An Introduction to Cyber Security – CS 573

Instructor: Dr. Edward G. Amoroso
eamoroso@tag-cyber.com

Week 9

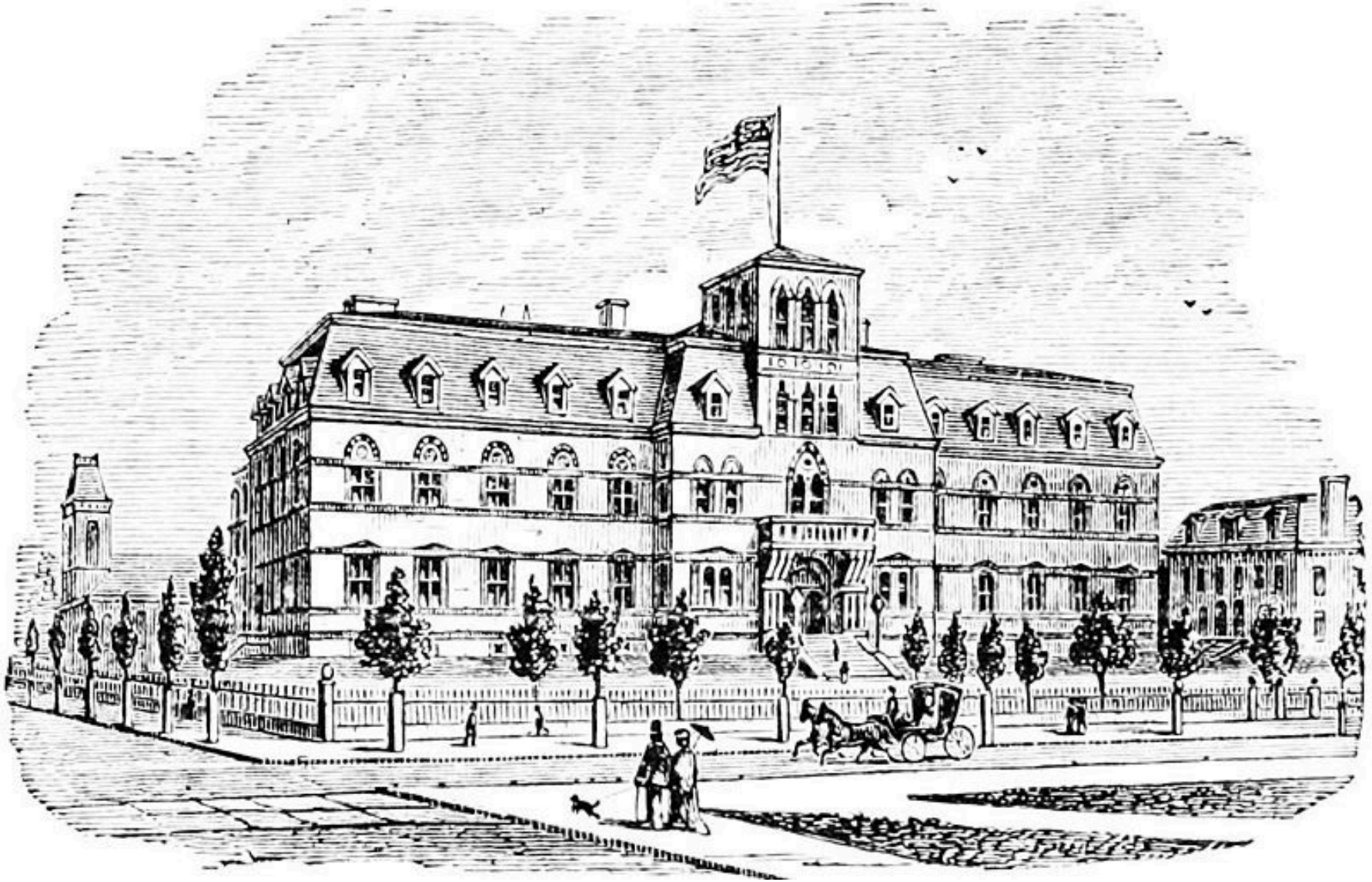
Required Week Nine Readings

1. "Looking Back at the Bell-LaPadula Model" David Bell, 2005 https://www.acsac.org/2005/papers/Bell.pdf

2. Finish the book: *From CIA to APT: An Introduction to Cyber Security*, E. Amoroso & M. Amoroso

Twitter: @hashtag_cyber
LinkedIn: Edward Amoroso

Week 9



Week 9: Security Policy Models

What are the Subject-Object Security Models?
(Hint: Bell-LaPadula, Biba)

Week 9

The Father of Computer Security Policy

Leonard J. LaPadula

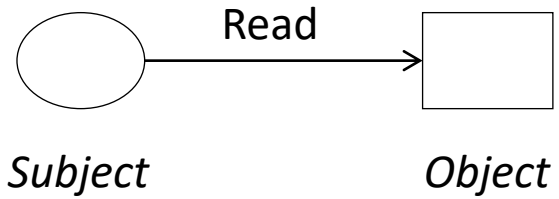
Co-author of the Bell-LaPadula model of computer security.

NATIONAL CYBER SECURITY HALL OF FAME

With a major in Mathematics, Len got his start in computers with IBM in the Time Life building in NYC in 1960. He continued in this new field as an Army Lieutenant with the Army Security Agency Training Center and School at Fort Devens, Massachusetts from 1961 to 1963. There, he worked with an IBM 650 (1000 words of main memory!) and taught Fortran programming. After over 40 years with The MITRE Corporation, where he worked on numerous federal government contracts, Len retired in 2013. His principal efforts with MITRE supported various initiatives to improve computer and network security, ending with a project on cyber resiliency. In 1973, he co-authored with David Bell a pioneering paper on computer security, which came to be known as the "Bell-LaPadula Model". This mathematical model became part of the computer science curriculum in many universities, influenced developments in computer systems, and contributed to the "Orange Book" series published by the National Security Agency. As a retiree, he enjoys quiet gardening.

Week 9

Subject-Object Model (Read)



Week 9

Subject-Object Model (Read, Write)

