Introduction

INFORMATICS AND ORGANIZATIONAL SECURITY

Security objectives (1/3)

Defense against catastrophic events

- Natural phenomena
  - Abnormal temperature, lightning, thunder, flooding, radiation, …
- Degradation of computer hardware
  - bad sectors in disks, failure of power supplies, bit errors in RAM cells, etc.

Solution:

- Realistic prevention
  - Mainly for the most probable events
- Information backup
- Replication
  - Information
  - Computational resources

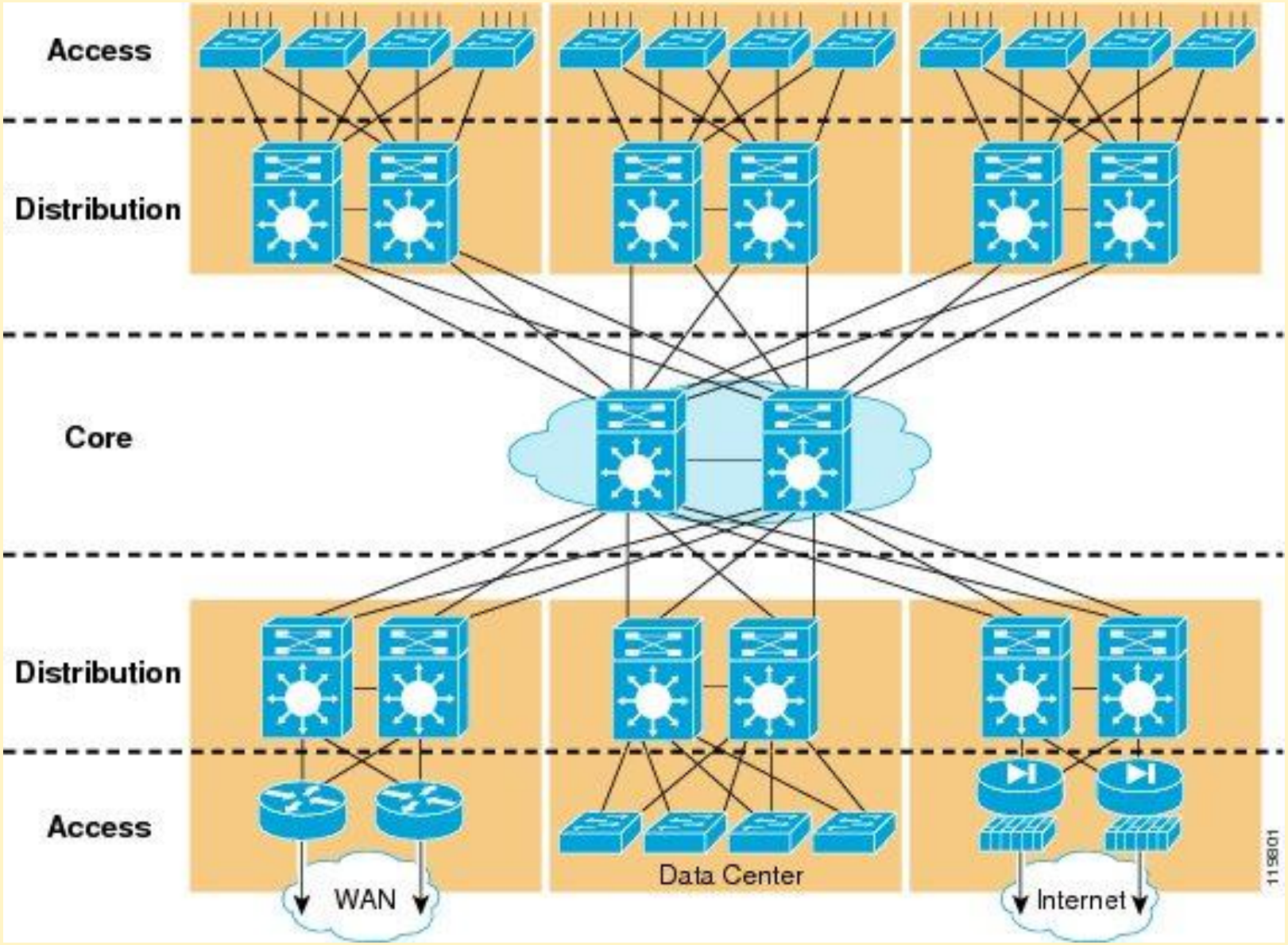
Security objectives (2/3)

