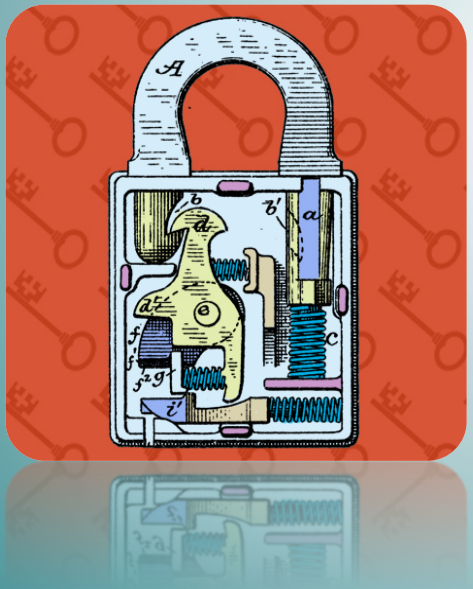
Garantía y Seguridad en Sistemas y Redes

Tema 1. Security Overview

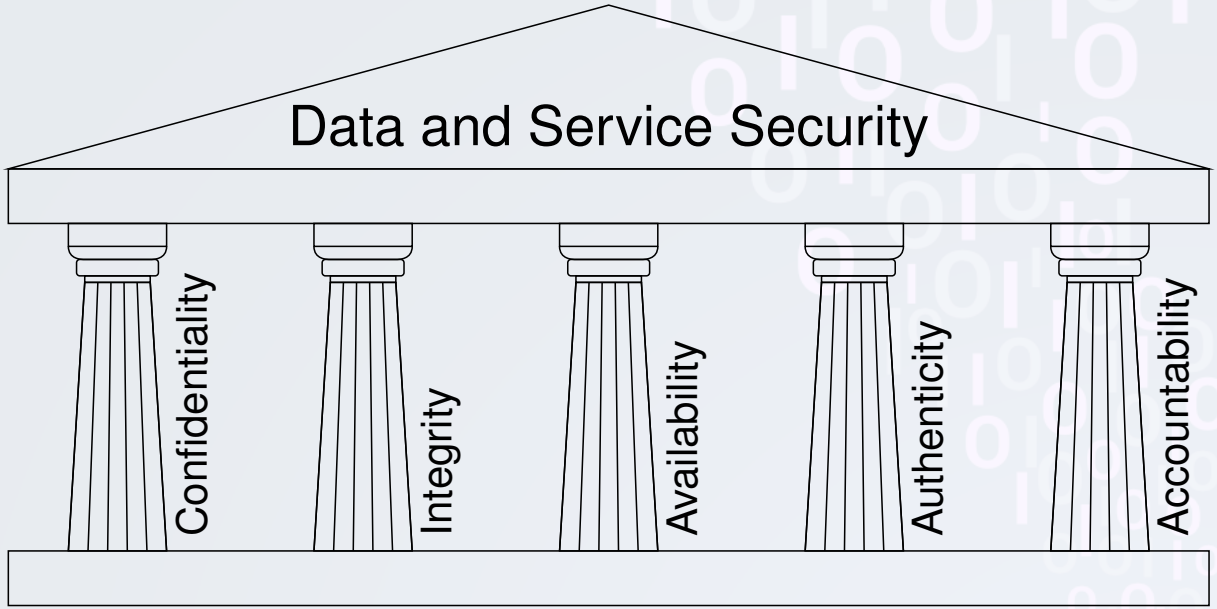


Esteban Stafford

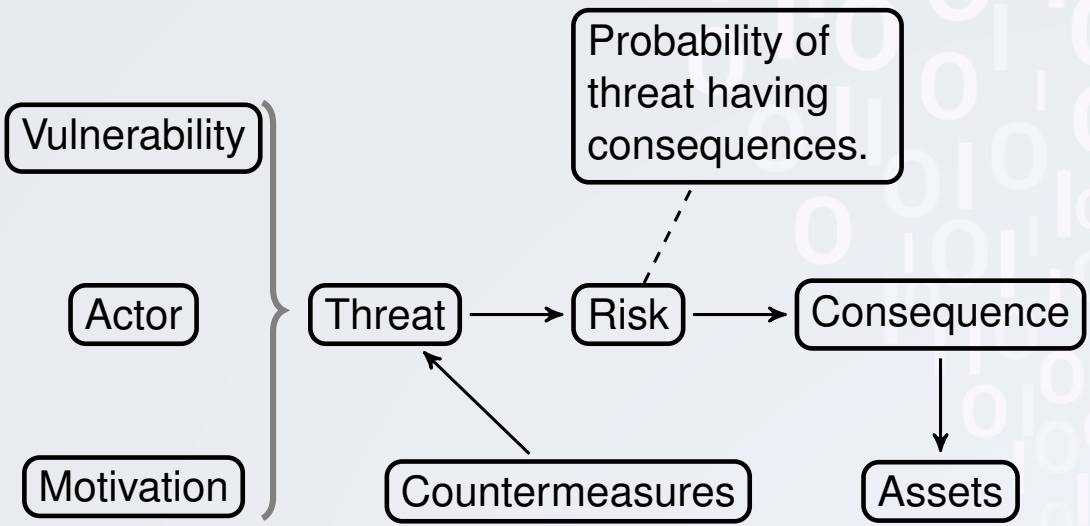
Departamento de Ingeniería Informática y Electrónica