Week 9

Confidentiality Label Hierarchy

Top Secret

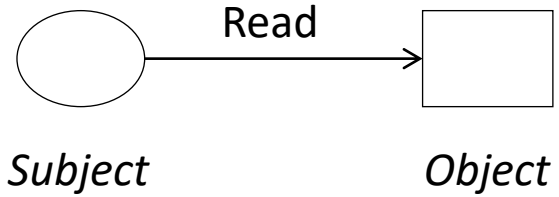
Secret

Unclassified

Week 9

Secret Subject Reads *Secret* Object

Top Secret



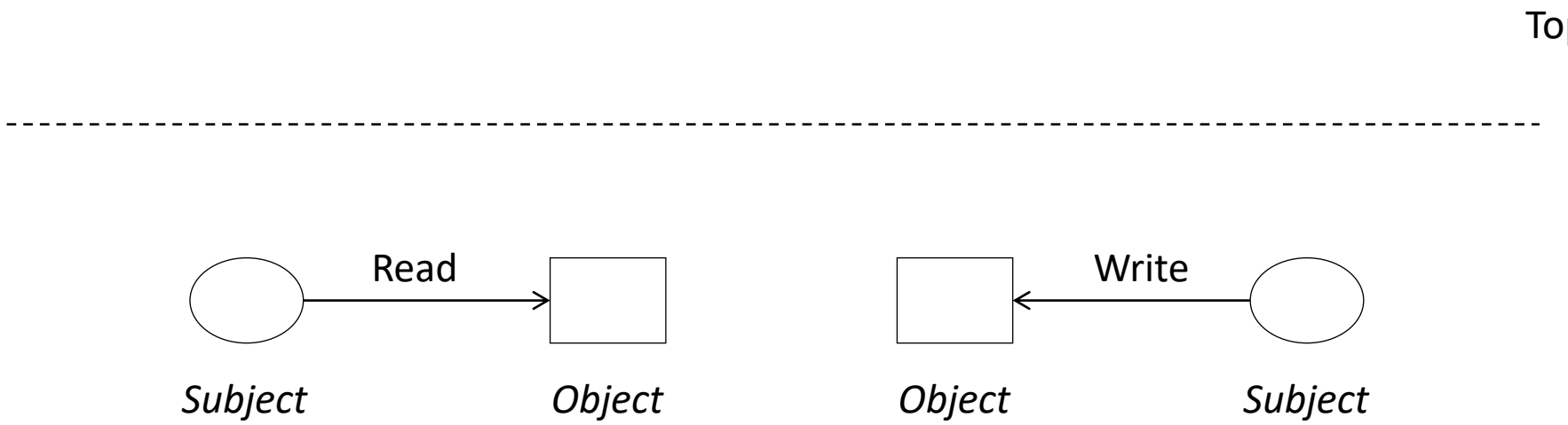
Secret

Unclassified

Week 9

Secret Subject Writes to *Secret* Object

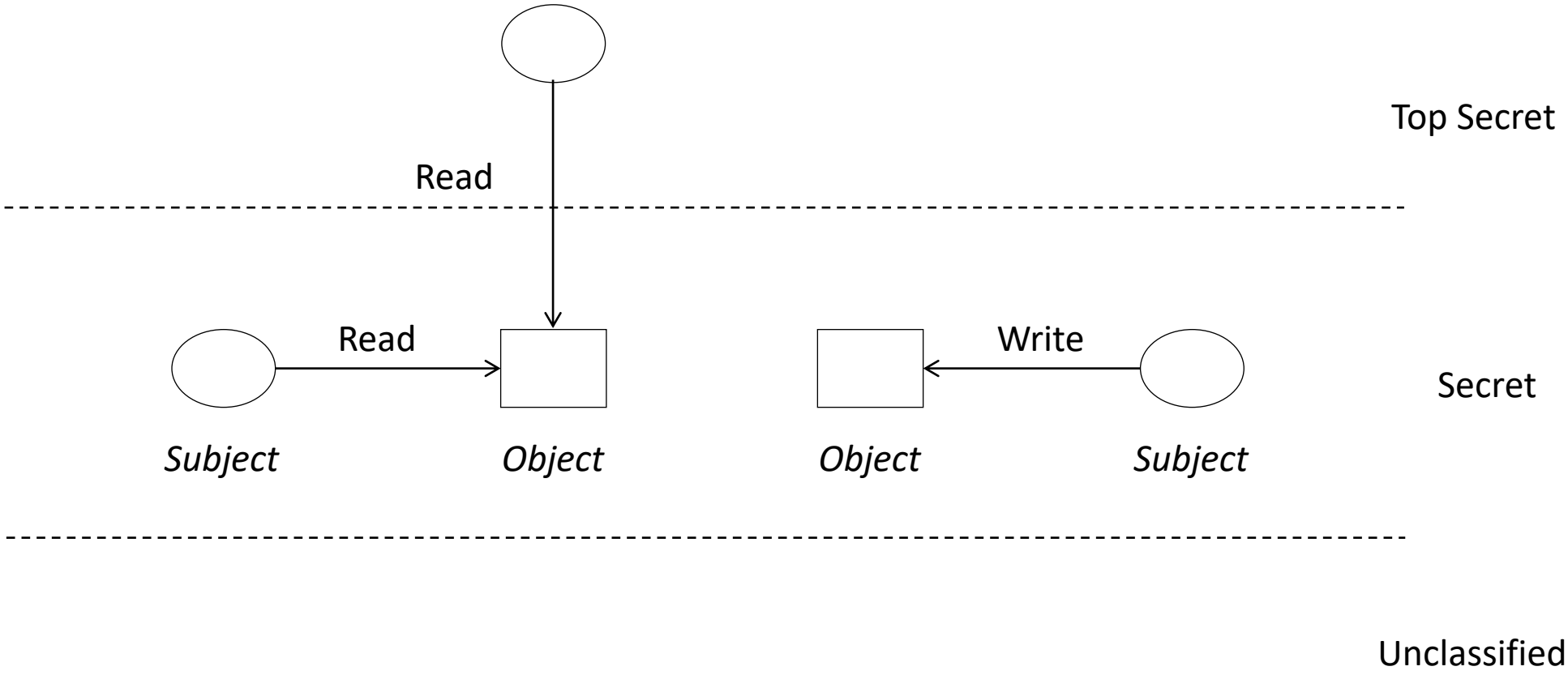
Top Secret



Unclassified

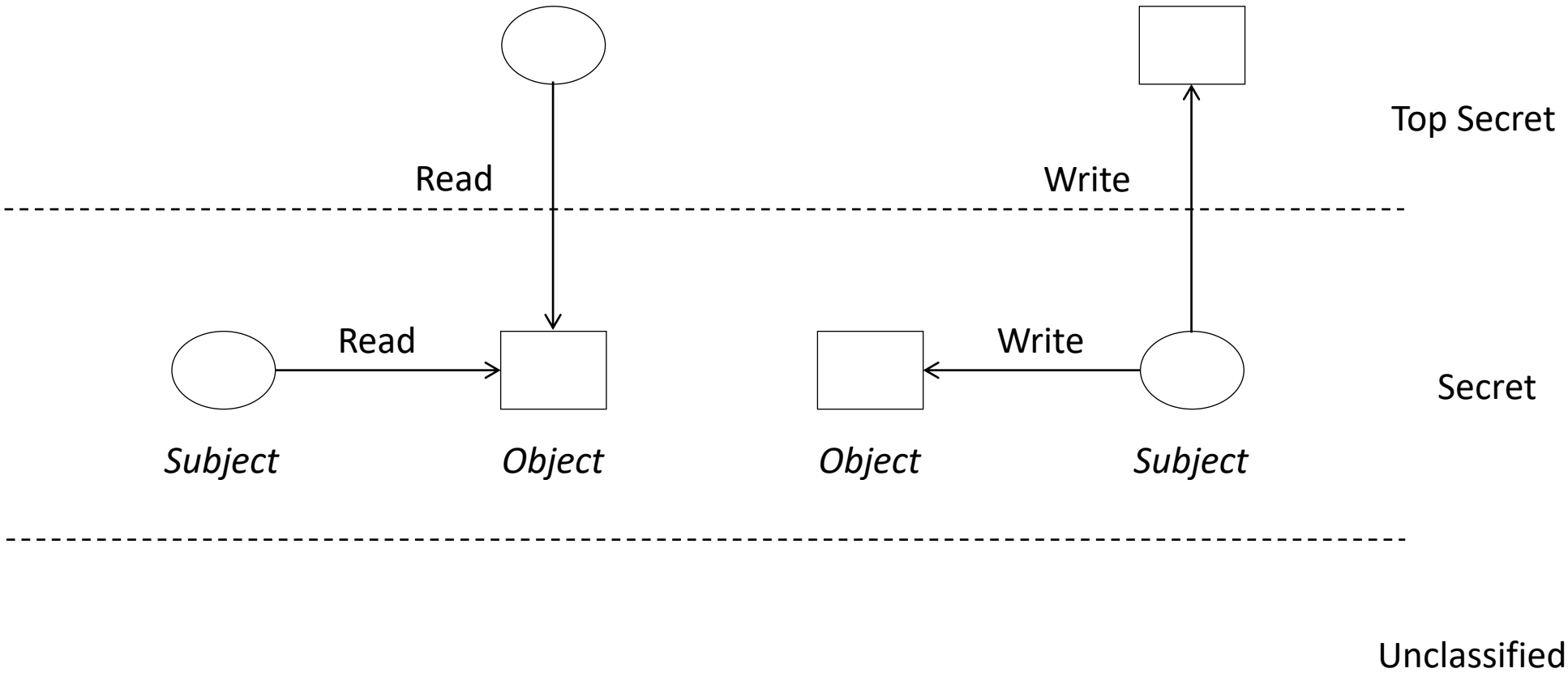
Week 9

Top Subject Reads *Secret* Object



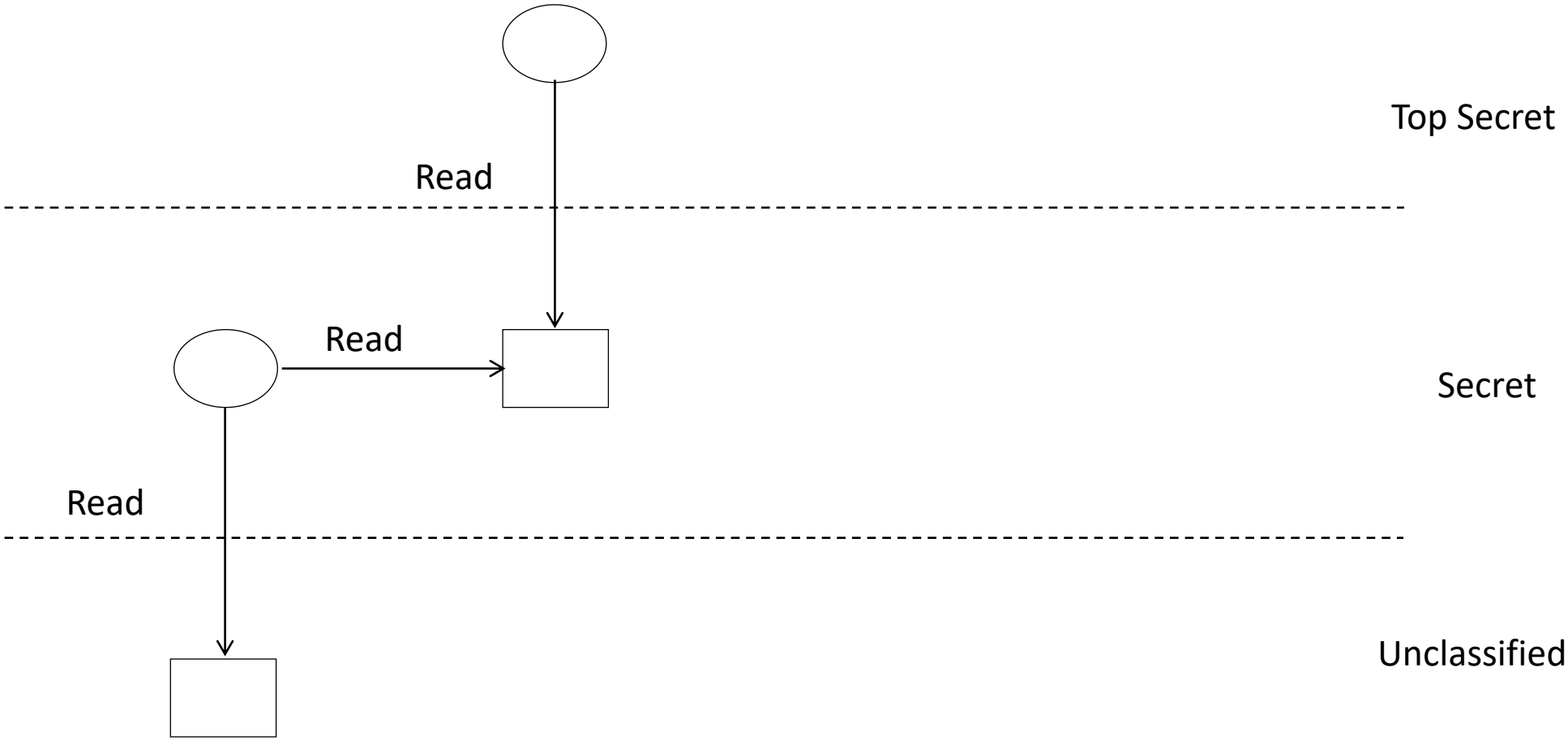
Week 9

Secret Subject Writes to *Top Secret* Object



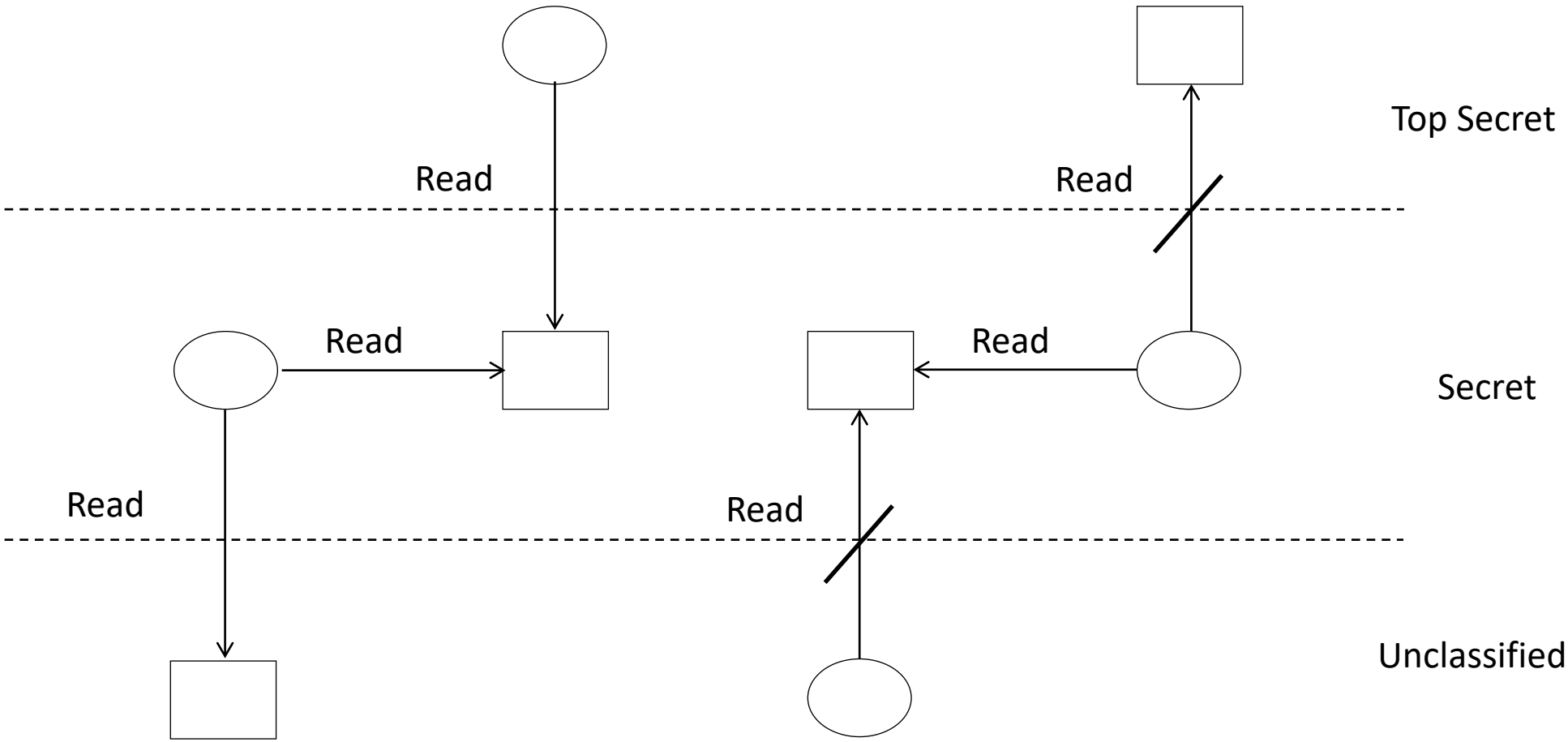
Week 9

Read “Down” Allowed



Week 9

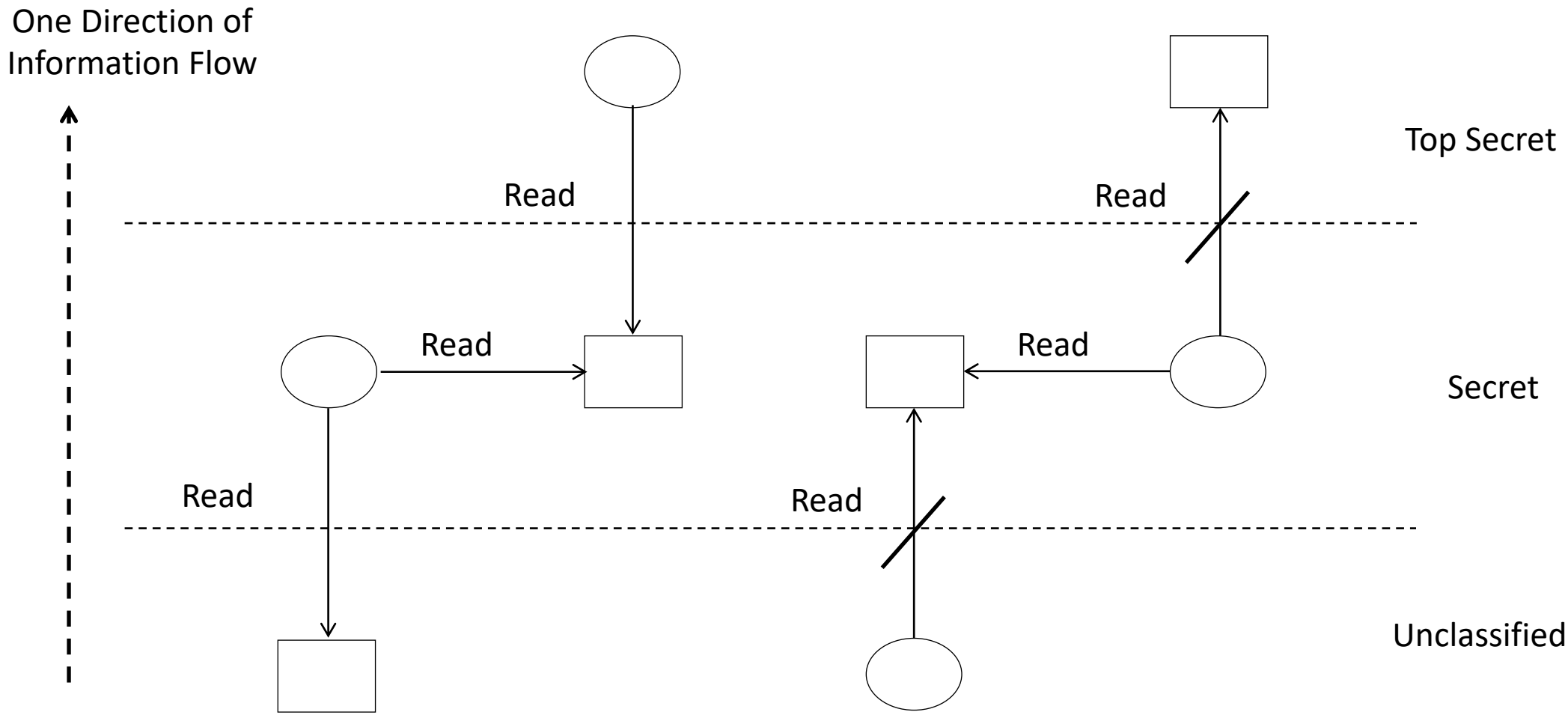
Read “Up” Disallowed



Week 9

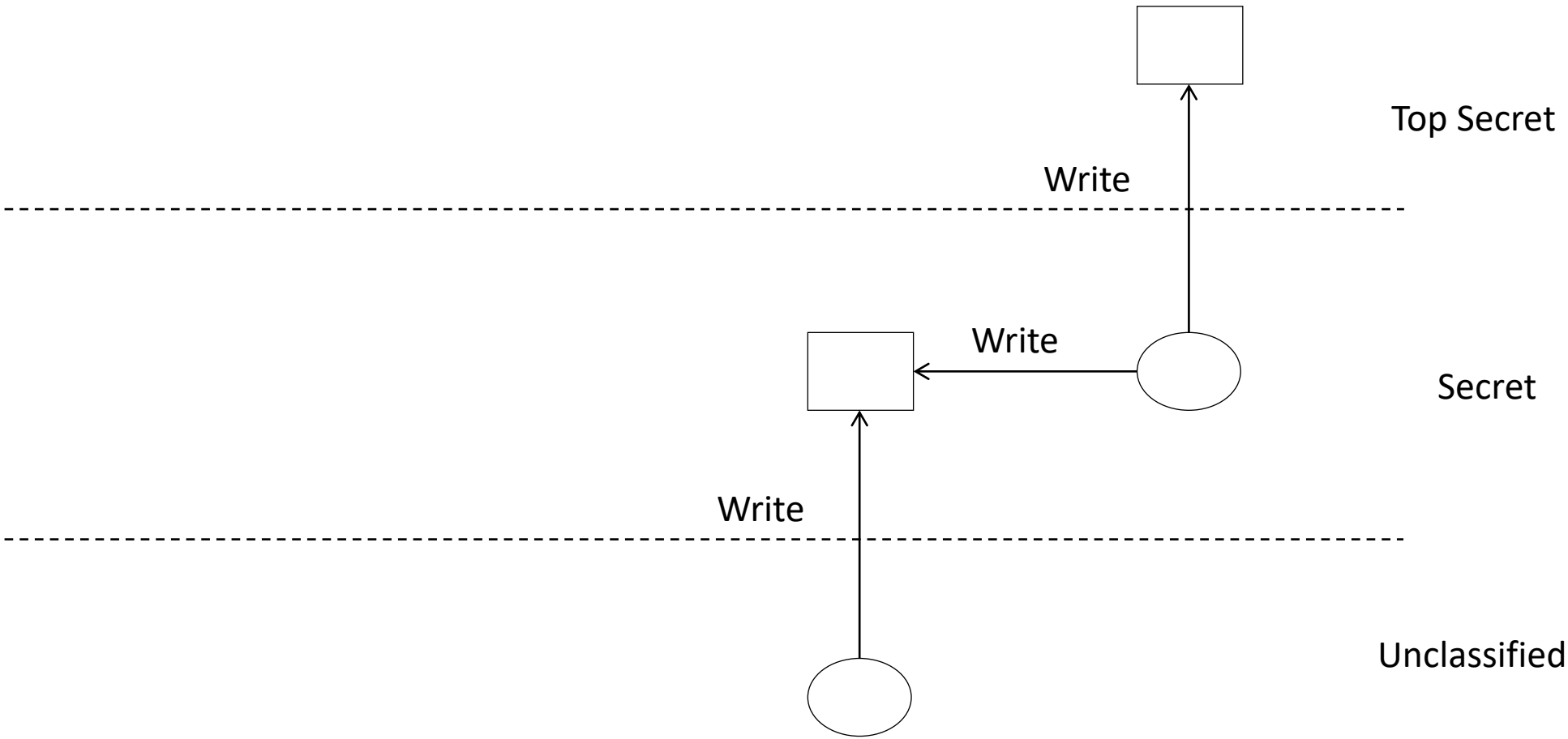
Bell-LaPadula Model – Simple Security Property – Confidentiality

"No Read Up"



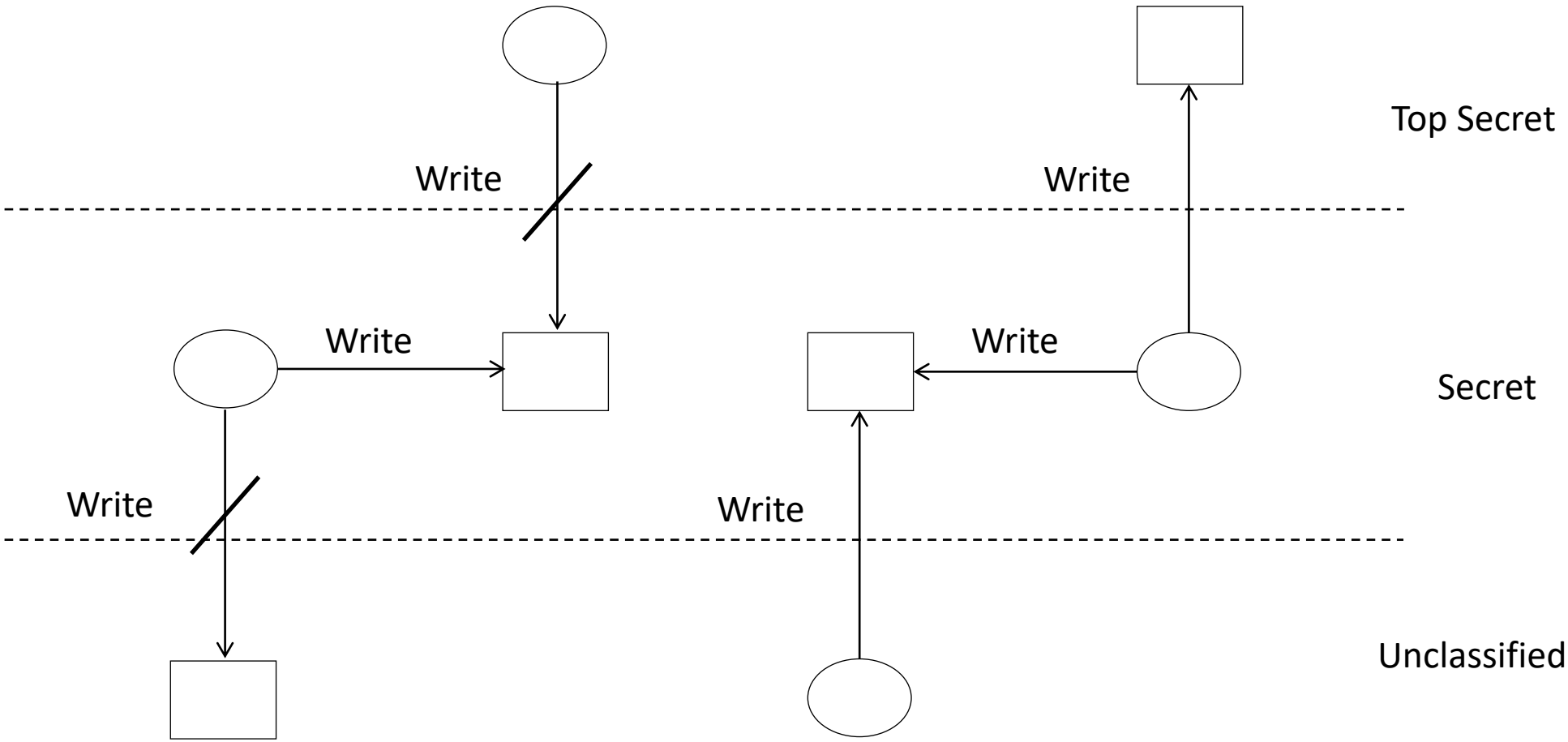
Week 9

Write “Up” Allowed



Week 9

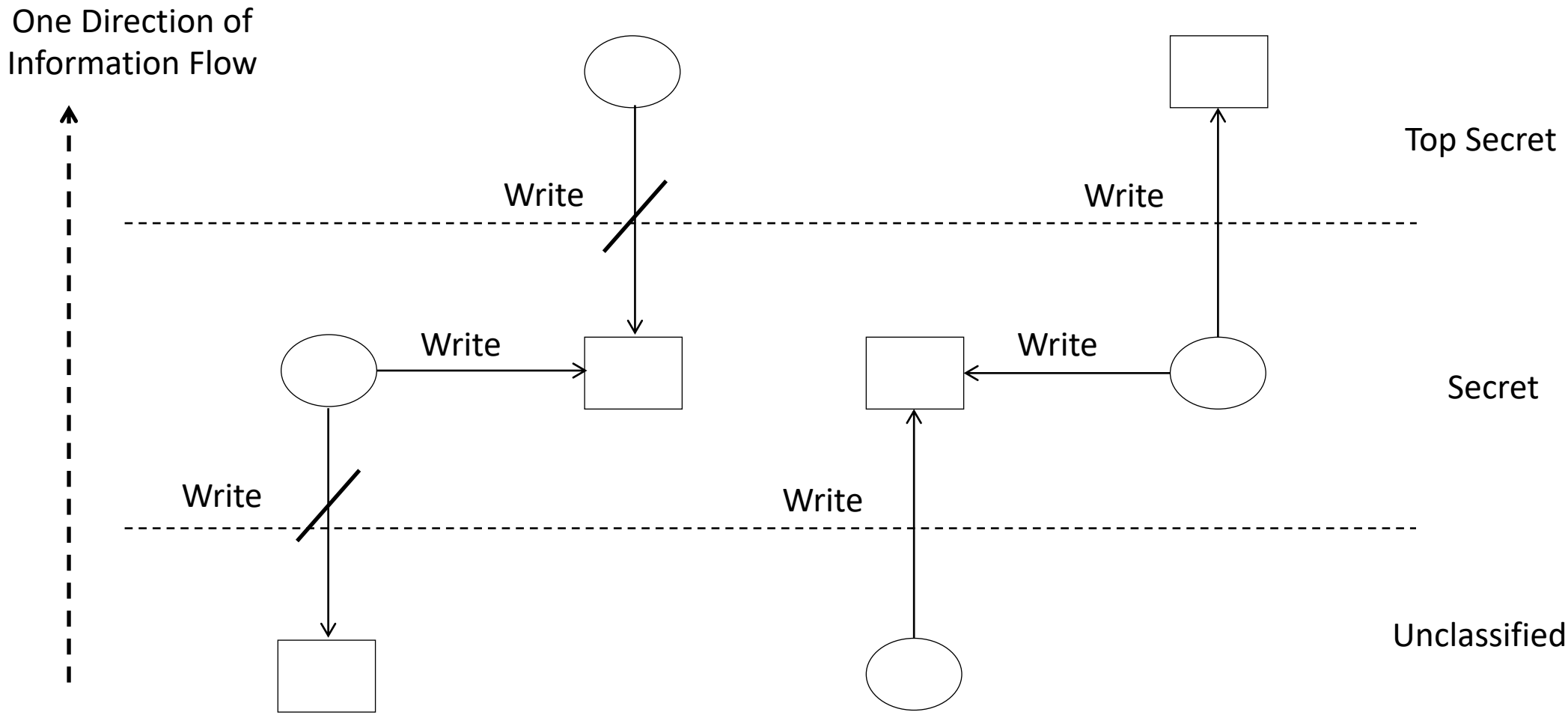
Write “Down” Disallowed



Week 9

Bell-LaPadula Model – Star Property – Confidentiality

"No Write Down"