Defense against ordinary faults / failures

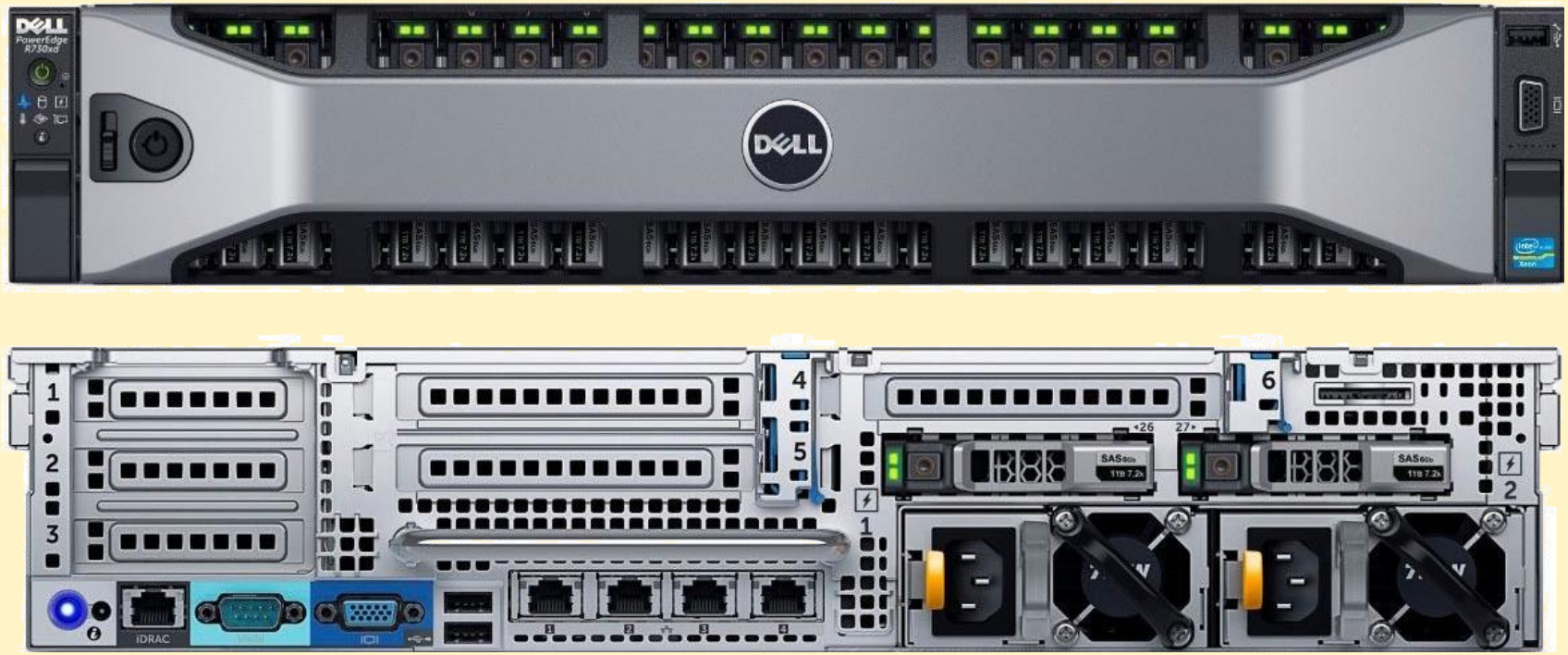
- Power outages
- Systems' internal failures
  - UNIX panic, Windows blue screen, OS X panic
  - Deadlocks
- Software faults / Communication faults...

Solution:

- Redundant / alternative power supplies (and fan, disks, links...)
- Transactional systems
- Dynamic Routing, re-transmission



