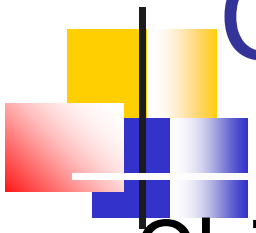
- A policy in this class defines
  - A set of consistency contrains each on some objects
  - Some sequences of operations on the objects (well formed transactions) that do preserve any consistency constrains
- If only these sequences are invoked, then the system evolution only navigate across states that satisfies the consistency constrains

Clark -Wilson -2



- Each well formed transaction is atomic, either is completed or it is undone
- Atomicity may be implemented by managing a backup copy of each of the involved objects
- It is the user responsibility to prove that each transaction is well formed, e.g. it does not violate the consistency constrains

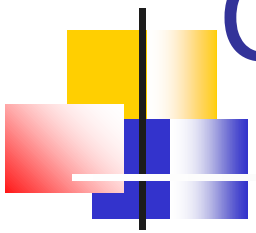
CW- Example



- Objects = Bank accounts
- Constrain
  1. If money is transferred betwen two accounts, their sum does not change = we add to an account the amount we withdraw from the other
  2. We record the amount of money cashed and the one that has been withdrawned from the accounts
  3. At the end of each day

  sum of the accounts = (cashed) - (withdrawals) +

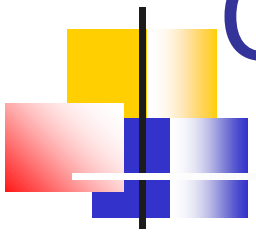
  (sum of the accounts at the beginnging of the day)
- Any transaction must be atomic



Chinese Wall

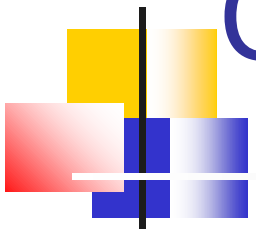
- Objects are partioned into classes
- As soon as a subject invokes an operation on an object
  - cannot invoke operations on objects in distinct classes
  - can invokes operation on objects in the class
- Avoid conflict of interest
- Time dependent
- Can be integrated with a MAC/DAC policy

Overall Policy – I



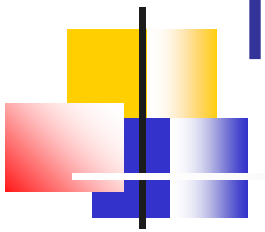
- A real policy can merge several of the previous policy
- As an example
  - No write down
  - Chinese wall
- We have rules that define which objects can be read and other that forbid the access to some other objects

Overall Policy – II



- Distinct policies can be applied to the same object/subject
- There are two levels for a subject, one for confidentiality and one for integrity
  - Some objects consider the confidentiality level (no write down)
  - Some objects consider the integrity level (no write up)

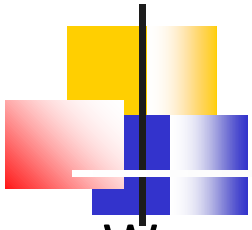
Trusted Computing Base



- TCB includes any component that is involved in the implementation of the security policy
- These components are highly critical because any bug in a TCB component is, almost always, a vulnerability
- Any system needs to trust all the TCB components
- Assurance of these components is very important

Size of the TCB

- The security level of a system and the trust in it increases as the size of the TCB decreases
- Correctness of a small TCB can be proved by applying formal method and this results in a high assurance level
- An important criteria to select one of a set alternative implementations of the same policy



All together now ...

- We can define
  - important resources by looking at process of the organization
  - subjects and objects in terms of these resources
  - rules on the resource usage and map them into rights
    - Default allow
    - Default deny
- Two frameworks for policy Mac (system wide cons)/Dac
- Integrity vs Confidentiality
- Static or watermark
  - Dac (no global constrain)
- Data Types + Run time check = Trusted Computing Base
- Size of TCB important for security