Contents

Definition


Data and Service Security
Confidentiality
Integrity
Availability
Authenticity
Accountability

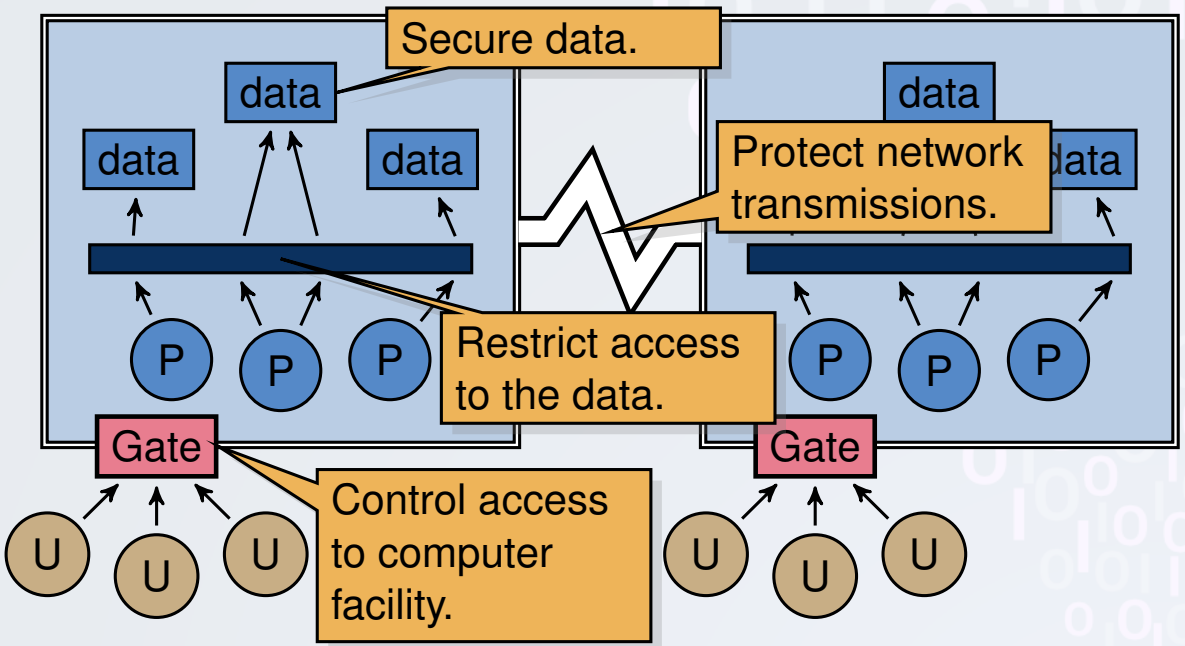
Terminology



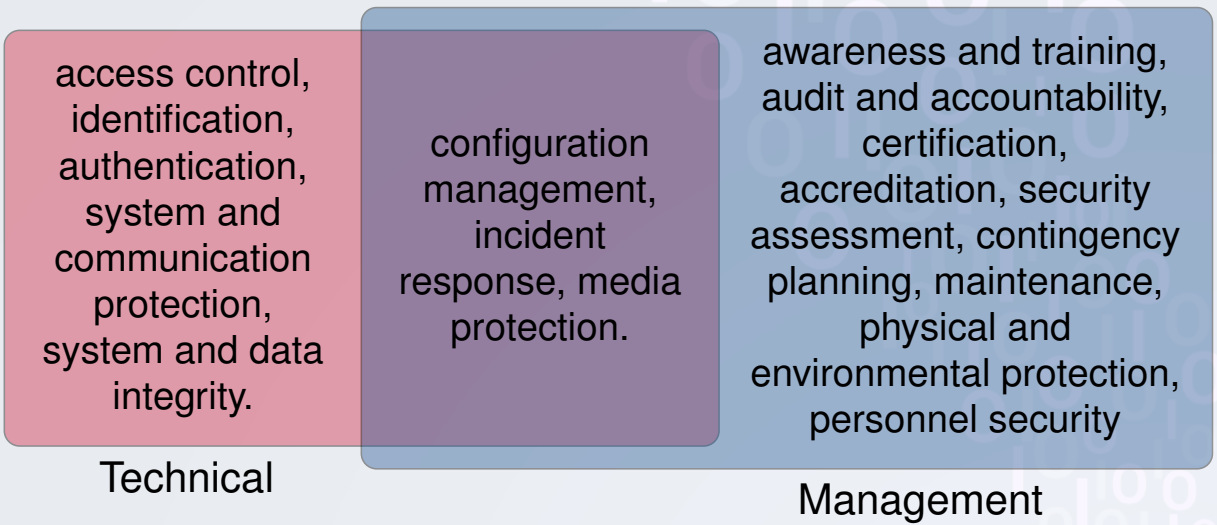
Threats and Attacks

Consequence	Attack
(Unauthorized) Disclosure	Exposure Interception Inference Intrusion
Deception	Masquerade Falsification Repudiation
Disruption	Incapacitation Corruption Obstruction
Usurpation	Misappropriation Misuse

Scope of Computer Security

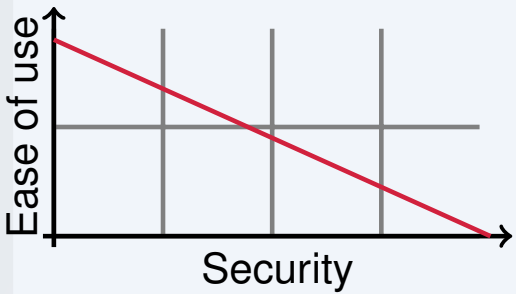


Technical and Management Measures



Security tradeoffs

Ease of use vs Security



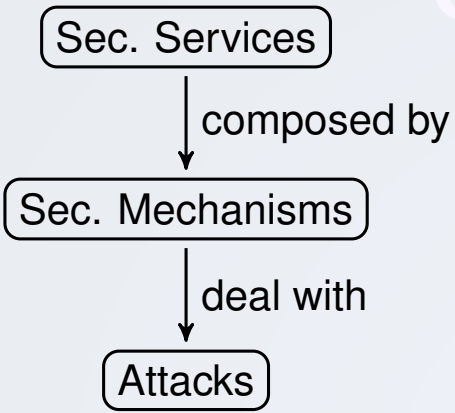
- Overwhelming security measures can turn users against you.

How much security is needed?

- Need to detect and evaluate the cost of attacks.
- Where is 100% risk protection?

Security Architecture for OSI

- ITU-T[4] Recommendation X.800.
- Helps Security manager to organize the task of providing security.
- Developed for communication systems but applicable to computer systems.



Security Services

- **Authentication**: Authenticate communicating peer and data sources.
- **Access control**: Protect against unauthorized use of resources.
- **Data confidentiality**: Protect data from unauthorized disclosure.
- **Data integrity**: Counter active threats to integrity of data.
- **Non-repudiation**: Protect against false denials of handling data.