Source: CISCO



Source: DELL

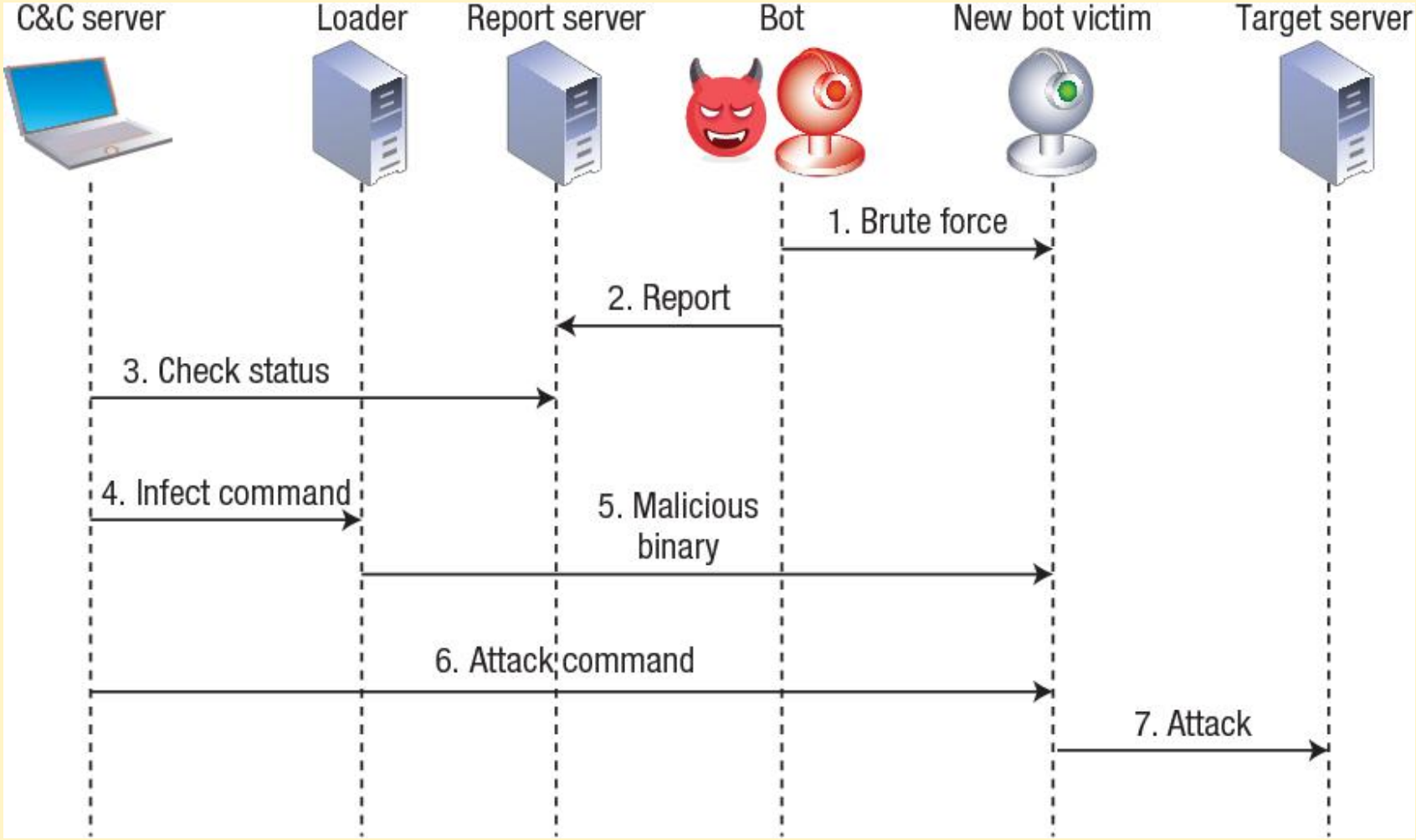
Security objectives (3/3)

Defense against non-authorized activities (adversaries)

- Initiated by someone “from outside” or “from inside”

Types of non-authorized activities:

- Information access
- Information alteration
- Resource usage
  - CPU, memory, print, network, etc.
- Denial of Service
- Vandalism
  - Interference with the normal system behavior without any benefit for the attacker



Mirai botnet operation and communication.
Mirai causes a distributed denial of service (DDoS) to a set of target servers by constantly propagating to weakly configured Internet of Things (IoT) devices.

source: Kolias, Constantinos et al. “DDoS in the IoT: Mirai and Other Botnets.” Computer 50 (2017): 80-84.

Security in computing systems: Complex problems

Computers can do much damage in a short time frame

- Computers manage huge amounts of information
- Process and communicate with very high speed

The number of weaknesses is always growing

- Due to the increased complexity
- Due to every reducing time-to-market

Networks allow novel attack mechanisms

- “Anonymous” attacks from any place in the planet
- Fast spread across geographical boundaries
- Exploitation of insecure hosts and applications

Users are mostly unaware and careless

- They do not know the problems, impact, good practices and solutions
- Because they take risks / do not care / do not estimate the risk correctly

The pragmatic approach

Perfect protection of a system is impossible

- Cost-efficiency balance
- Problem: determine cost and efficiency

Security is expensive

- Dedicated technology, additional resources, skilled professionals, processes
- Solution: Deploy the minimum required

Protection, value and punishment

- Protection that is “good enough” for the most frequent attacks
- Less interference with daily work than the damage caused by the attackers
- Use the policy and courts to track and prosecute attackers
  - It is important in any society or organization to avoid the notion of total impunity

Lexicon

Vulnerability

- A system weakness that makes it sensible to attacks
- Can be present in its design, development, installation, operation or maintenance