Week 9

Integrity Label Hierarchy

High Integrity

Medium Integrity

Low Integrity

Week 9

Subject Reads and Writes – Integrity

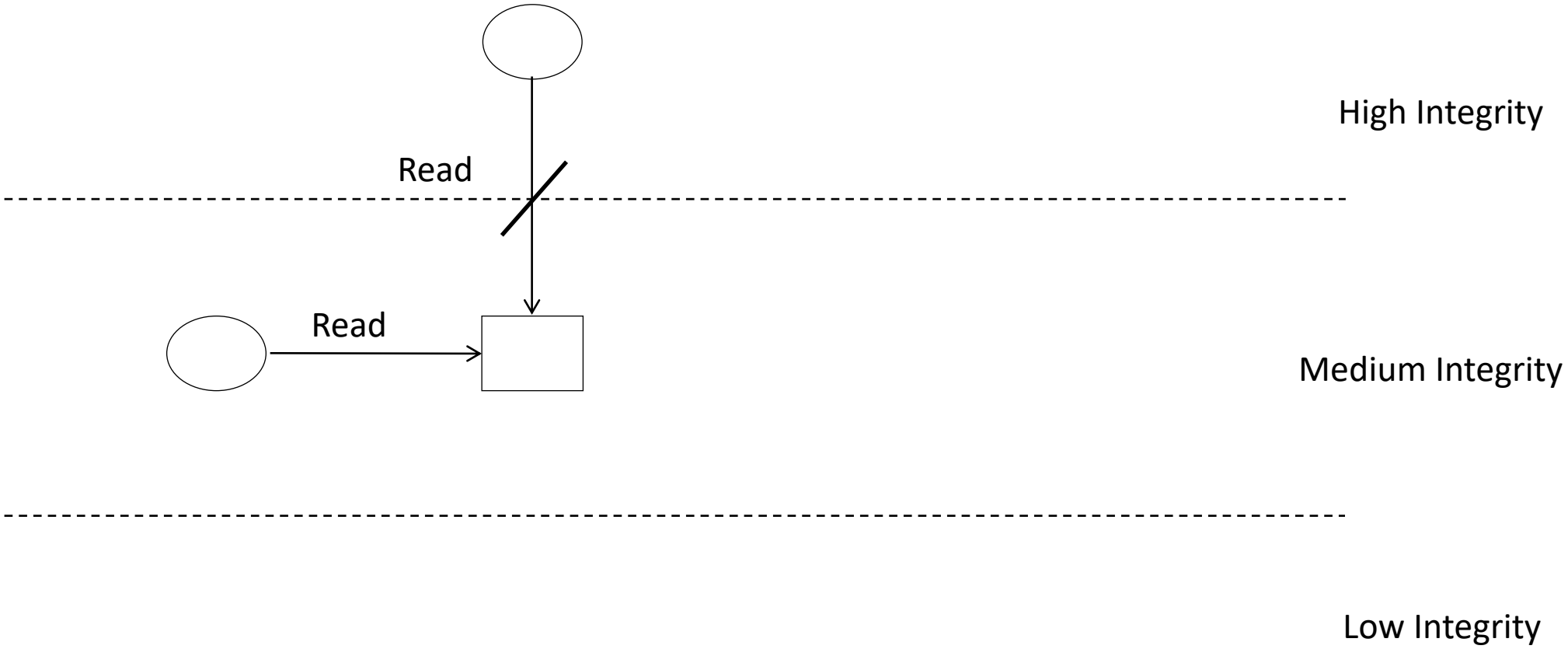
High Integrity

Medium Integrity

Low Integrity

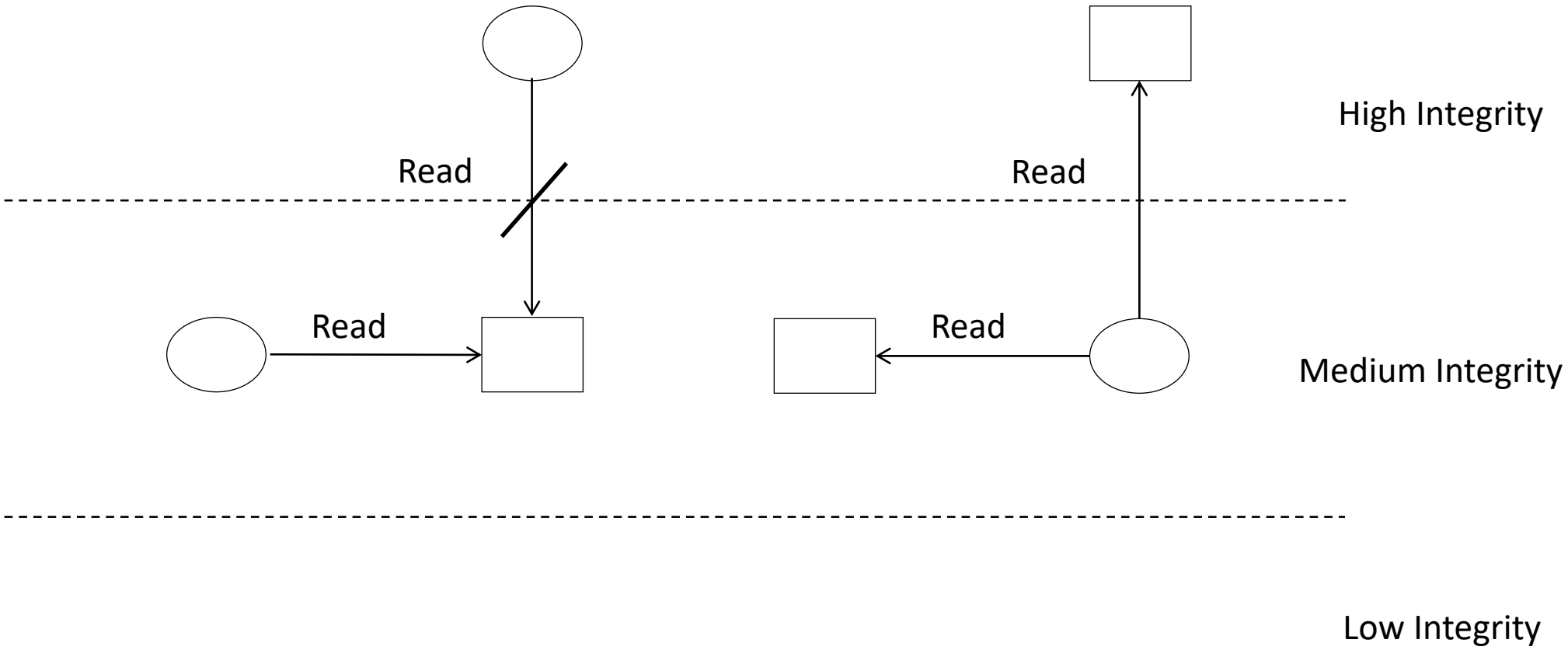
Week 9

Read Down Disallowed



Week 9

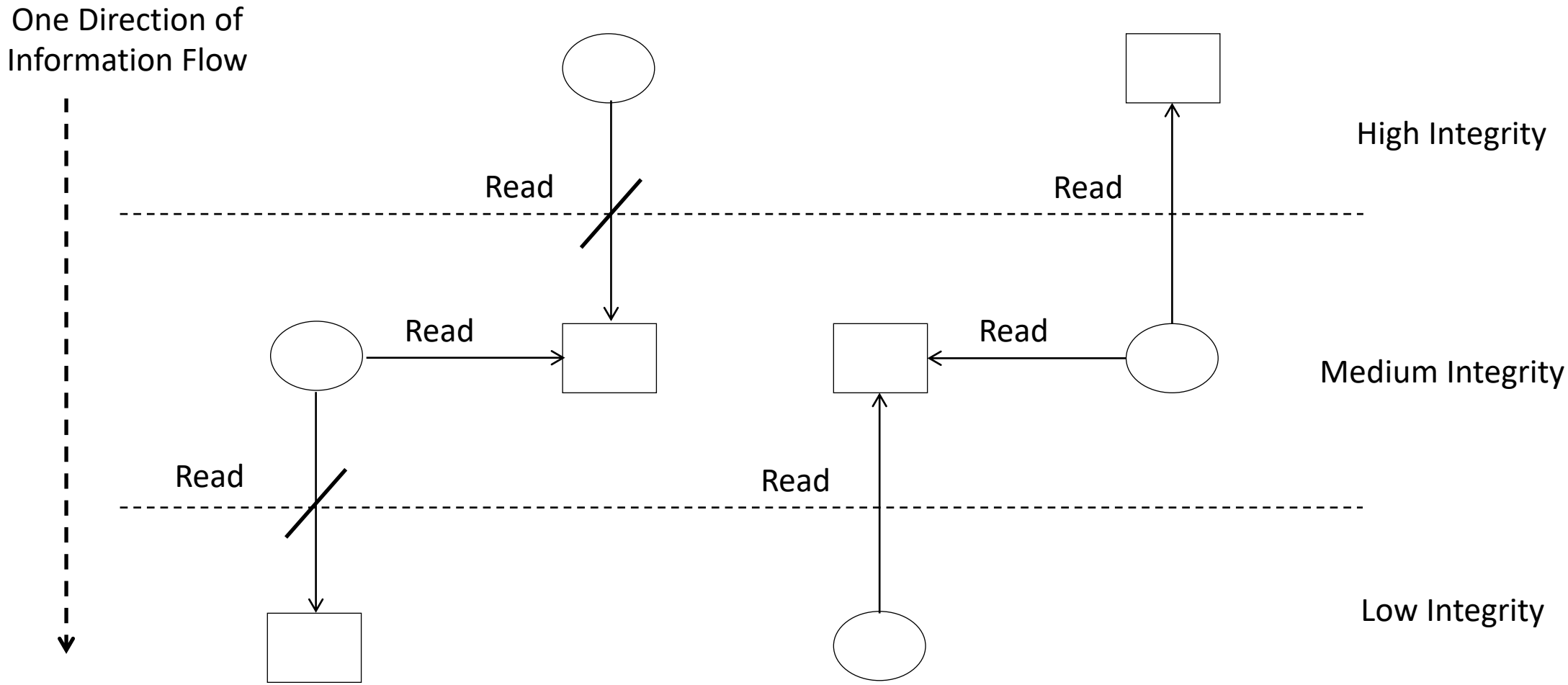
Read Up Allowed



Week 9

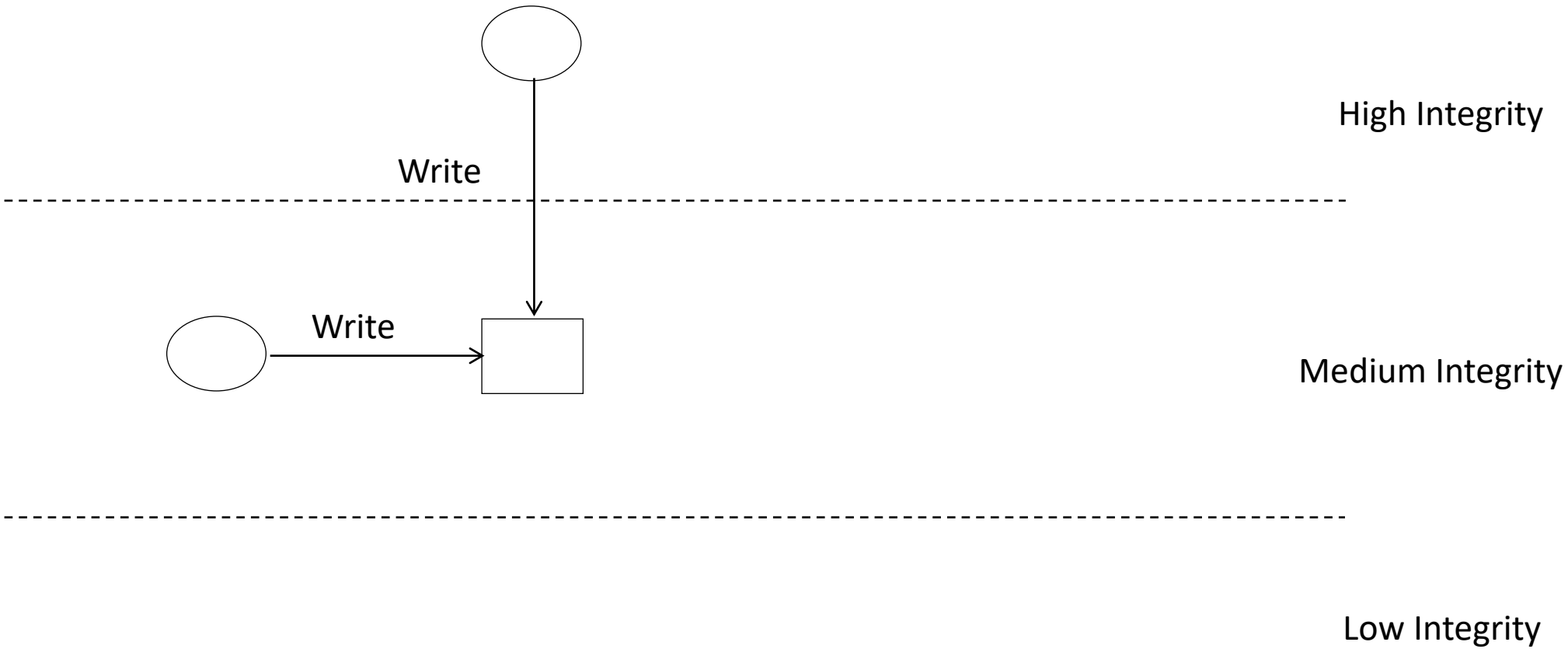
Biba Model – Mandatory Read Integrity Property

"No Read Down"



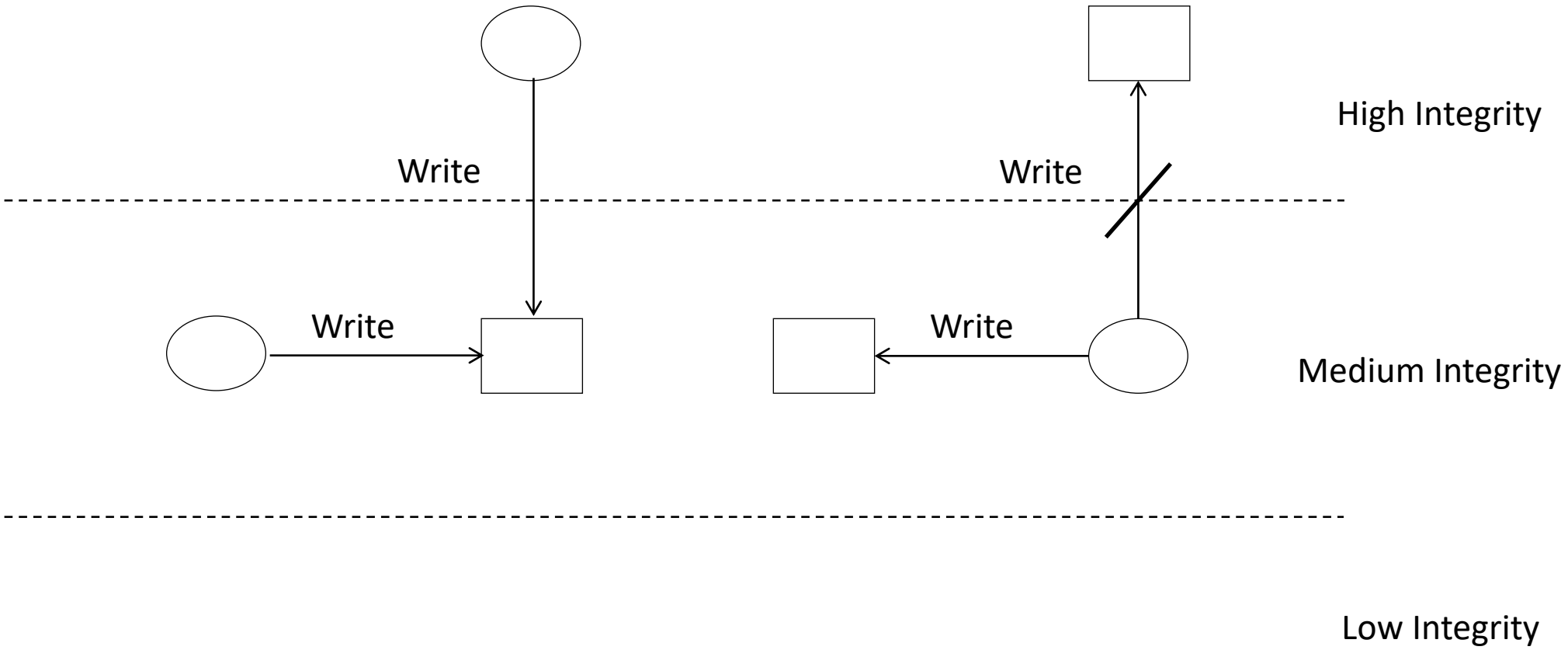
Week 9

Write Integrity Down Allowed



Week 9

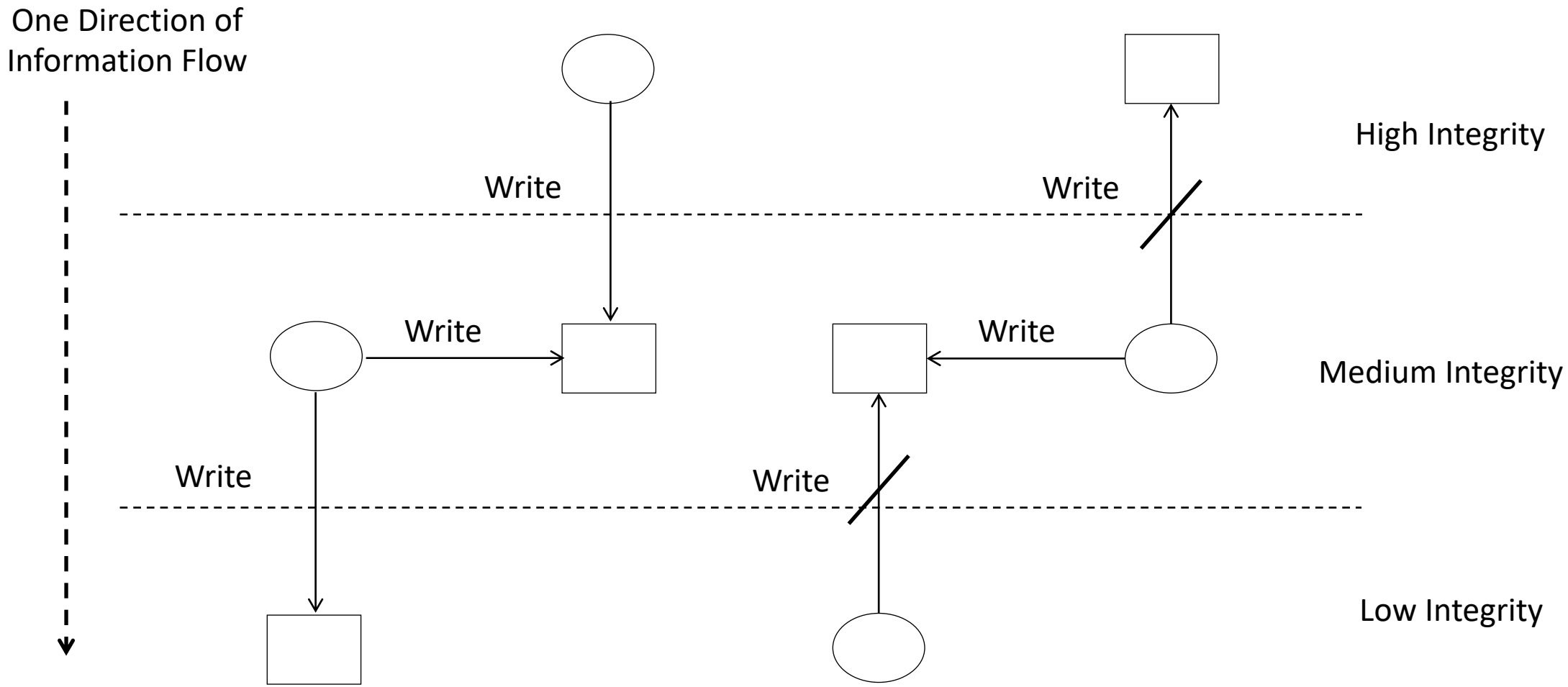
Write Up Disallowed



Week 9

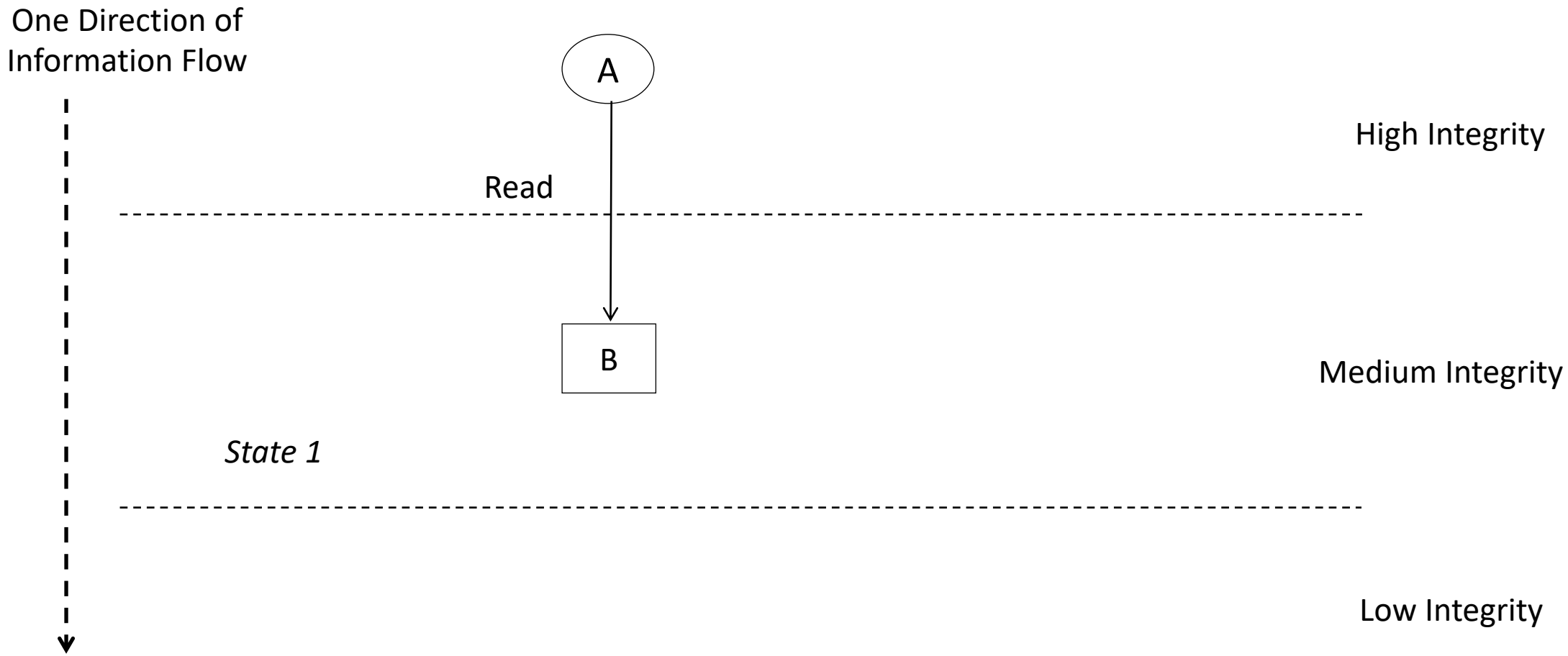
Biba Model – Mandatory Write Integrity Property