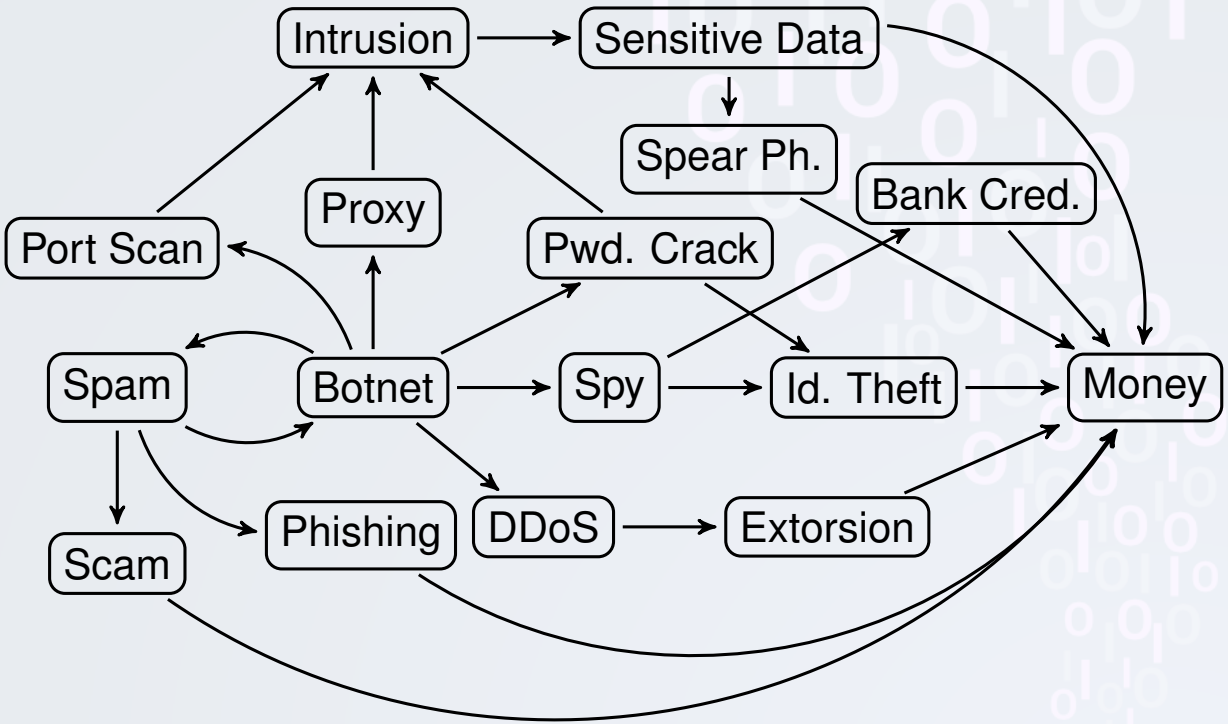
Security Mechanisms

- Specific security mechanisms:
  - **Encipherment**: Make data unreadable without key.
  - **Data integrity**: Detect/avoid data corruption.
  - **Digital signature**: Relate data to specific sender.
  - **Authentication**: Ensure identity of user.
  - **Access control**: Manage user permissions.
  - **Traffic padding**: Add data to hinder traffic analysis.
  - **Routing control**: Reroute data through secure channels.
  - **Notarization**: Use trusted 3rd party to assure security.
- Pervasive security mechanisms:
  - **Trusted functionality**: Use trusted systems/protocols.
  - **Security labels**: Mark data by its security attributes.
  - **Event detection**: Detection of security related events.
  - **Security audit trail**: Recording security data for auditing.
  - **Security recovery**: Recovery in response to events and functions.

Who is against us?

> "Know your enemy and know yourself and you can fight a thousand battles without disaster."
>
> — Sun Tzu. *The Art of War*

Cybercrime Today



Money? How much?

Item pricing in black market

Rank			Percentage		
2008	2007	Item	2008	2007	Price Range
1	1	Credit card information	32%	21%	$0.06 - $30
2	2	Bank account credentials	19%	17%	$10 - $1000
3	9	Email account	5%	4%	$0.10 - $100
4	3	Email addresses	5%	6%	$0.33/MB - $100/MB
5	12	Proxies	4%	3%	$0.16 - $20
6	4	Full identities	4%	6%	$0.70 - $60
7	6	Mailers	3%	5%	$2 - $40
8	5	Cash out services	3%	5%	$200 - $2000
9	17	Shell scripts	3%	2%	$2 - $20
10	8	Scam hosting	3%	5%	$3 - $40/week

Source: Symantec, Internet Security Threat Report Volume XIV: April, 2009

Money? How much?

Attack tool pricing in black market

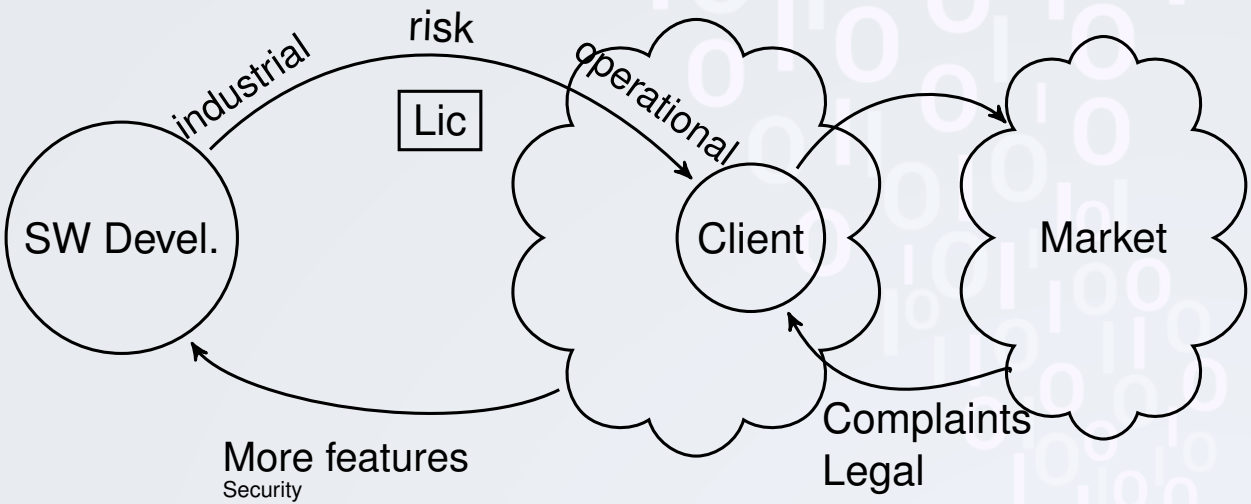
Tool type	Avg. Price	Price Range
Botnet	$225	$150 - $300
Autorooter	$70	$40 - $100
SQL injection tools	$63	$15 - $150
Shopadmin exploiter	$33	$20 - $45