Attack

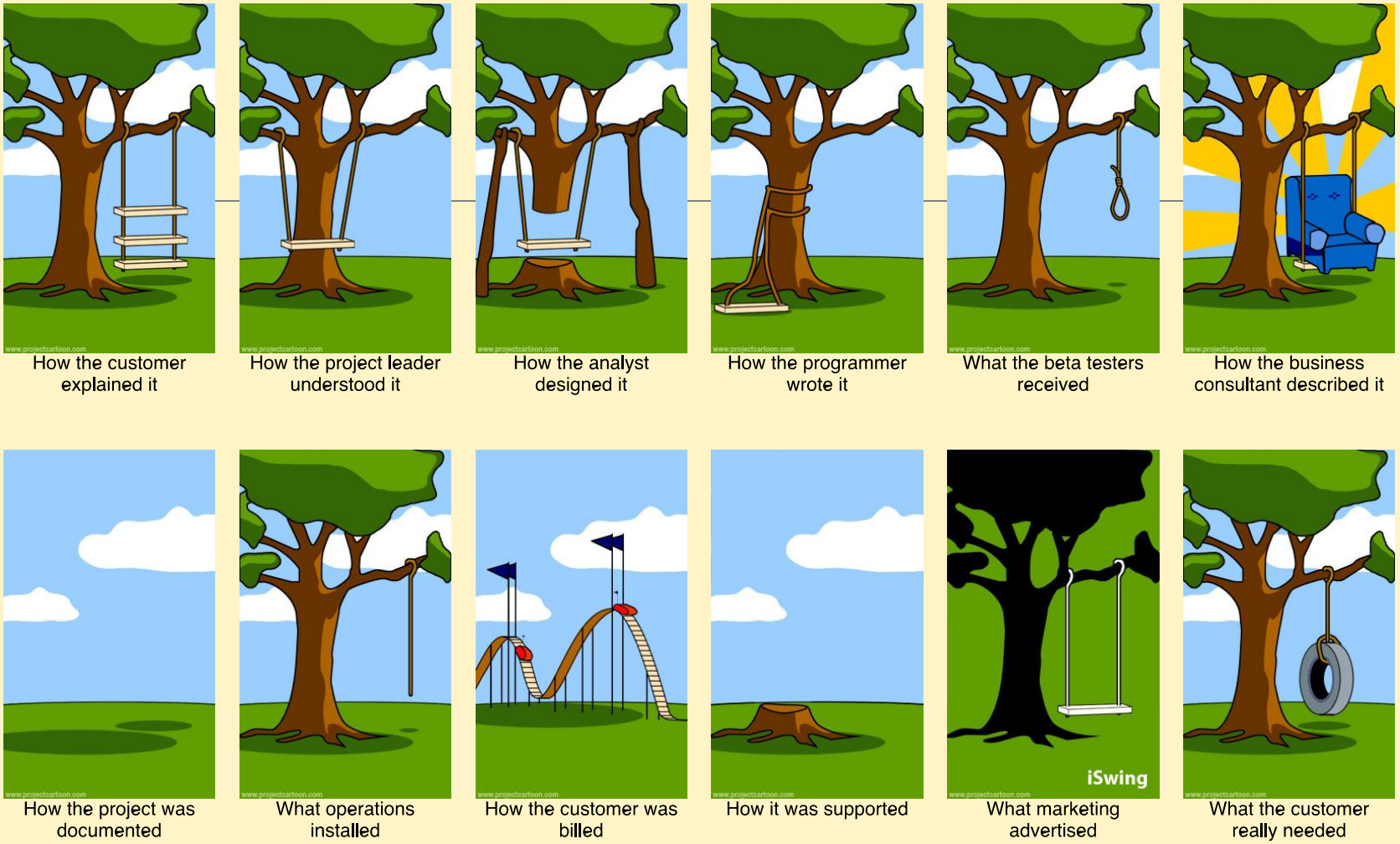
- A set of actions that lead to the execution of illegal activities
  - Attacks usually explore vulnerabilities

Risk / threat

- The possible damage resulting from an attack

Defense

- Set of policies, mechanisms and technologies aiming at
  - Reduce the amount of vulnerabilities
  - Detect as fast as possible actual and past attacks
  - Reduce the risks of the systems



Project development is complex, crosses several expert domains, vulnerabilities can be introduced in every point

Security risks

Information, time e resources (money)

- Destruction or tampering of information

Confidentiality

- Non-authorized access to information

Privacy

- Non-authorized gathering of personal information
- Data warehousing (or distribution) of personal information

Resources Availability

- Disruption of systems, communications, processes

Impersonation

- Non-authorized exploitation of personal accounts /profiles
  - accounts related to people, services, entities, etc...

Main vulnerability sources

Hostile applications or bugs in applications

- First: Morris Worm 1988: Buffer overflows + discovery of weak passwords
- root kits, virus, worms, offensive tools

Users

- Ignorant or careless
  - telnet vs. ssh, IMAP vs. IMAPS, HTTP vs HTTPS
  - False sense of security (I have an anti-virus, so I'm protected!)
- Hostile