"No Write Up"



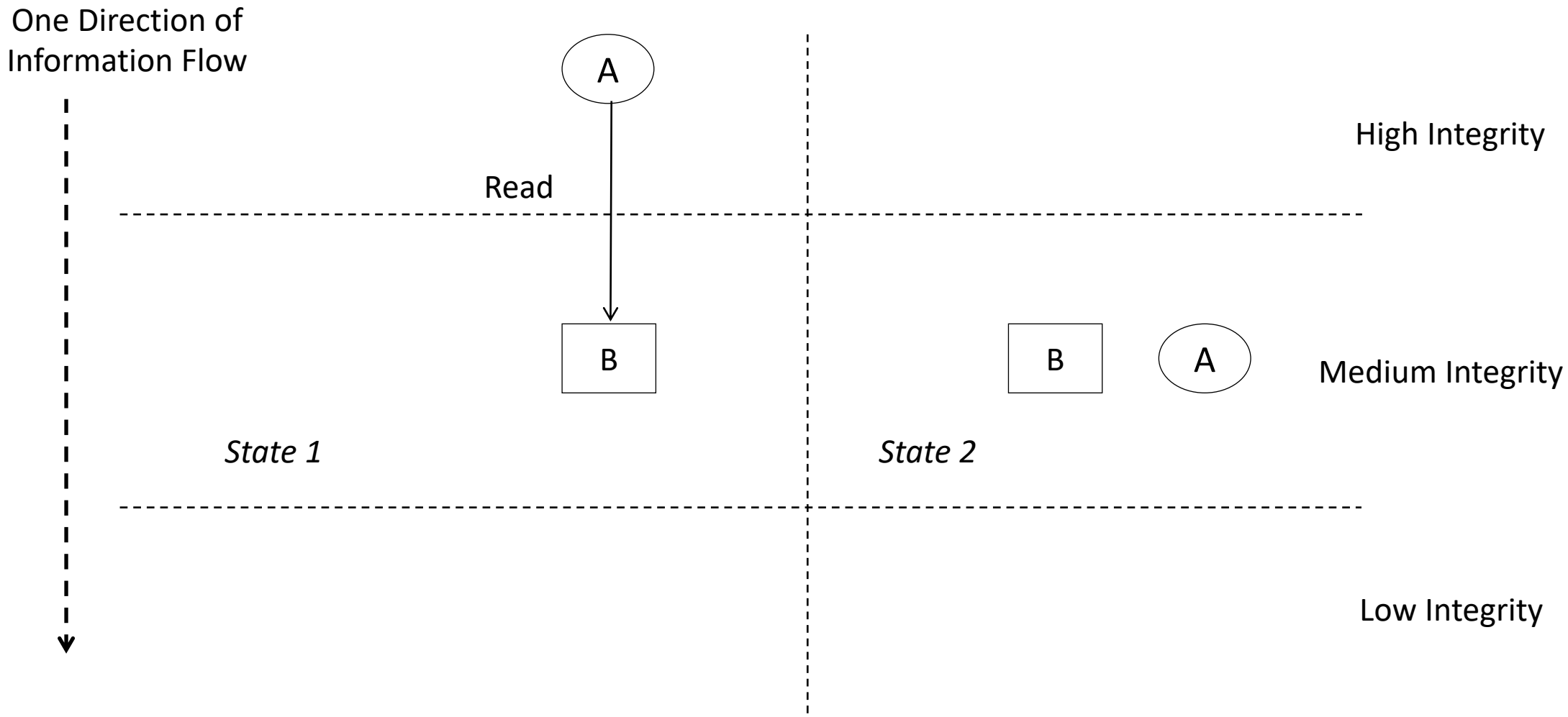
Week 9

Biba Model – Permissive Read Integrity Property



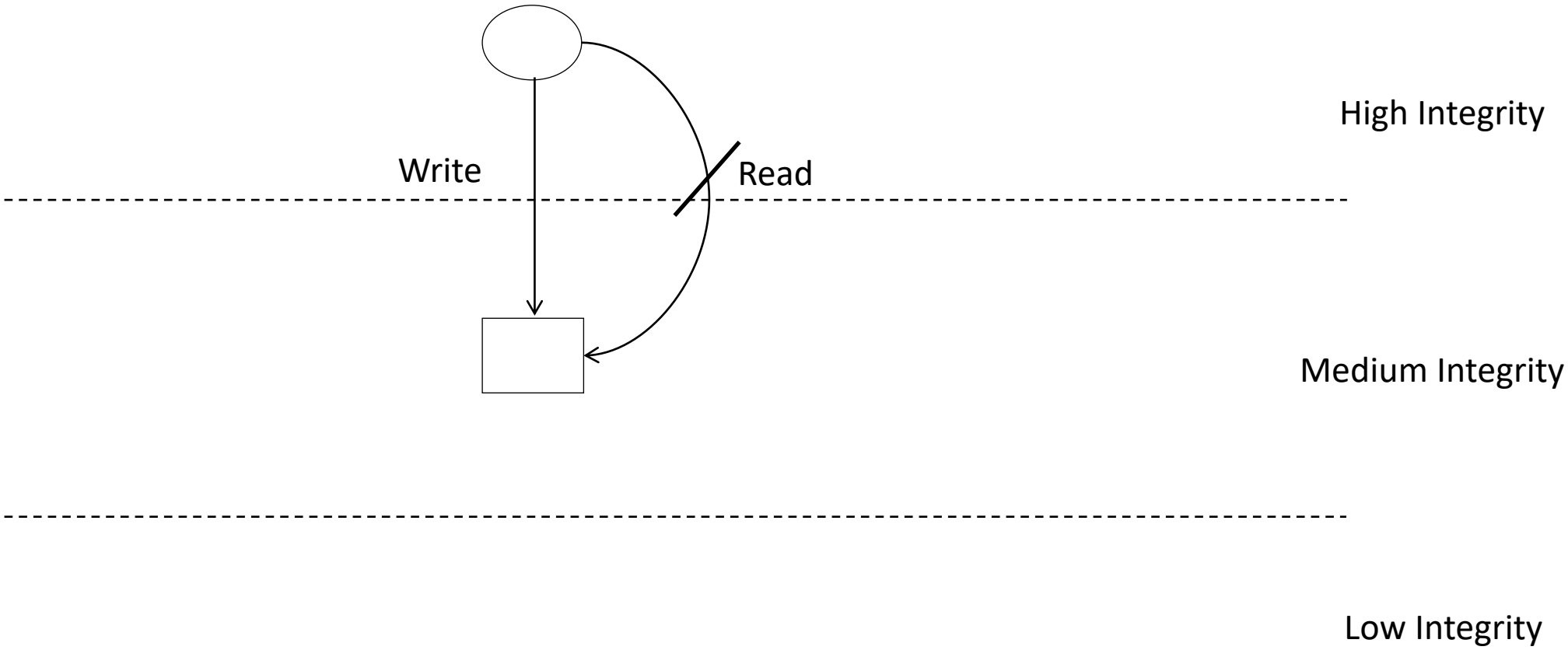
Week 9

Biba Model – Permissive Read Integrity Property



Week 9

Blind Write Problem – Exposed on Unix Implementation

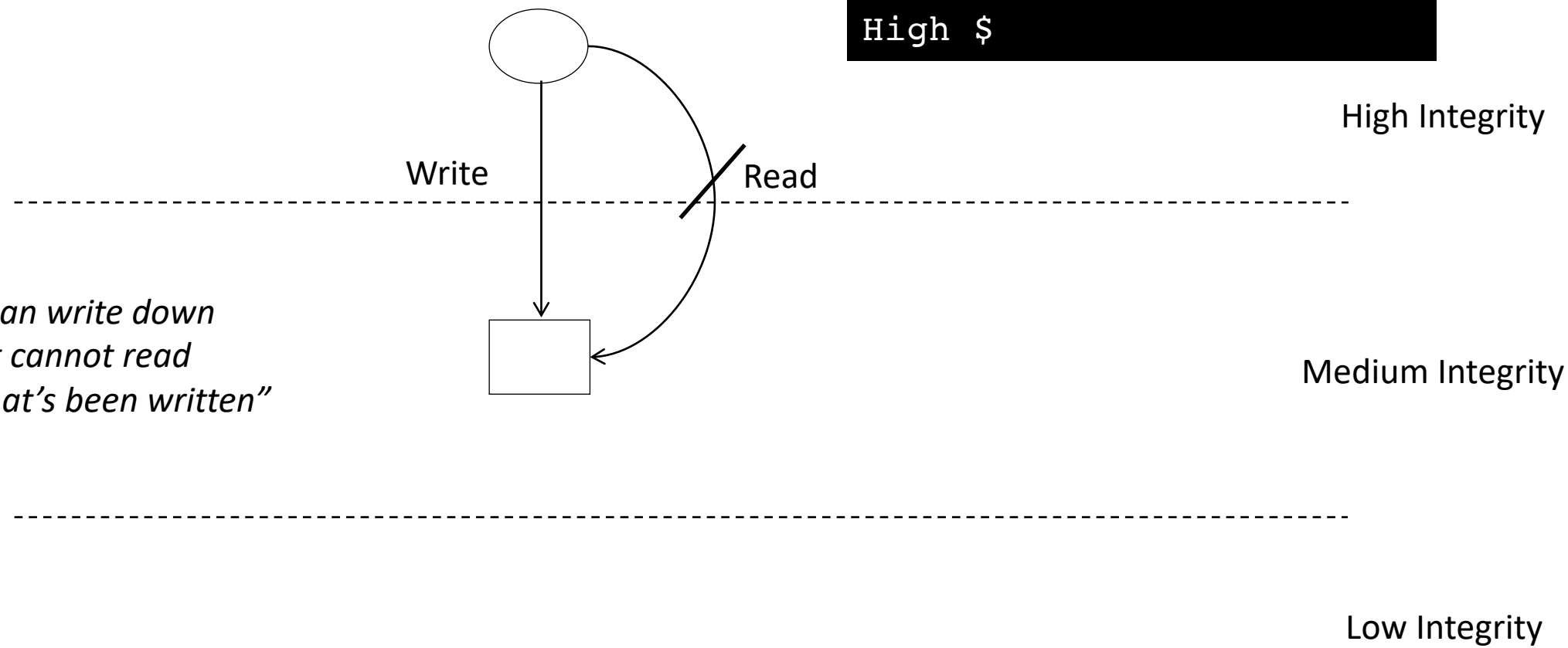


Week 9

Blind Write Problem – Exposed on Unix Implementation

"No Write Down"

```
High $ echo hello >> foo
High $ cat foo
     <file not found>
High $
```

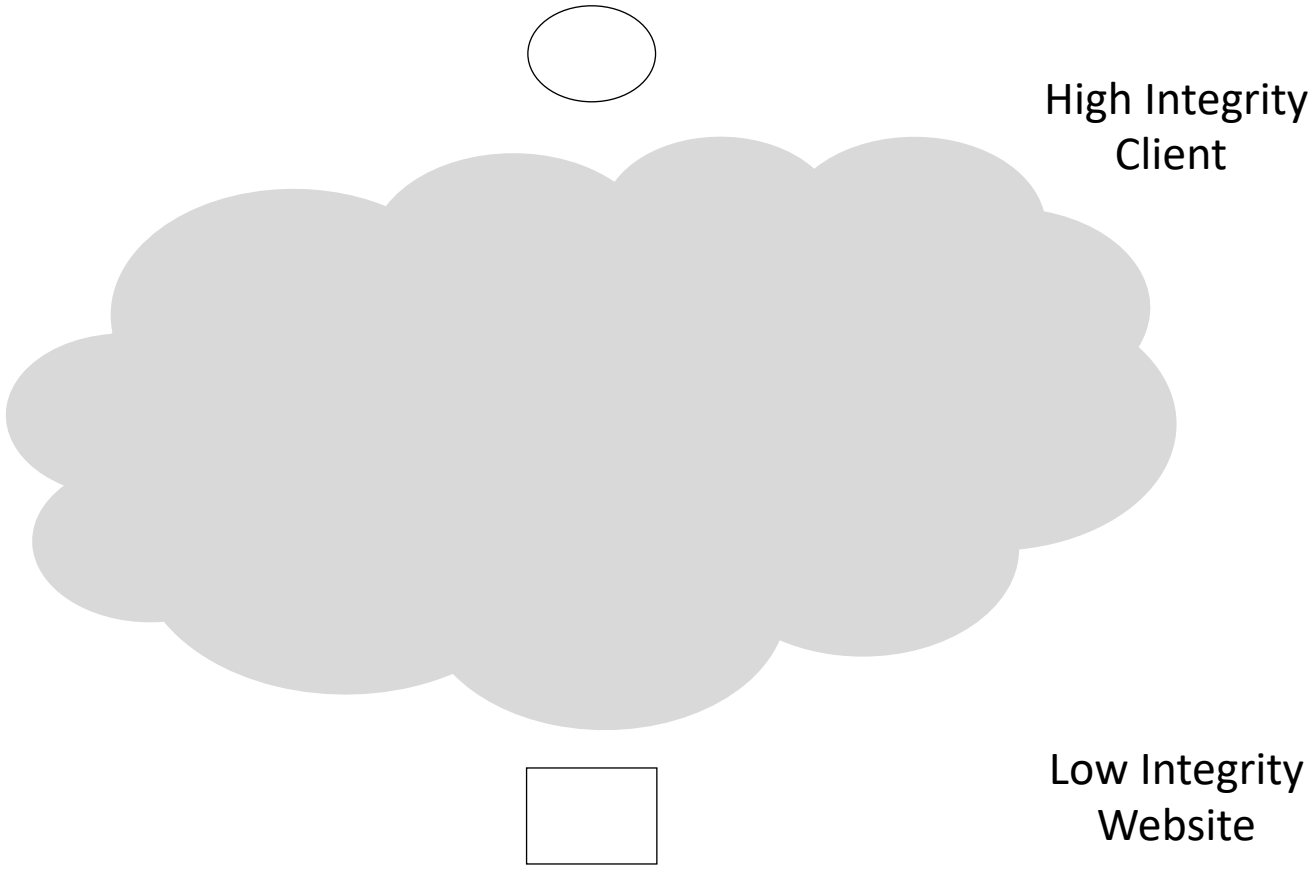


"Process can write down data, but cannot read down what's been written"

Week 9

Distributed Write Implementation Problem

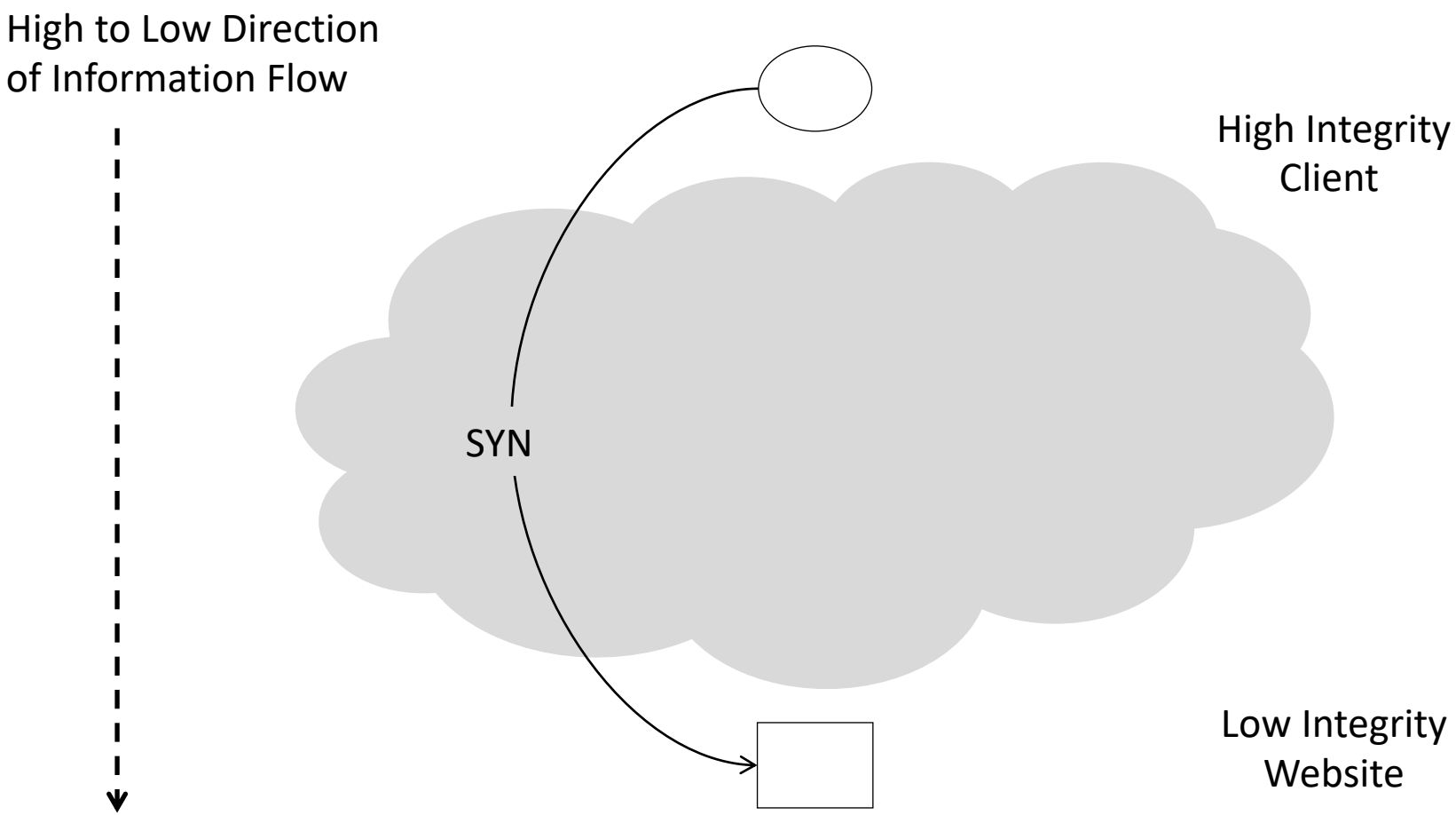
“No Write Up”



Week 9

Distributed Write Implementation Problem – TCP Protocol

"No Write Up"