Exploit pricing in black market

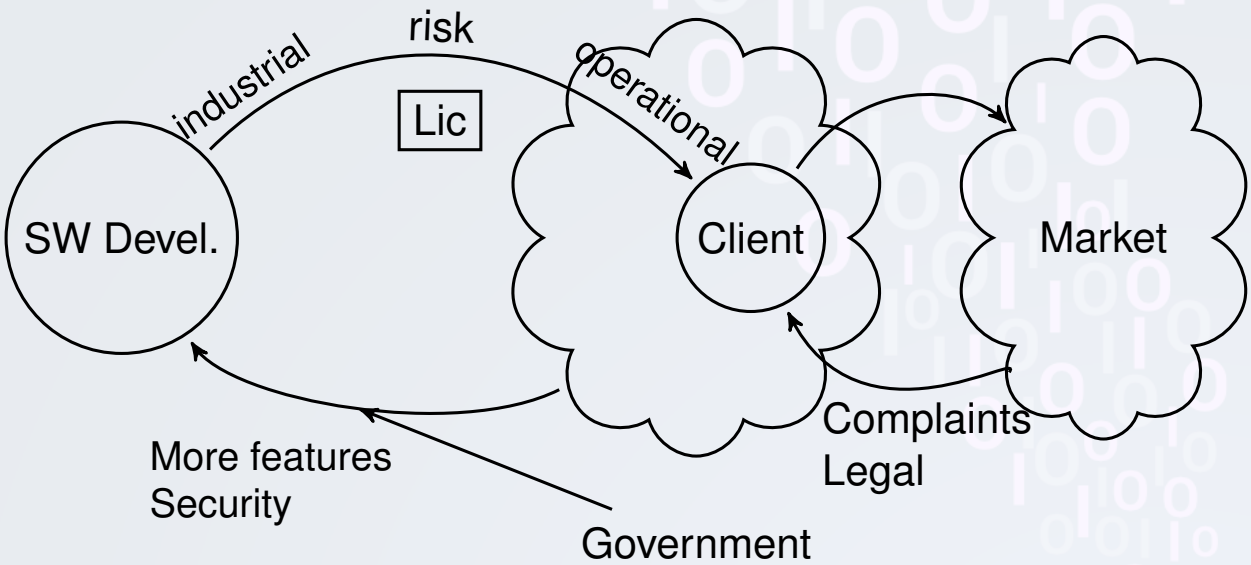
Exploit type	Avg. Price	Price Range
Site-specific vulnerability (financial site)	$740	$100 - $3000
Remote file include exploit (500 links)	$200	$150 - $250
Shopadmin (50 exploitable shops)	$150	$100 - $200
Browser exploit	$37	$5 - $60
Remote file include exploit (100 links)	$34	$20 - $50
Remote file include exploit (200 links)	$70	$50 - $80
Remote operating system exploit	$9	$8 - $10

Source: Symantec

Risk transfer and market feedback



Risk transfer and market feedback



Complexity of computer systems

- Current situation overwhelms programmers:
  - C Language: Over 700 pages of spec.
  - C Compiler: 3.7 million lines of code.
  - Runtime library: 1.7 million lines of code.
  - Operating system: Thousands of pages of spec and millions of lines of code
  - Processor and other hardware
- In the future:
  - Hide complexity in layers. HW, OS, Compiler, Libraries, Application.
  - Well described interfaces do not have undefined outcome.
  - Each layer developer is responsible for the source code level mistakes.