Defective administration

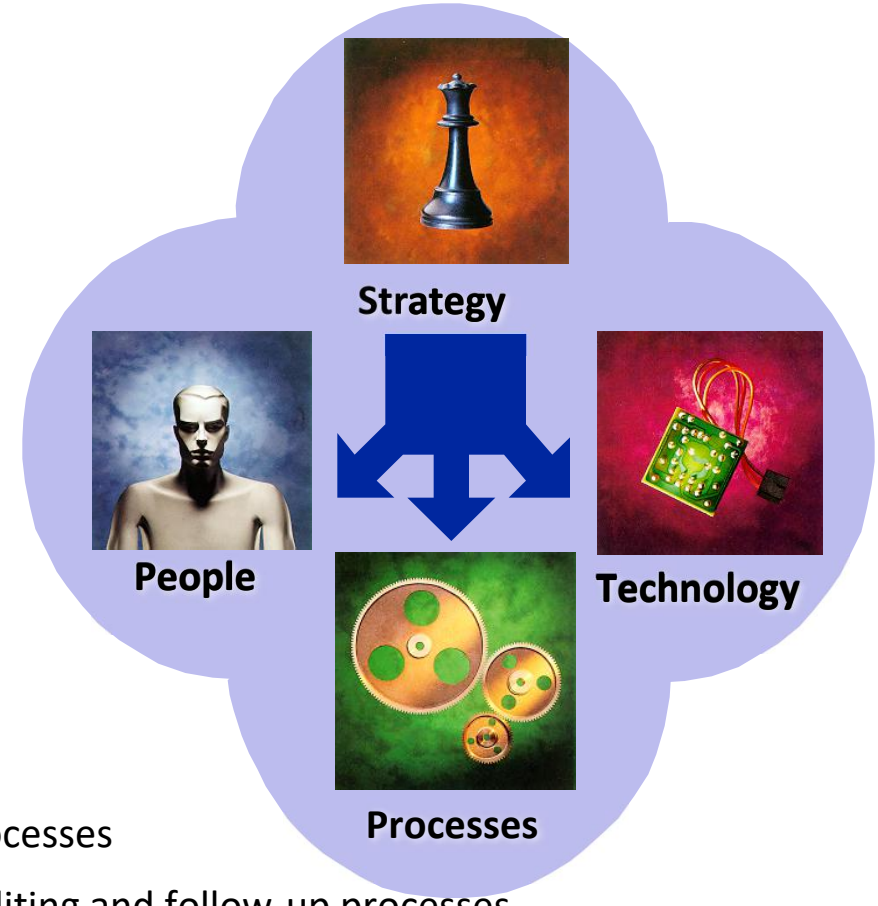
- Default configuration is seldom the most secure
- Security restriction vs flexible operation
- Exceptions to individuals

Communication over uncontrolled/unknown network links

- Public hotspots, campus networks, hostile governments

Dimensions to consider

- Training
- Awareness
- Organization of security



- Vulnerability scanning
- Firewalls
- Authentication
- Access Control
- Auditing
- Cryptography
- Digital Signatures
- Certification authorities
- Certification hierarchies
- etc...

- Security policies
- Security administration processes
- Continued evolution of auditing and follow-up processes

Security Policies

Define the power of each subject

- Least privilege principle; Hardening

Define security procedures

- Who does what in which circumstances

Define the minimum security requirements of a domain

- Security levels, Security Groups
- Required authorization
  - And the related minimum authentication requirements (Strong/weak, single/multifactor, remote/face-to-face)

Define defense strategies and fight back tactics

- Defensive architecture
- Monitoring of critical activities or attack signs
- Reaction against attacks or other abnormal scenarios

Define what are legal and illegal activities

- All that is not forbidden is allowed
- All that is not allowed is forbidden

Security mechanisms

Mechanisms implement policies

- Policies define, at a higher level, what needs to be done or exist
- Mechanisms are used to deploy policies

Generic security mechanisms

- Confinement (Sandboxing)
- Authentication
- Access control
- Privileged Execution
- Filtering
- Logging
- Inspection
- Auditing
- Cryptographic algorithms
- Cryptographic protocols

Security level of a computer

Depends on:

- Available security policies
- Correctness and effectiveness of their specification/implementation

Evaluation criteria

- NCSC Trusted Computer System Evaluation Criteria (TCSEC, Orange Book)
  - Classes: **D**, **C** (1, 2), **B** (1, 2, 3) e **A** (1)
  - D: insecure (minimum protection level)
  - A1: most secure
    - very demanding and expensive protection policies
    - formal validation of the specification with highly supervised implementation
- EC Information Technology Security Evaluation Criteria (ITSEC)
  - Levels: **E1** to **E6**
    - Level of formal specification
    - Correctness of the implementation

Case Study: NCSC TCSEC (C)

C1 – Discretionary Security Protection

- Identification and authentication
- Separation of users and data
- Discretionary Access Control (DAC) capable of enforcing access limitations on an individual basis
- Required System Documentation and user manuals

C2 – Controlled Access Protection

- More finely grained DAC
- Individual accountability through login procedures
- Audit trails
- Object reuse
- Resource isolation

Security policies for distributed systems

Must encompass several hosts and networks

Security Domains

- Definition of the set of hosts and networks of the domain
- Definition of the set of accepted/authorized users
- Definition of the set of accepted/not accepted activities