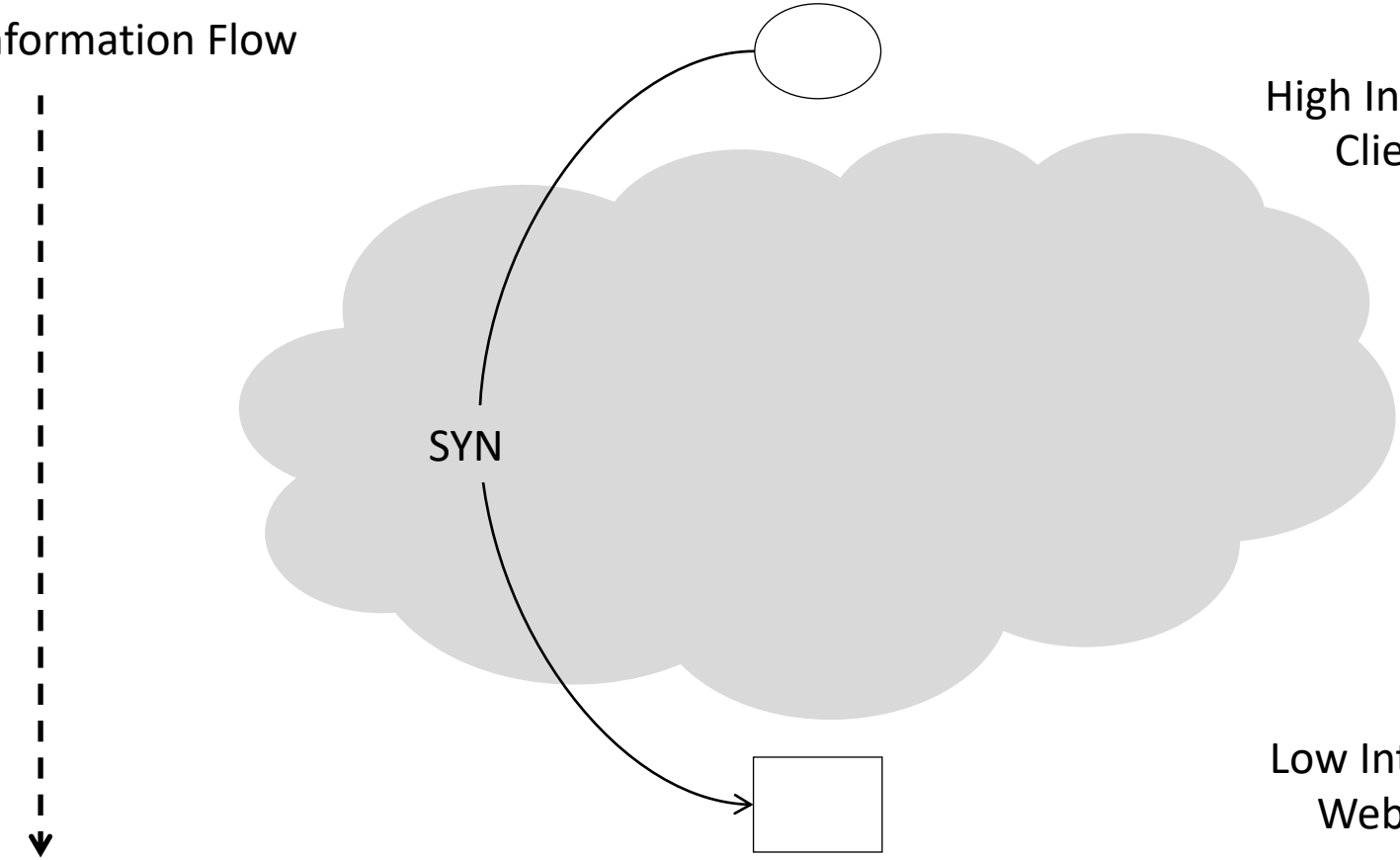


Week 9

Distributed Write Implementation Problem – TCP Protocol

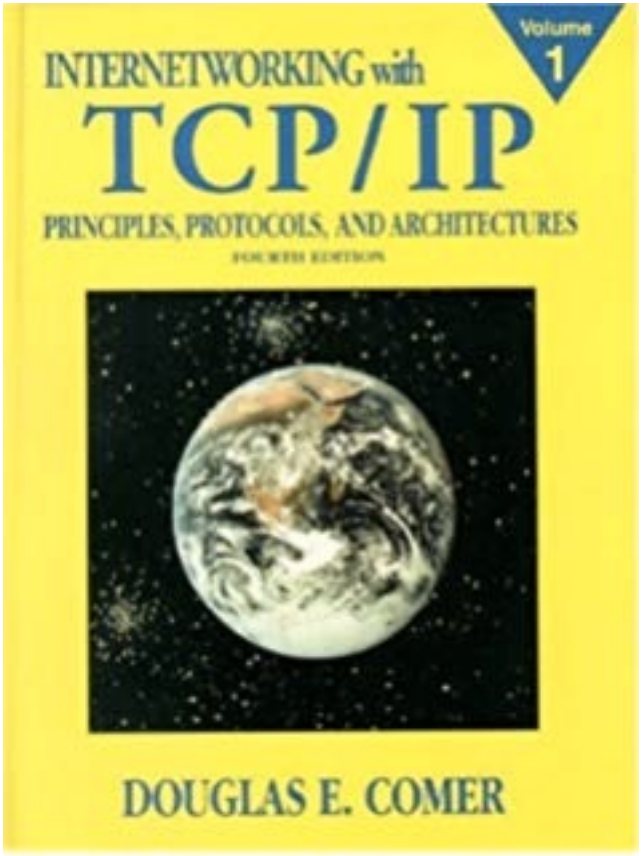
"No Write Up"

High to Low Direction of Information Flow



High Integrity Client

Low Integrity Website

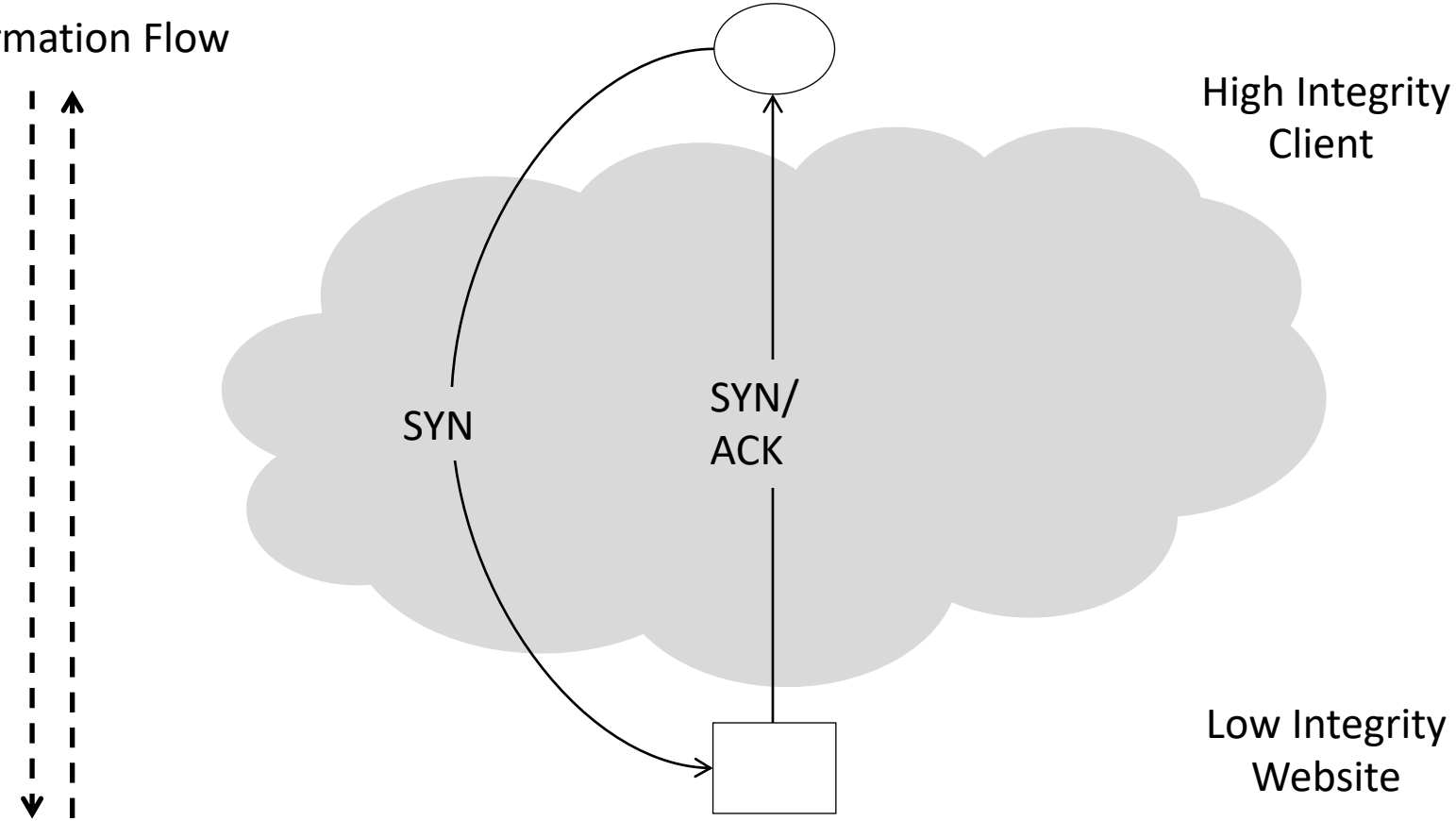


Week 9

Distributed Write Implementation Problem – TCP Protocol

"No Write Down"

Two Directions of Information Flow



High Integrity Client

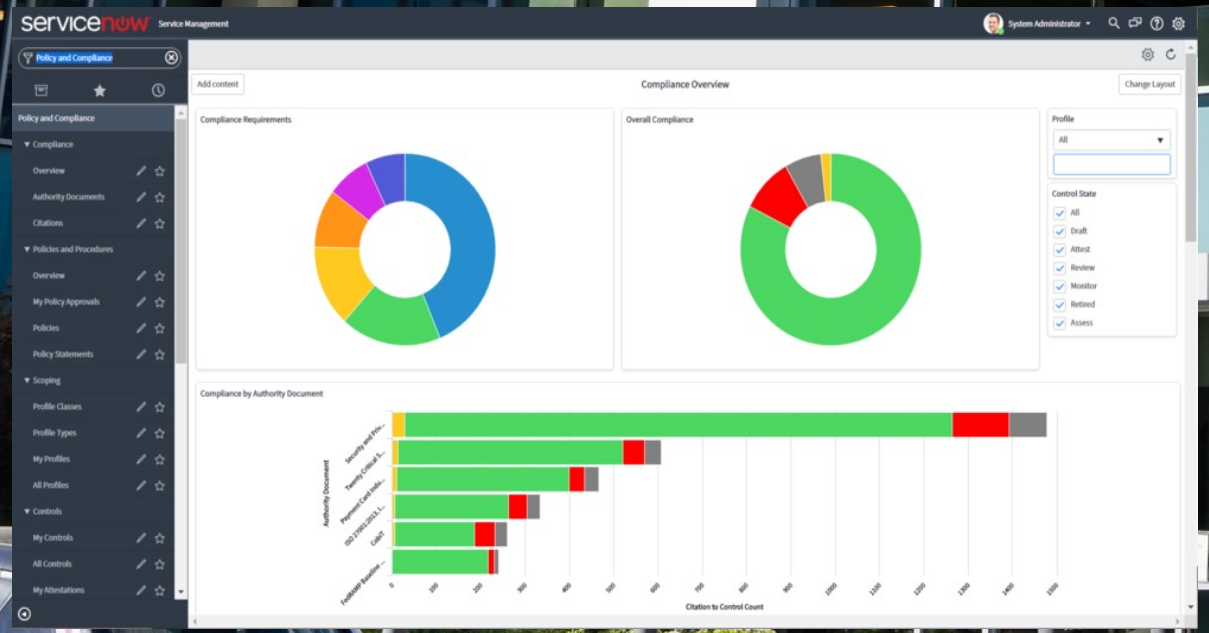
Low Integrity Website

Week 9

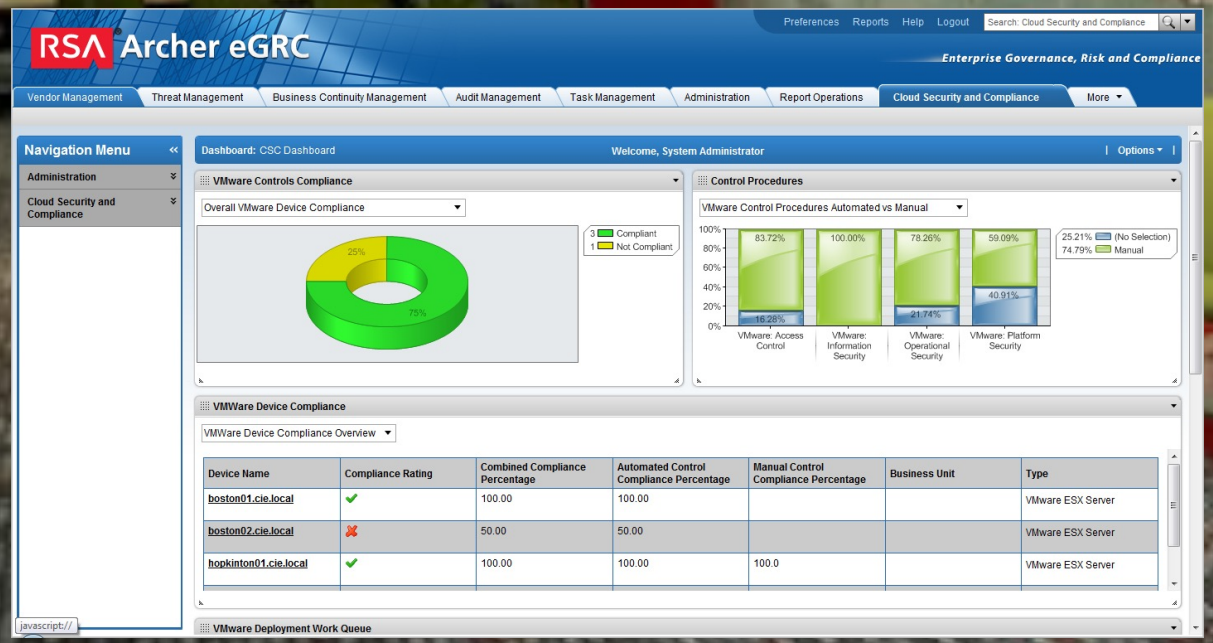

DEPARTMENT OF DEFENSE STANDARD
DEPARTMENT OF DEFENSE TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA
DECEMBER 1985
DEPARTMENT OF DEFENSE
PASSWORD MANAGEMENT
GUIDELINE
COMPUTER SECURITY REQUIREMENTS
NCE FOR APPLYING THE DEPARTMENT OF DEFENSE
STED COMPUTER SYSTEM EVALUATION CRITERIA
IN SPECIFIC ENVIRONMENTS
TECHNICAL RATIONALE BEHIND CSC-STD-003-85:
COMPUTER SECURITY REQUIREMENTS
GUIDANCE FOR APPLYING THE DEPARTMENT OF DEFENSE
TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA
IN SPECIFIC ENVIRONMENTS
NATIONAL COMPUTER SECURITY CENTER
NCSC-TG-011
VERSION-1
NCSC-TG-010
VERSION-1
NCSC-TG-009
VERSION-1
NCSC-TG-004
VERSION-1
NCSC-TG-002
VERSION-1
NCSC-TG-001
VERSION-2
A GUIDE TO
UNDERSTANDING
AUDIT
IN
TRUSTED SYSTEMS
1 June 1988
NCSC-TG-021
VERSION-1
NCSC-TG-020-A
VERSION-1
NCSC-TG-019
VERSION-1
NCSC-TG-016
VERSION-1
NCSC-TG-018
VERSION 1
NCSC-TG-008
VERSION-1
Guidelines
for
Formal Verification Systems
NCSC-TG-027
VERSION-1
NCSC-TG-028
VERSION 1
NCSC-TG-025
VERSION-2

What is Governance, Risk, and Compliance (GRC) and How is it Used to Enforce Policy?

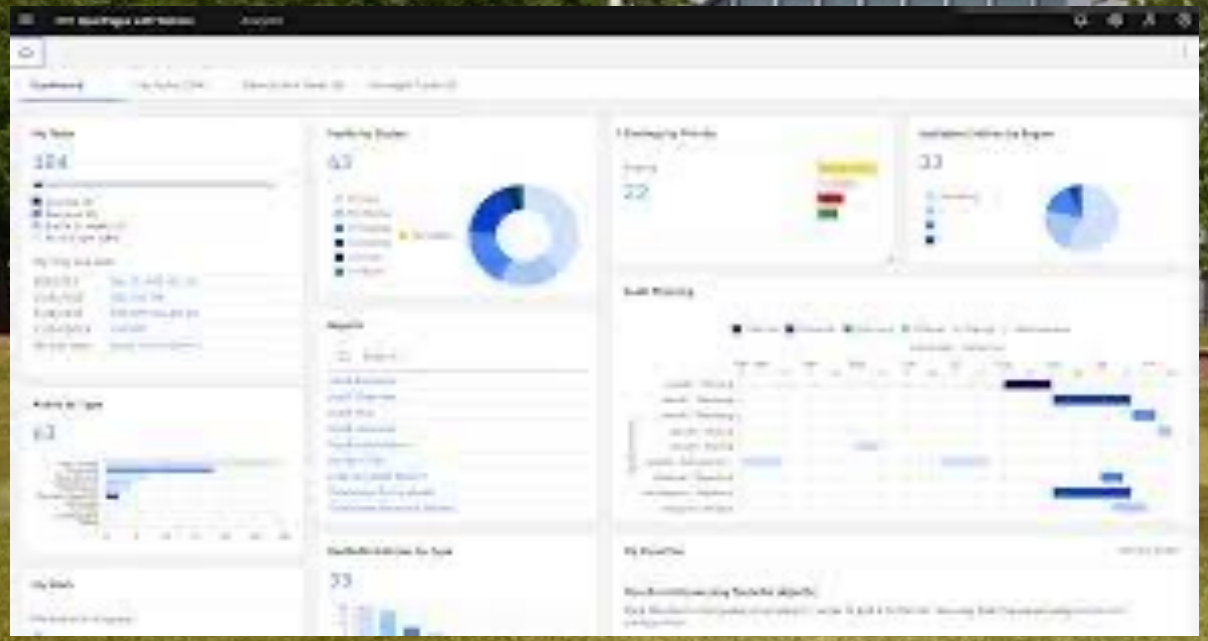
Week 9


servicenow
Service Management
System Administrator
Policy and Compliance
Add content
Compliance Overview
Change Layout
Compliance Requirements
Overall Compliance
Profile
All
Control State
All
Draft
Attest
Review
Monitor
Retired
Assess
Compliance by Authority Document
Authority Document
Citation to Control Count
Policy and Compliance
Compliance
Overview
Authority Documents
Citations
Policies and Procedures
Overview
My Policy Approvals
Policies
Policy Statements
Scoping
Profile Classes
Profile Types
My Profiles
All Profiles
Controls
My Controls
All Controls
My Attestations

Week 9


RSA Archer eGRC
Preferences Reports Help Logout
Search: Cloud Security and Compliance
Enterprise Governance, Risk and Compliance
Vendor Management
Threat Management
Business Continuity Management
Audit Management
Task Management
Administration
Report Operations
Cloud Security and Compliance
More
Navigation Menu
Administration
Cloud Security and Compliance
Dashboard: CSC Dashboard
Welcome, System Administrator
Options
VMware Controls Compliance
Overall VMware Device Compliance
3 Compliant
1 Not Compliant
25%
75%
Control Procedures
VMware Control Procedures Automated vs Manual
25.21% (No Selection)
74.79% Manual
100%
80%
60%
40%
20%
0%
83.72%
100.00%
78.26%
59.09%
16.28%
21.74%
40.91%
VMware: Access Control
VMware: Information Security
VMware: Operational Security
VMware: Platform Security
VMWare Device Compliance
VMWare Device Compliance Overview
Device Name
Compliance Rating
Combined Compliance Percentage
Automated Control Compliance Percentage
Manual Control Compliance Percentage
Business Unit
Type
boston01.cie.local
100.00
100.00
VMware ESX Server
boston02.cie.local
50.00
50.00
VMware ESX Server
hopkinton01.cie.local
100.00
100.00
100.0
VMware ESX Server
VMware Deployment Work Queue
javascript://

Week 9



Week 9

Governance, Risk, and Compliance (GRC)

Corporate Security Requirements:

R1: Passwords must be six characters long

R2: Manager must approve critical access requests

. . .

Corporate Security Team Types/Enters Requirements into GRC Tool

Governance, Risk, and Compliance (GRC) Tool
(e.g., RSA Archer, MetricStream, Allgress)

Week 9

Governance, Risk, and Compliance (GRC)

Corporate Security Requirements:

R1: Passwords must be six characters long

R2: Manager must approve critical access requests

. . .