Security gateways

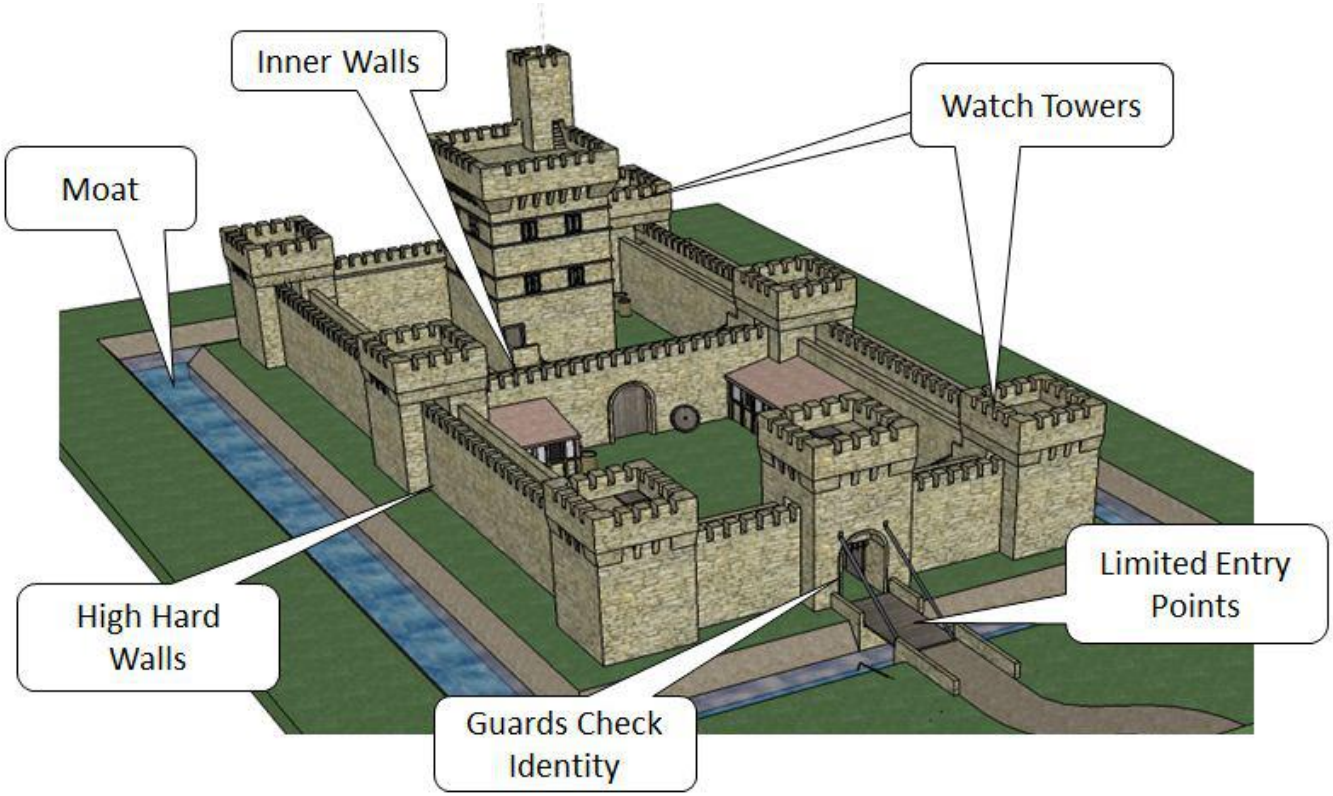
- Definition of the set of allowed in-out interactions