Framework 1 Security Requirements:

R1: Passwords must be eight characters long

R2: Manager must approve all access requests

. . .

Corporate Security Team Downloads Framework 1 into GRC Tool

Governance, Risk, and Compliance (GRC) Tool

Week 9

Governance, Risk, and Compliance (GRC)

Corporate Security Requirements:

R1: Passwords must be **six** characters long

R2: Manager must approve **critical** access requests

. . .

Framework 1 Security Requirements:

R1: Passwords must be **eight** characters long

R2: Manager must approve **all** access requests

. . .

Governance, Risk, and Compliance (GRC) Tool

Week 9

Governance, Risk, and Compliance (GRC)

Corporate Security Requirements:

R1: Passwords must be **eight** characters long

R2: Manager must approve **all** access requests

. . .

Framework 1 Security Requirements:

R1: Passwords must be **eight** characters long

R2: Manager must approve **all** access requests

. . .

Security Team

Entire Company

Governance, Risk, and Compliance (GRC) Tool

Week 9

Governance, Risk, and Compliance (GRC)

Corporate Security Requirements:

R1:Password “eight”

R2: Access “all”

. . .

Framework 1 Requirements:

R1:Password “eight”

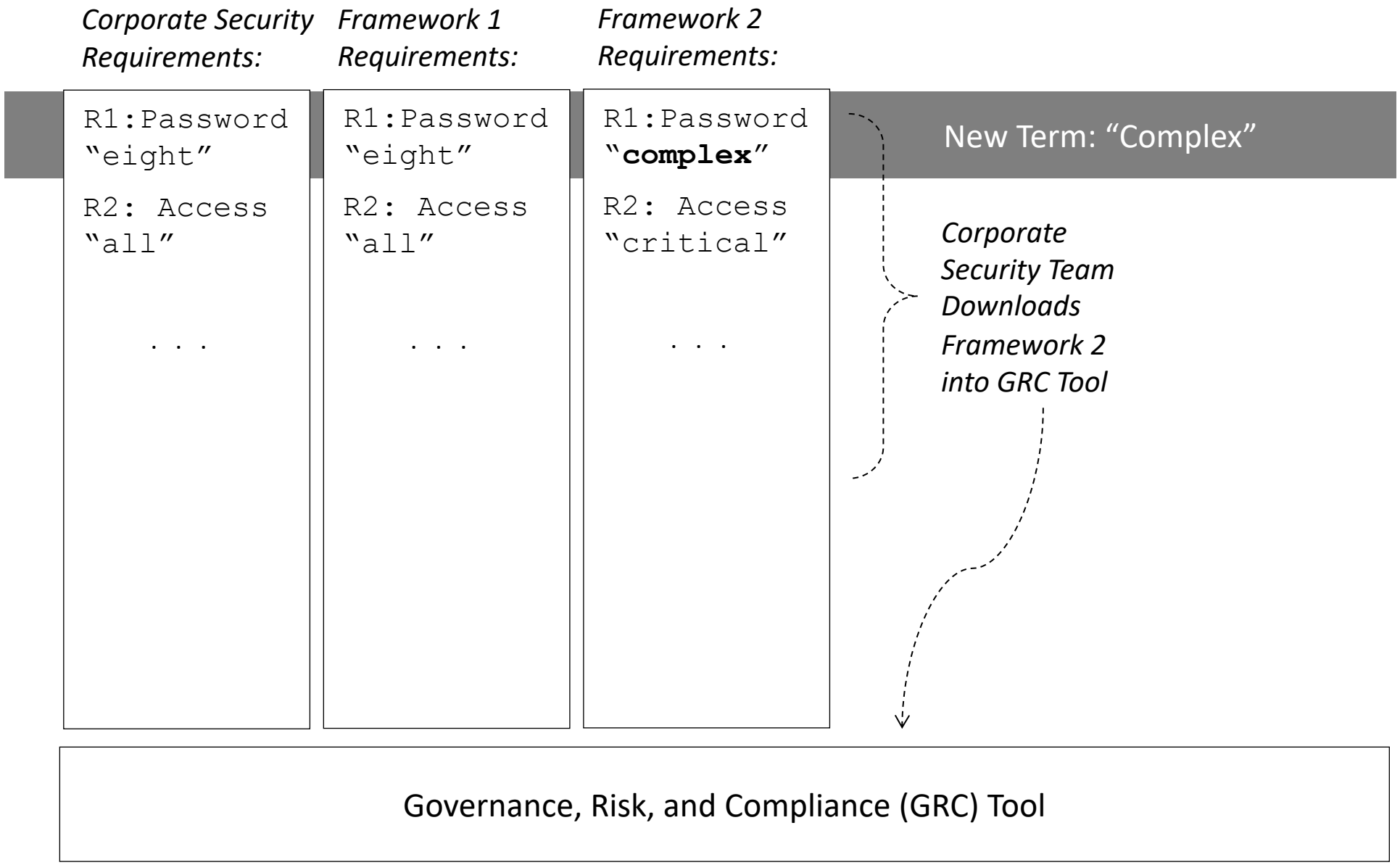
R2: Access “all”

. . .

Governance, Risk, and Compliance (GRC) Tool

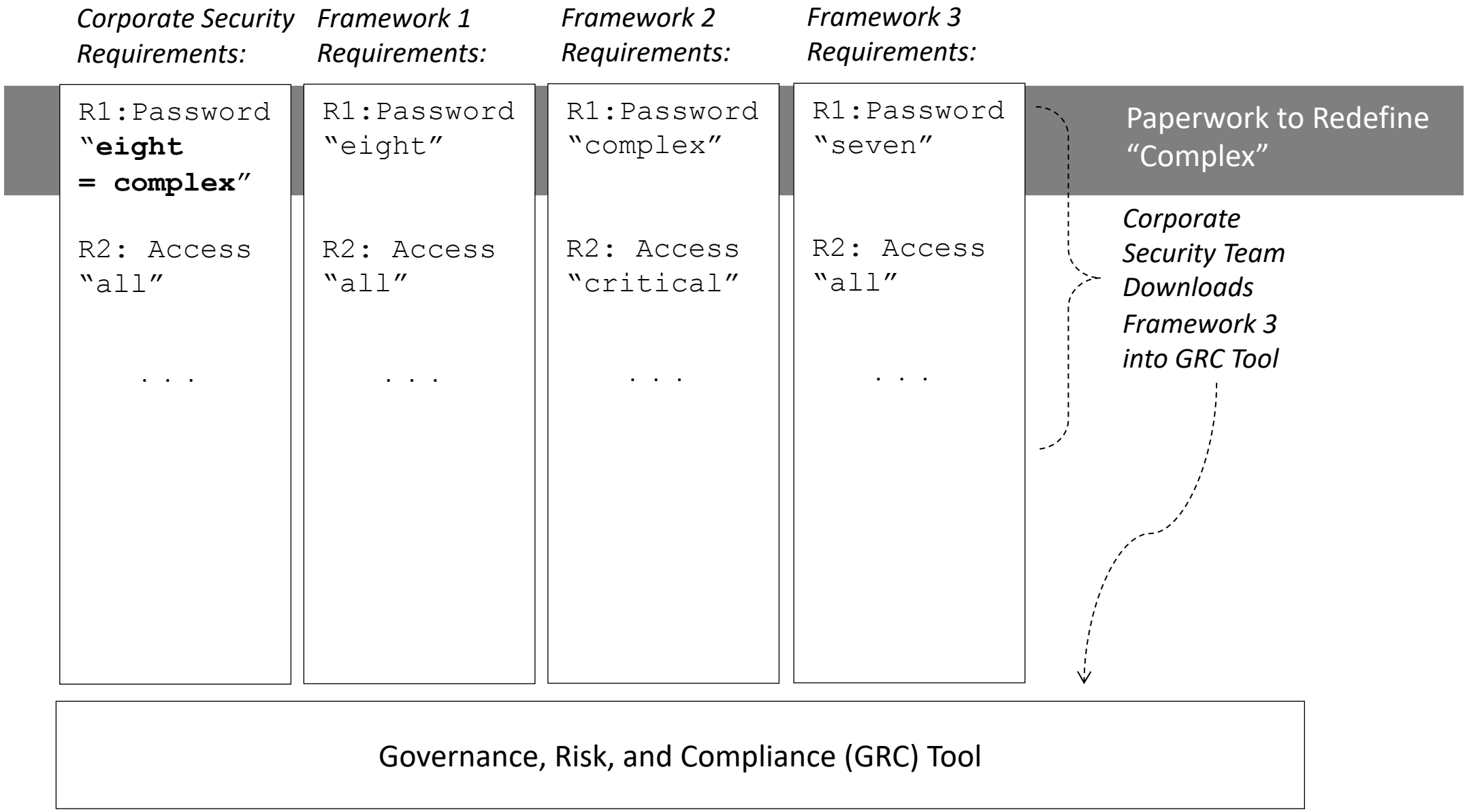
Week 9

Governance, Risk, and Compliance (GRC)



Week 9

Governance, Risk, and Compliance (GRC)



Week 9

Sample Cyber Security Compliance Frameworks

- **ISO 27K** – Series of international security management standards required by many global firms
- **Sarbanes-Oxley** – Control framework for auditing American public corporations to reduce fraud
- **PCI-DSS** – Data security standards framework designed to reduce retail point of sale security risk
- **SAS 70/SSAE 16/SOC** – Evolving standards for data center security with emphasis on internal controls
- **BITS** – Financial institution Shared Assessment Program for evaluating security controls of IT providers
- **NIST Cybersecurity Framework** – US framework that provides structure for cyber security programs
- **OWASP Open Cybersecurity Framework** – Includes top ten critical application security vulnerabilities
- **FedRAMP** – US civilian government security compliance frameworks and assessment programs
- **201 CMR 17.00 (Mass.)** – Example US state compliance and audit checklist and framework (1 of 50)
- **NERC 1300** – US cyber security standard for nuclear-related critical infrastructure controls/protection
- **HIPAA** – Health insurance portability and accountability designed to protect individual medical privacy
- **Common Criteria** – ISO/IEC protection profile 15408 for computer security certification of products
- **FISMA** – US legislation defining comprehensive framework to protect government information and assets
- **ISA/IEC-62443** – System network/system security standards for industrial automation and control systems
- **GLBA** – Financial services modernization act includes standards for financial systems security and protection
- **IASME** – Information assurance and cyber security standard for small and medium sized businesses
- **SANS/CIS Critical Security Controls** – Center for Internet Security list of effective cyber defense controls
- **ITIL** – Information Technology Infrastructure Library of best practices including cyber security protections
- **COBIT** – COBIT 5 from ISACA for day-to-day enterprise information security guidance and protection
- **ISF Standard of Good Practice** – Information Security Forum good practice standard for cyber security
- **CMU OCTAVE** – CMU framework used to assess an organization’s information security practices
- **GDPR** – EU-based framework for data protection, security assurance, and information security.

How Did Subject-Object Models Evolve? (Hint: Deducibility Security)

Week 9


NAVAL RESEARCH LABORATORY
NAVAL RESEARCH LABORATORY
NAVAL RESEARCH LABORATORY WASHINGTON, D.C.

Week 9

Research Consequential to Enterprise Connectivity

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 16, NO. 6, JUNE 1990

A Hookup Theorem for Multilevel Security

DARYL MC CULLOUGH

Abstract—**In this paper, the author describes a security property for trusted multilevel systems, *restrictiveness*, which restricts the inferences a user can make about sensitive information. This property is a *hookup property*, or *composable*, meaning that a collection of secure restrictive systems when hooked together form a secure restrictive composite system. The author argues that the inference control and composability of restrictiveness make it an attractive choice for a security policy on trusted systems and processes.**

Index Terms—**Composable, information, process, security.**

I. Introduction

MULTILEVEL security requires that sensitive information be disclosed only to authorized personnel. In the ''paper world,'' this is enforced by assigning to each document and each employee a *security level* indicating sensitivity and authority. Commonly used levels in the government are *unclassified*, *confidential*, *secret*, and *top_secret*. The levels form a partially ordered set, so that an employee can be said to be authorized to read a document only if his level is greater than or equal to that of the document.

For information processing systems multilevel security becomes more complicated, because not all information is in the form of documents and not all consumers of information are employees. Generally for such systems the problem of multilevel security consists of two aspects:

1) *Access control*—determining who can see information of a given sensitivity leaving the system.

2) *Correct labeling*—determining the sensitivity of in-

or not), people instead try to build computer systems which are secure even in the presence of malicious programs.

In this paper we will describe a security property that addresses these issues, called *restrictiveness*, which is *composable*. This means that for a collection of trusted processes ''hooked up'' to make a system, or for a collection of system ''hooked up'' to make a network, the system or network is secure if each component is secure.

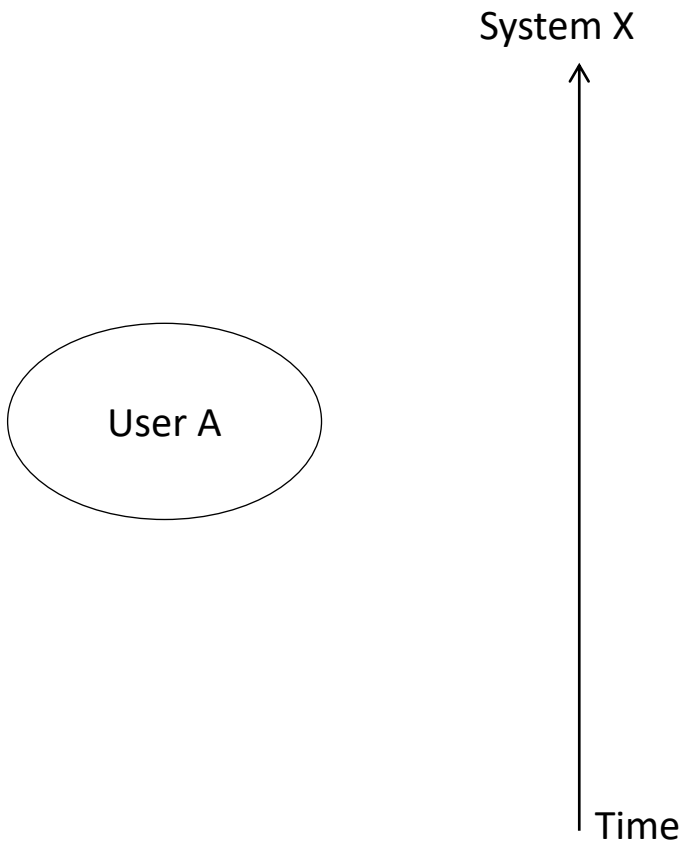
Using restrictiveness as a definition of security for trusted systems provides confidence in building large, complex systems from smaller, easier to verify trusted components. Security is modularized and so becomes more manageable. We first consider two earlier models for security, and point out some of their shortcomings.

A. The Bell–LaPadula Model: Trusted and Untrusted Processes

The issues of access control and correct labeling are addressed by the Bell–LaPadula model [1]. In this model, all entities involved with a computer system—users, files, programs, etc.—are divided into two classifications: *subjects* and *objects*. Subjects are the active entities, such as users and processes, which are capable of reading and modifying system state information, while objects are passive containers for information, such as files. Subjects which are processes are further divided into *trusted processes* and *untrusted processes*.

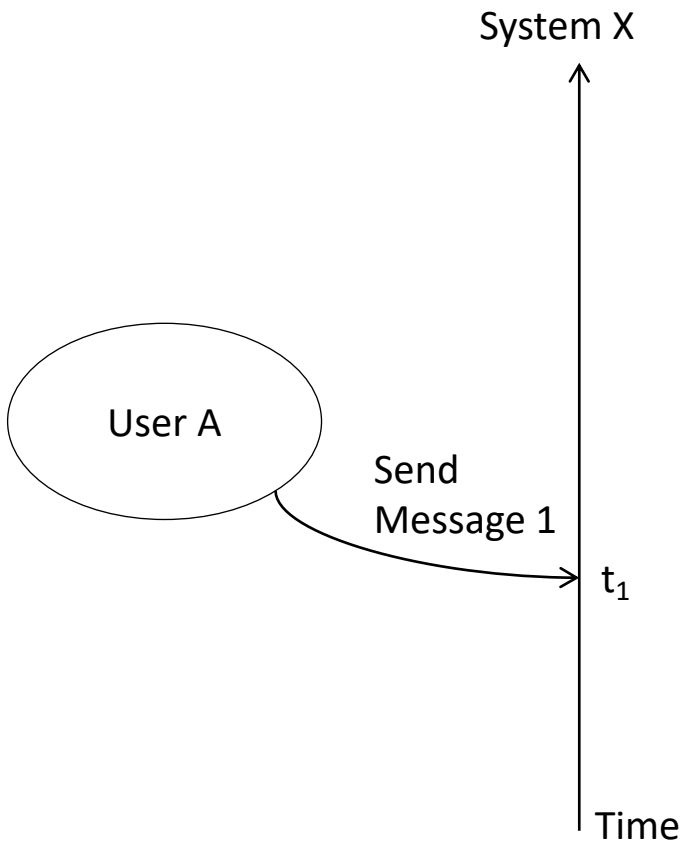
Week 9

User A and System X



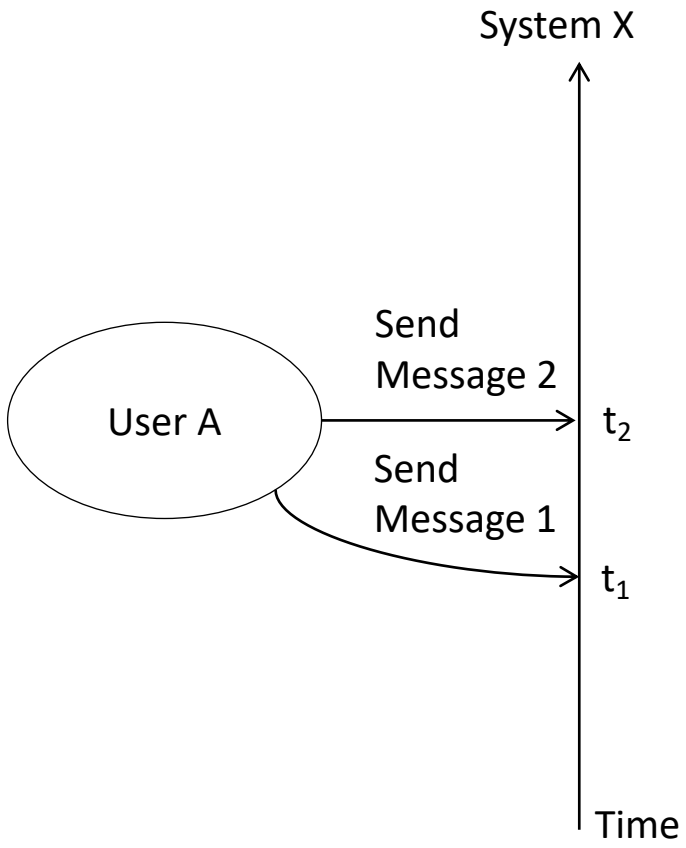
Week 9

User A Sends a Message



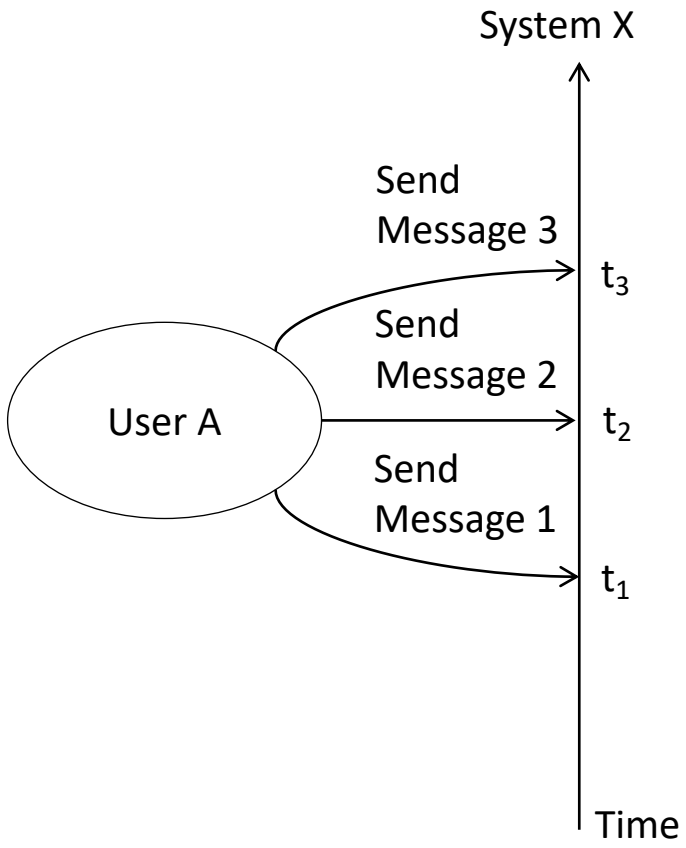
Week 9

User A Sends a Second Message



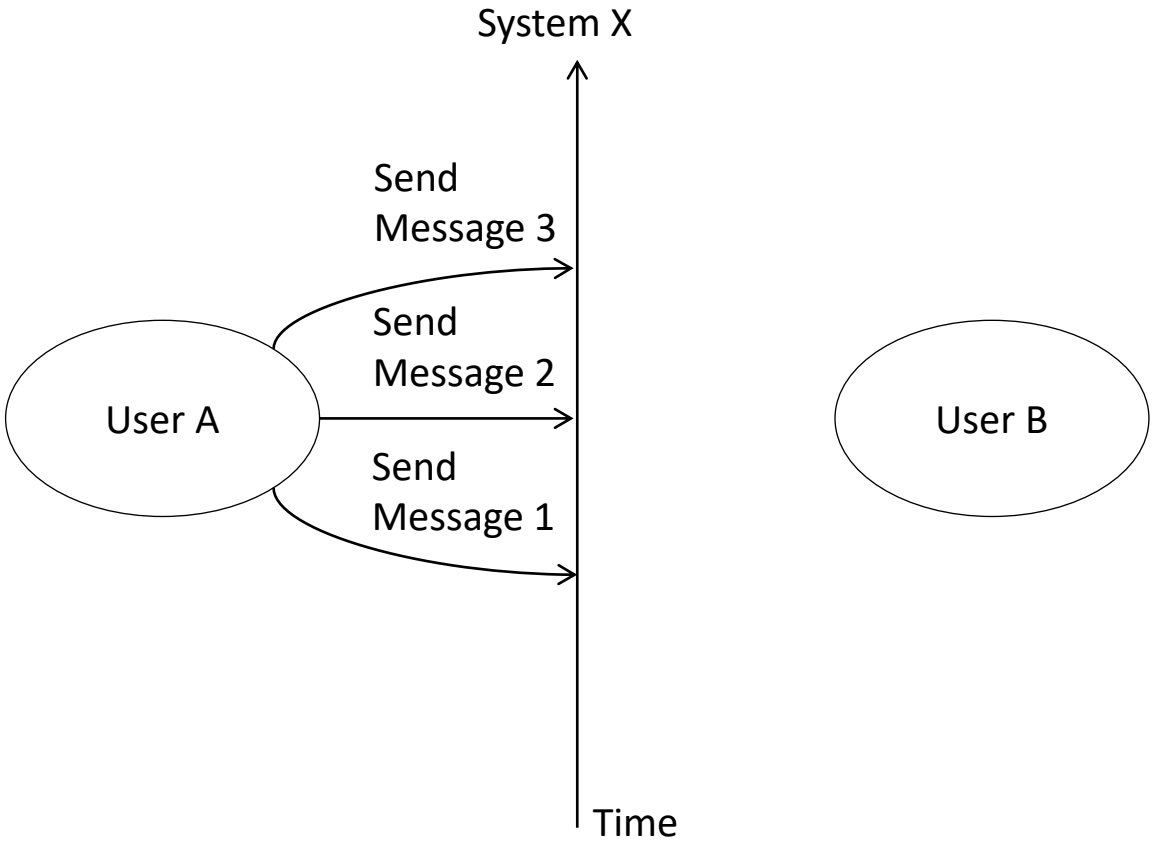
Week 9

User A Sends a Third Messages



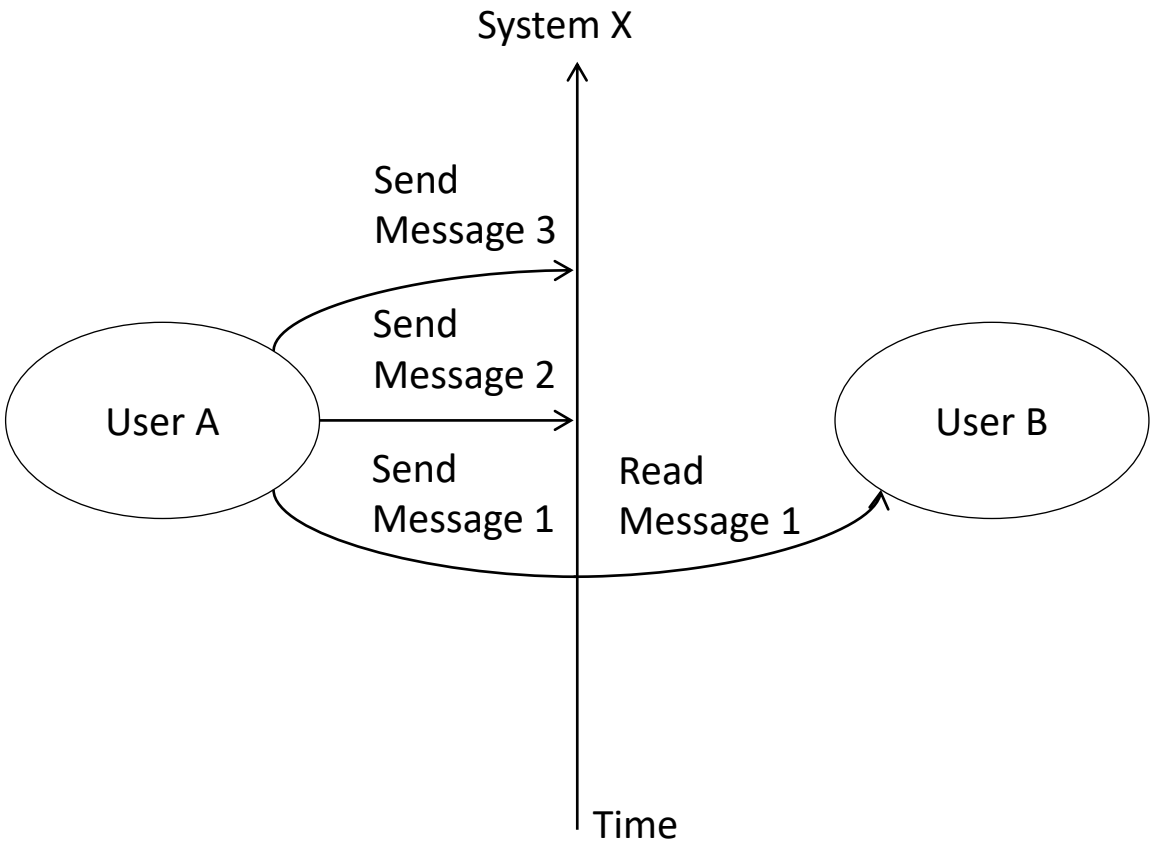
Week 9

User B and System X



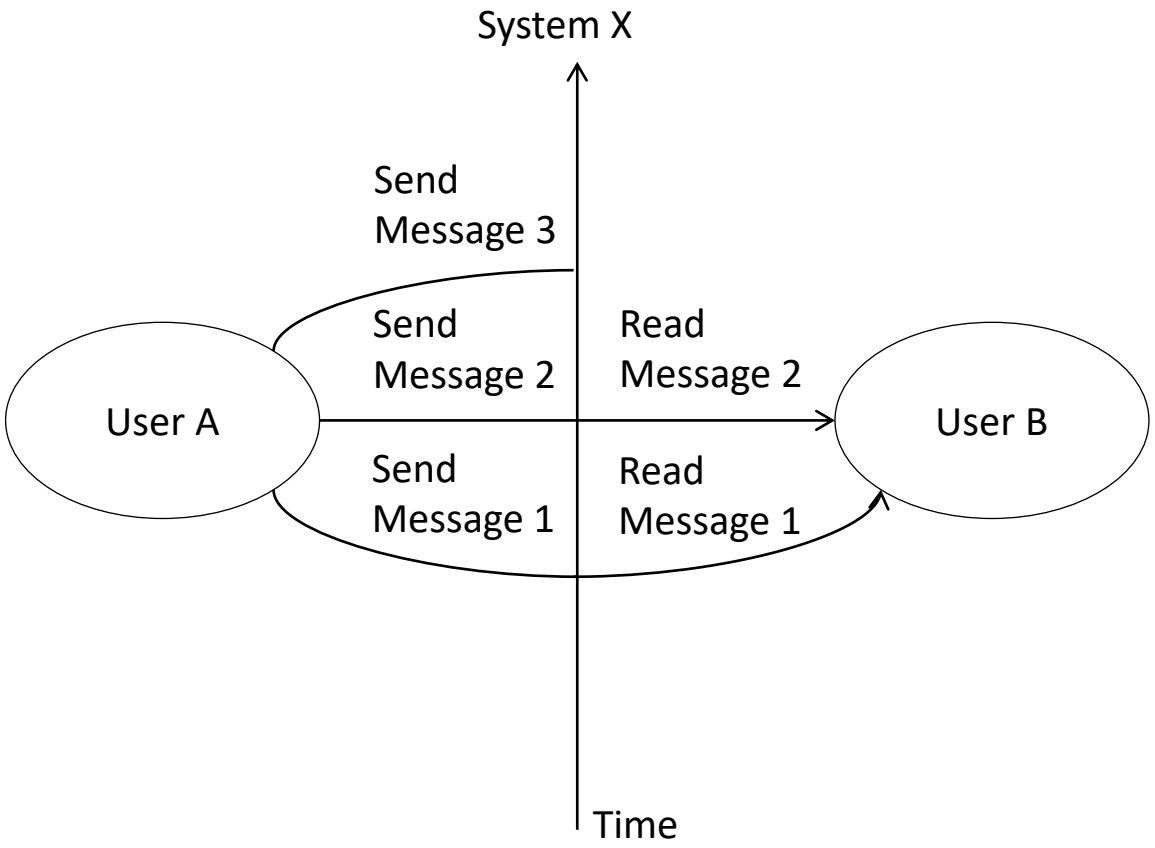
Week 9

User B Reads User A's Messages



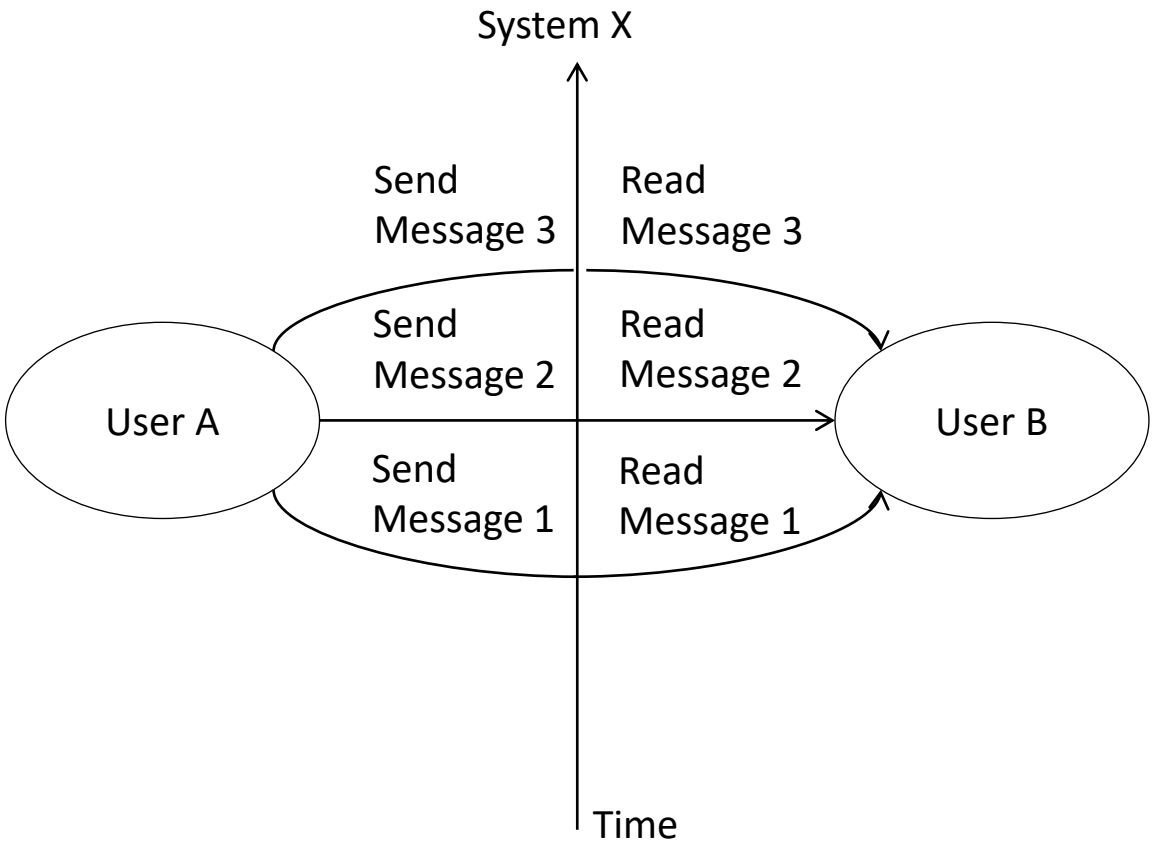
Week 9

User B Reads User A's Messages



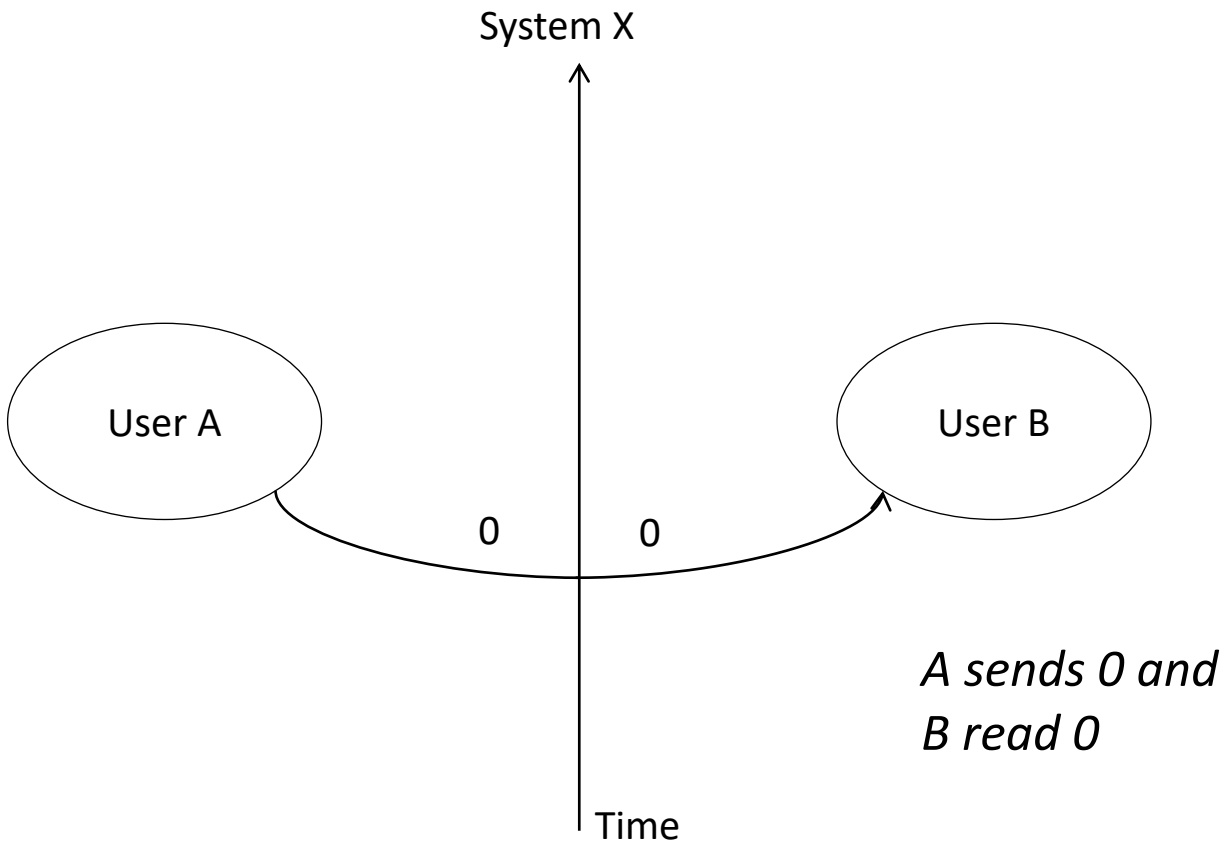
Week 9

User B Reads User A's Messages



Week 9

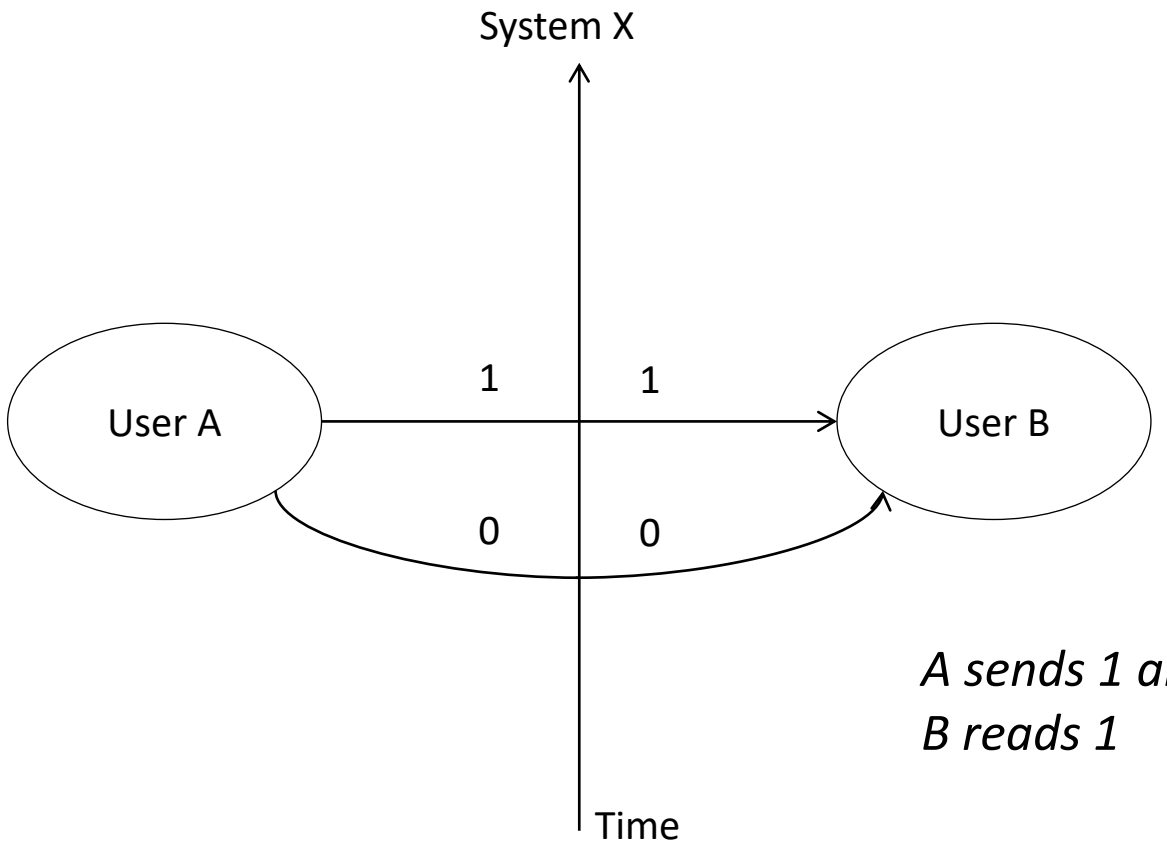
System X: Overt Channel Between Users A and B