Security policies for distributed systems

Perimeter defense

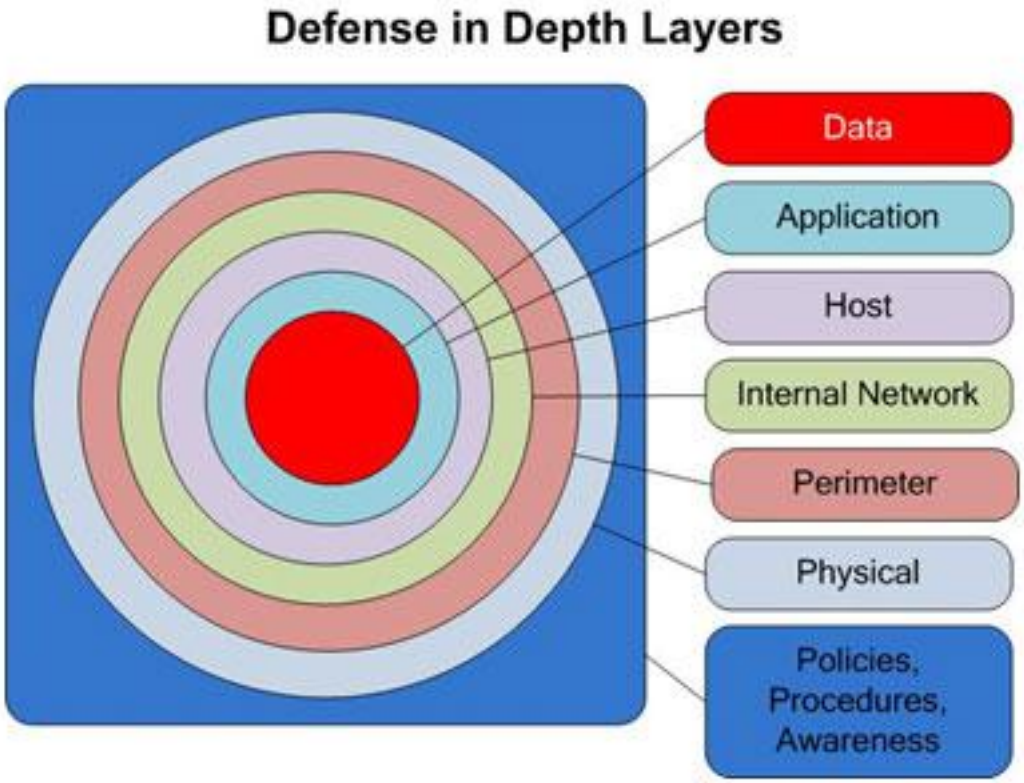
Security policies for distributed systems

Defense in depth



Security policies for distributed systems

Perimeter Defense vs Defense in Depth



Source: https://technet.microsoft.com/en-us/library/cc512681.aspx

Security policies for distributed systems

Target-specific attacks

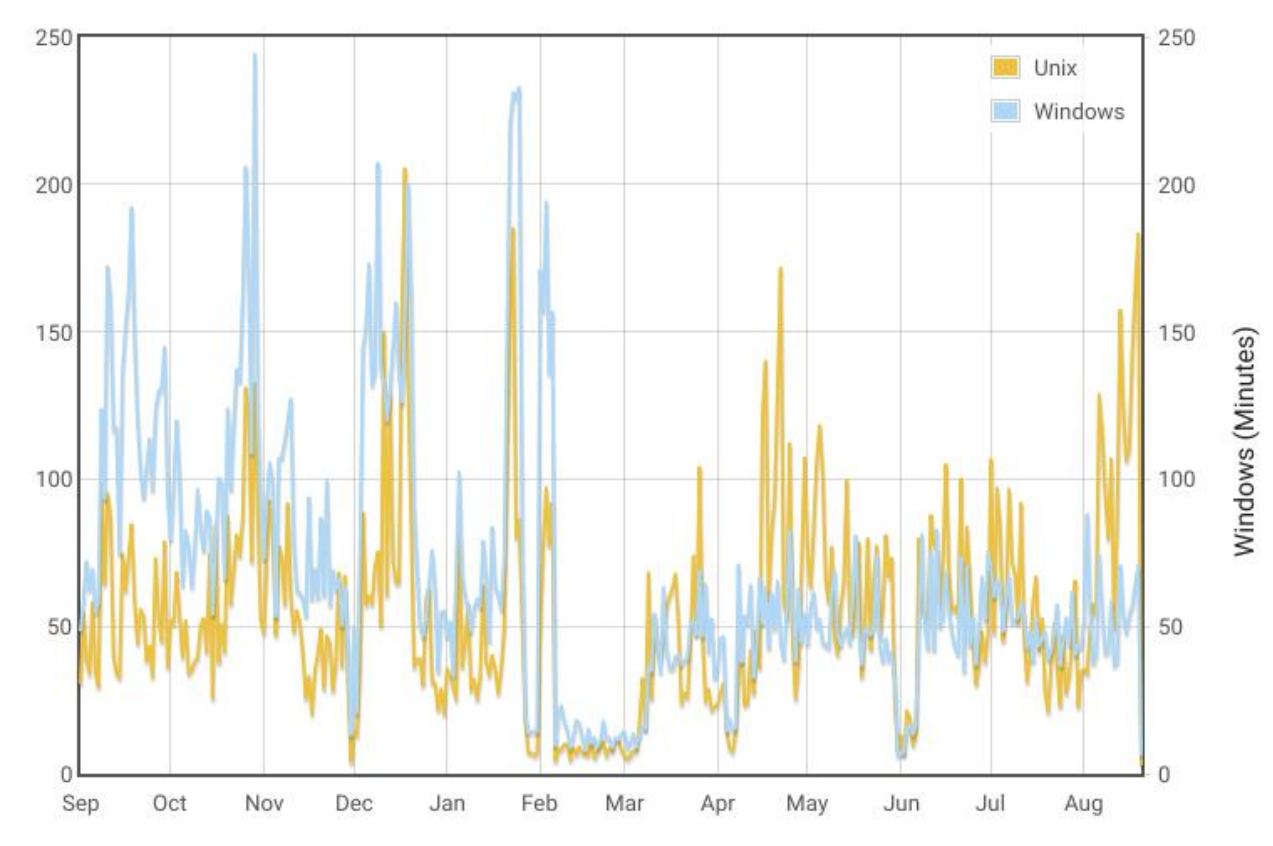
- Conceived for a particular host / network
- Idealized and conducted in real-time by specialists

Generic, autonomous attacks

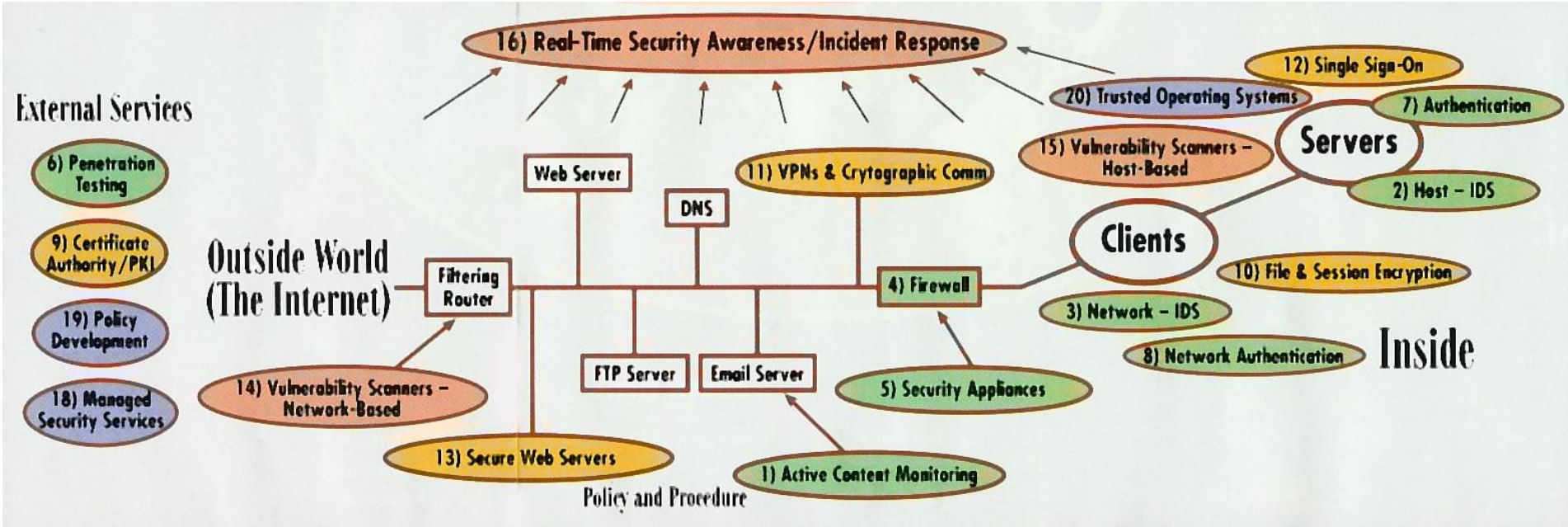
- Conceived for exploiting well-known, common vulnerabilities
- Coded for many scenarios and targets
- Mean survivability time :
  - Time between two consecutive automatic attacks
  - There are “network sensors” that help to compute it
- Executed by professionals, curious individuals, students

Case Study: Mean Survival Time

(http://isc.sans.org/survivaltime.html)



Mechanisms for distributed systems (1/5)



Mechanisms for distributed systems (2/5)

Trusted Operating Systems

- Security levels, certification
- Secure execution environments for servers
- Sand-boxing / virtual machines

Firewalls & Security Appliances

- Traffic control between networks
- Monitoring (traffic load, etc.)

Secure communications / VPNs

- Secure channels over insecure, public networks
- Secure extension of organizational networks

Mechanisms for distributed systems (3/5)

Authentication

- Local
- Remote (network authentication)
- Single Sign-On
- Using secrets, token, bio-metrics, device, location

Certification Authorities / PKI

- Management of public key certificates

Encryption of files and sessions

- Privacy / confidentiality of network data
- Privacy / confidentiality of long-term stored data

Mechanisms for distributed systems (4/5)

Intrusion detection

- Detention of forbidden / abnormal activities
- Network-Based / Host-based

Vulnerability scanners

- Scanning for problem fixing or exploitation
- Network-based / Host-based

Penetration testing

- Vulnerability assessment
- Demo penetration attempts
- Testing of installed security mechanisms
- Assessment of badly implemented security policies

Mechanisms for distributed systems (5/5)

Content monitoring

- Detection of virus, worms or other cyber plagues

Security administration

- Development of security policies
- Distributed enforcement of policies
- Co-administration / outsourcing of security services

Real-Time Security Awareness / Incident Response

- Capacity to detect and react to security incidents in real-time
- Means for a rapid and effective incident reaction