A sends 0 and
B read 0

Week 9

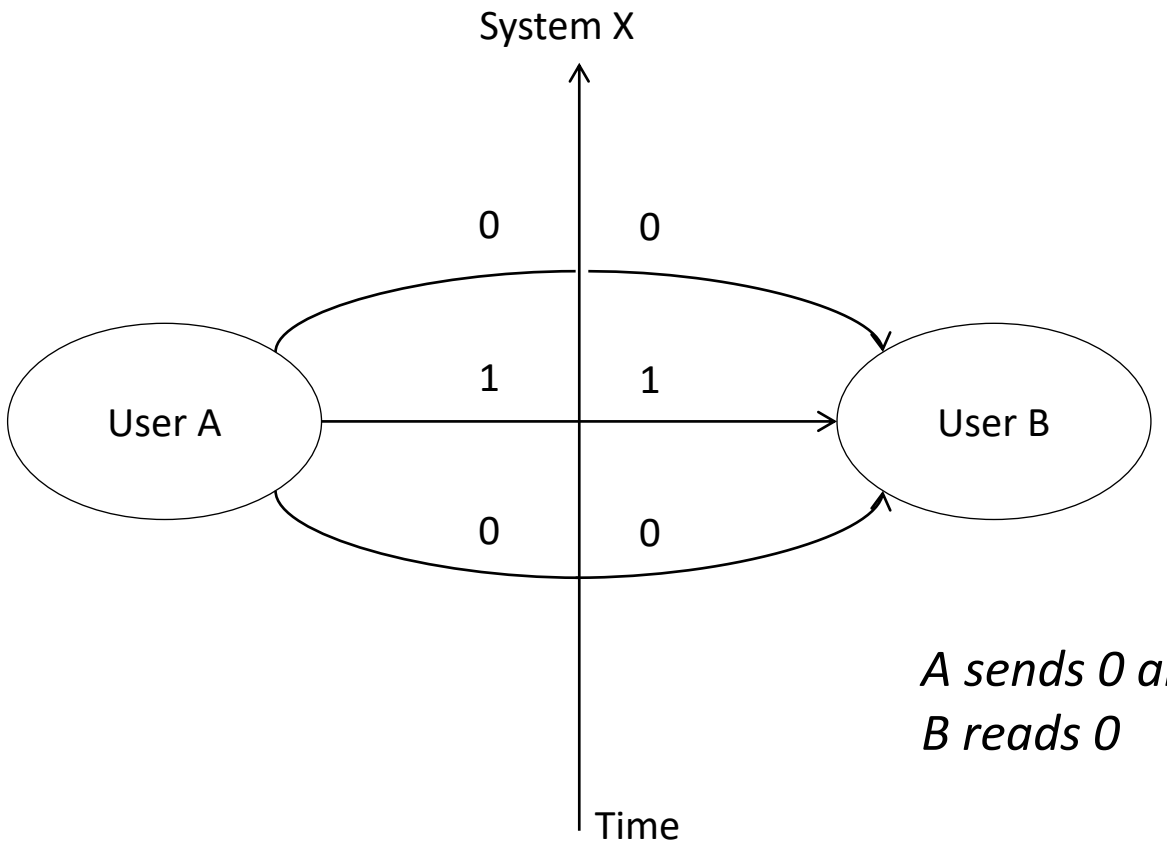
System X: Overt Channel Between Users A and B



A sends 1 and
B reads 1

Week 9

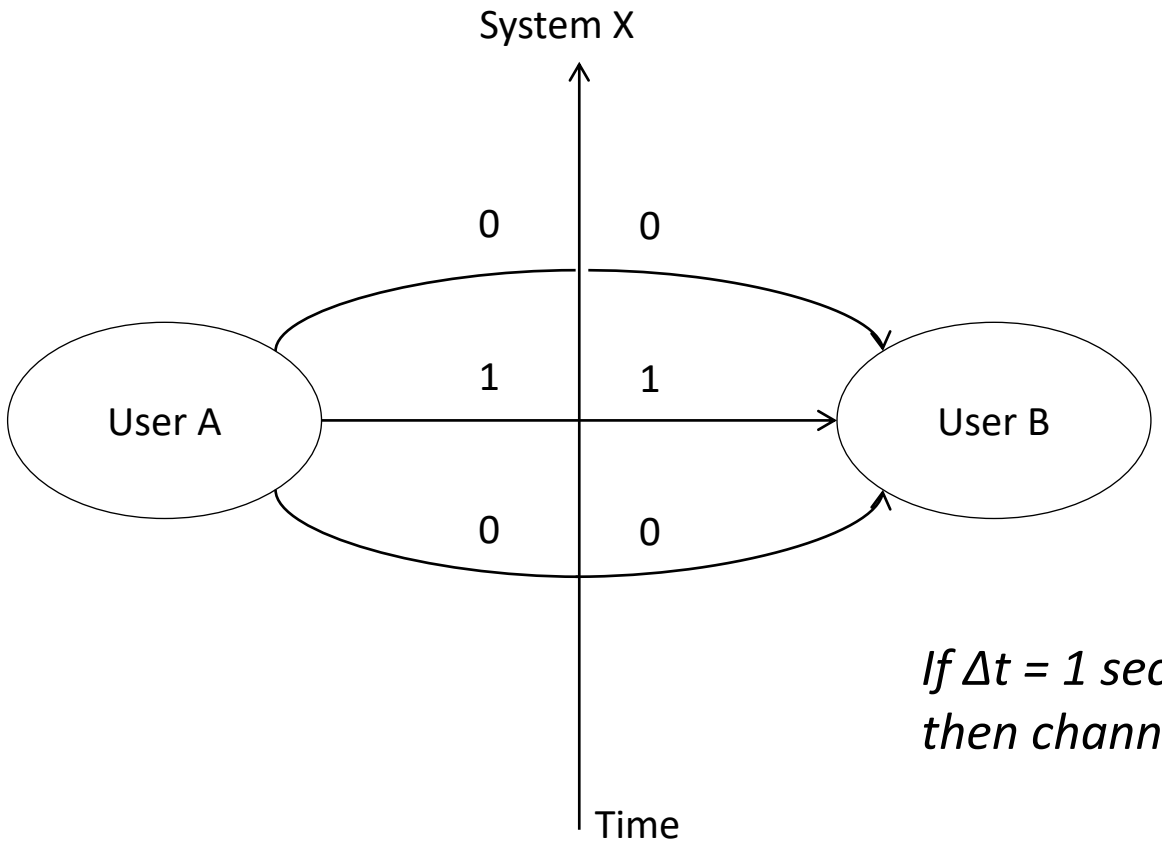
System X: Overt Channel Between Users A and B



A sends 0 and
B reads 0

Week 9

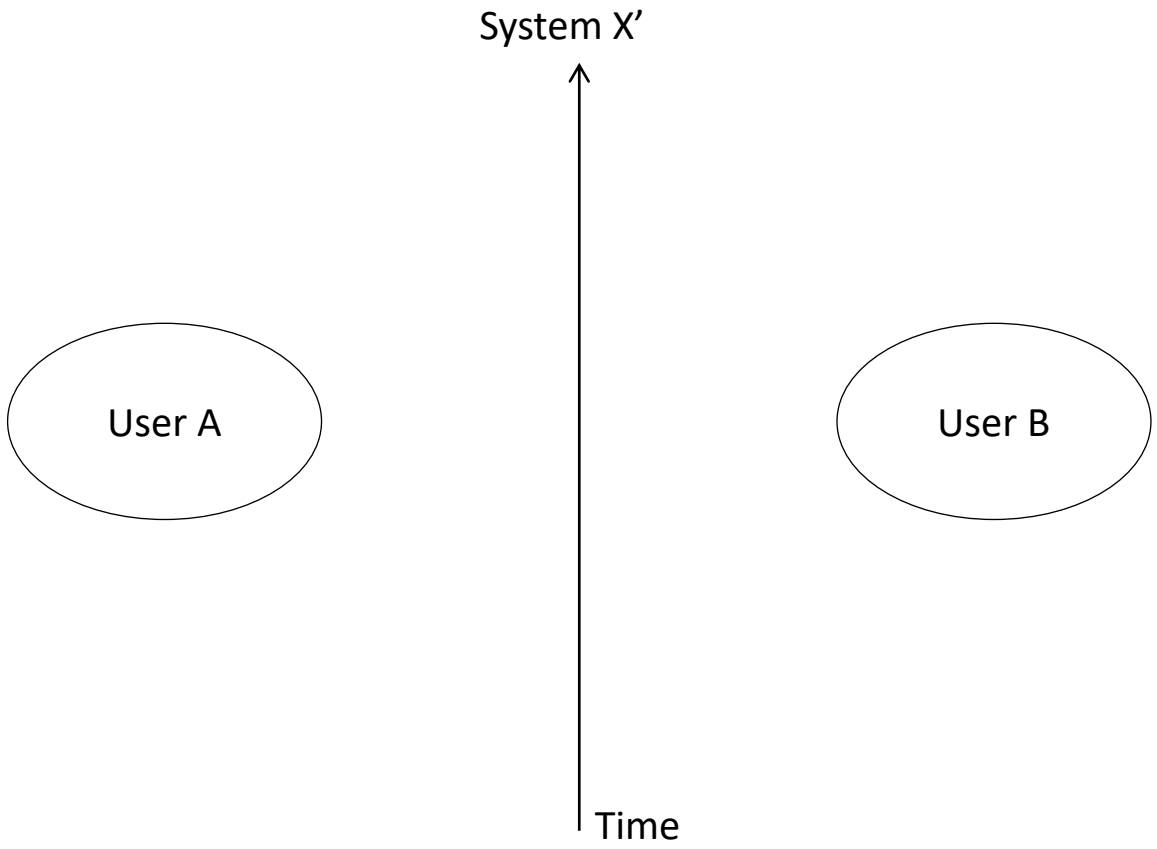
System X: Overt Channel Between Users A and B



If Δt = 1 second,
then channel is 1 bps

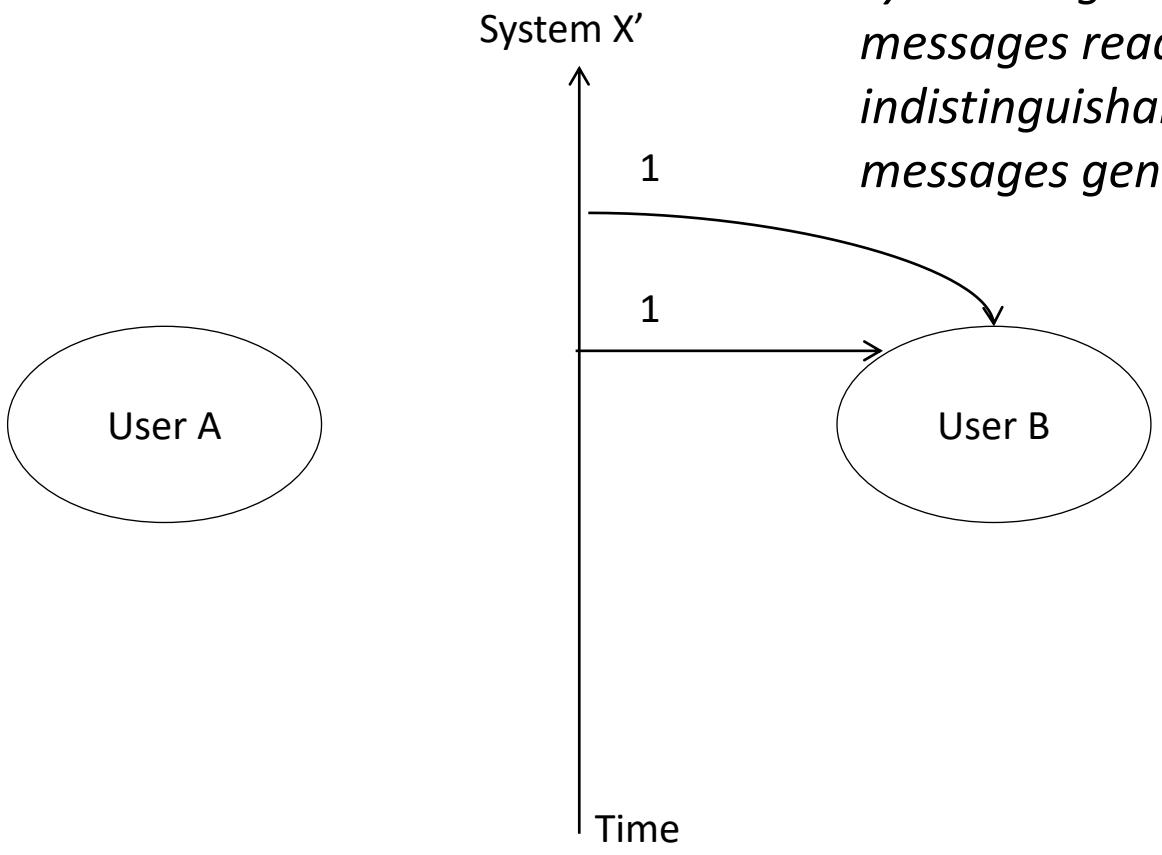
Week 9

New System X’



Week 9

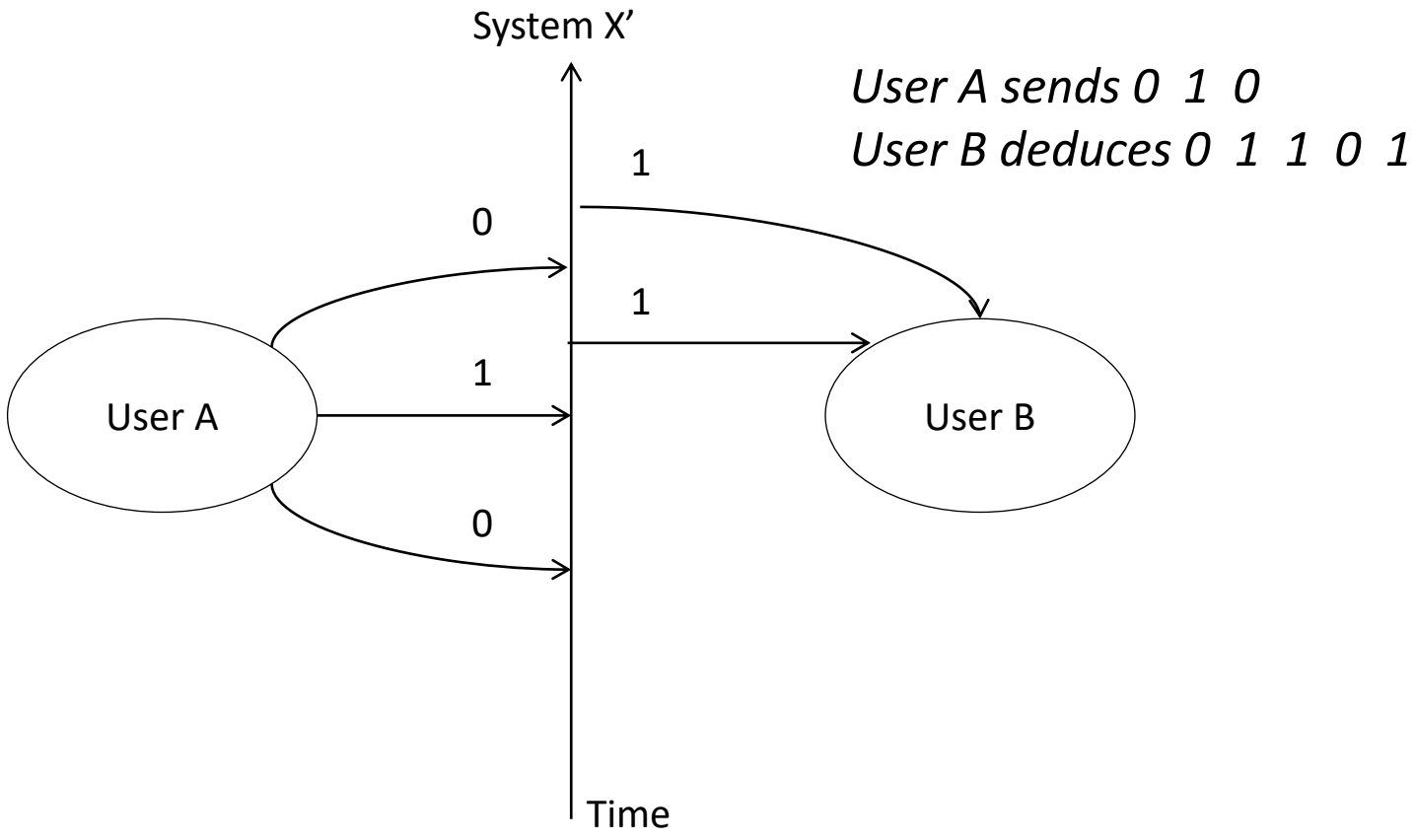
New System X' Generates Noise



System X' generates random "noise" messages readable by User B, but indistinguishable from any deliberate messages generated by User A

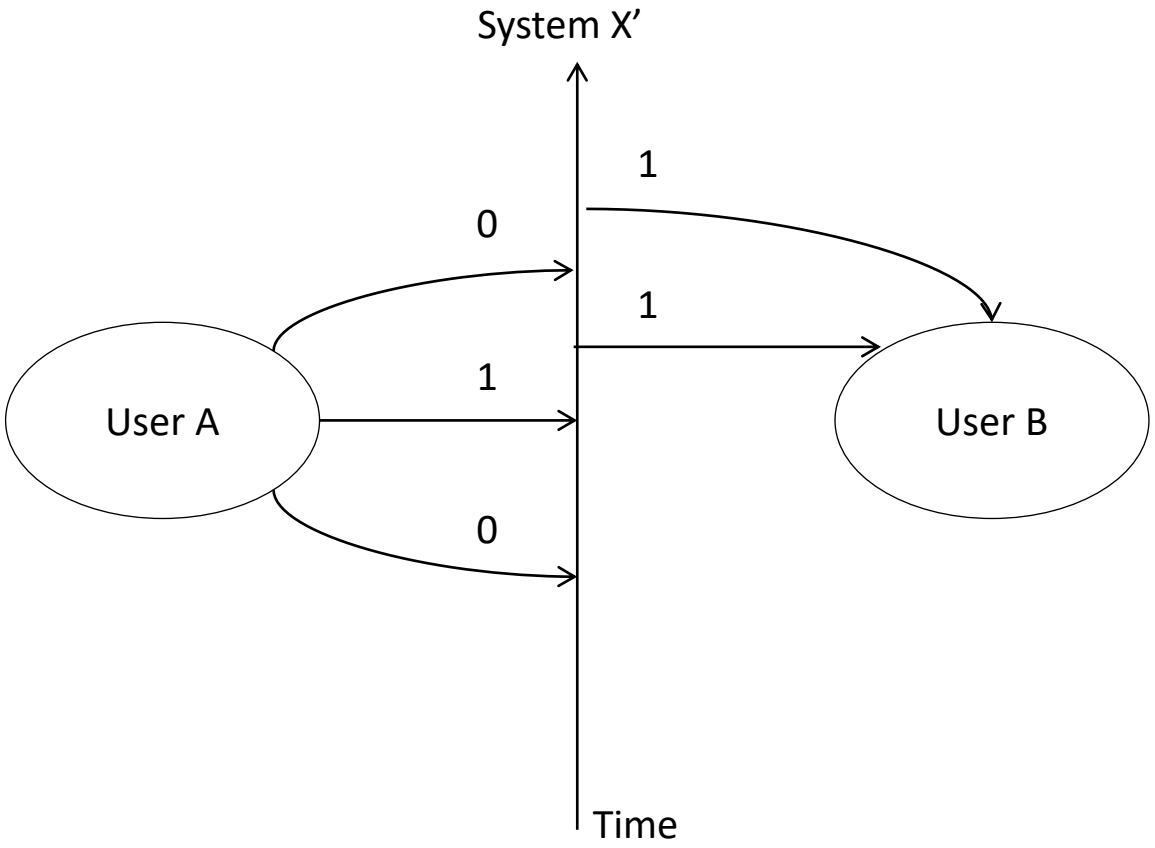
Week 9

User B Reads Messages from A and System X’



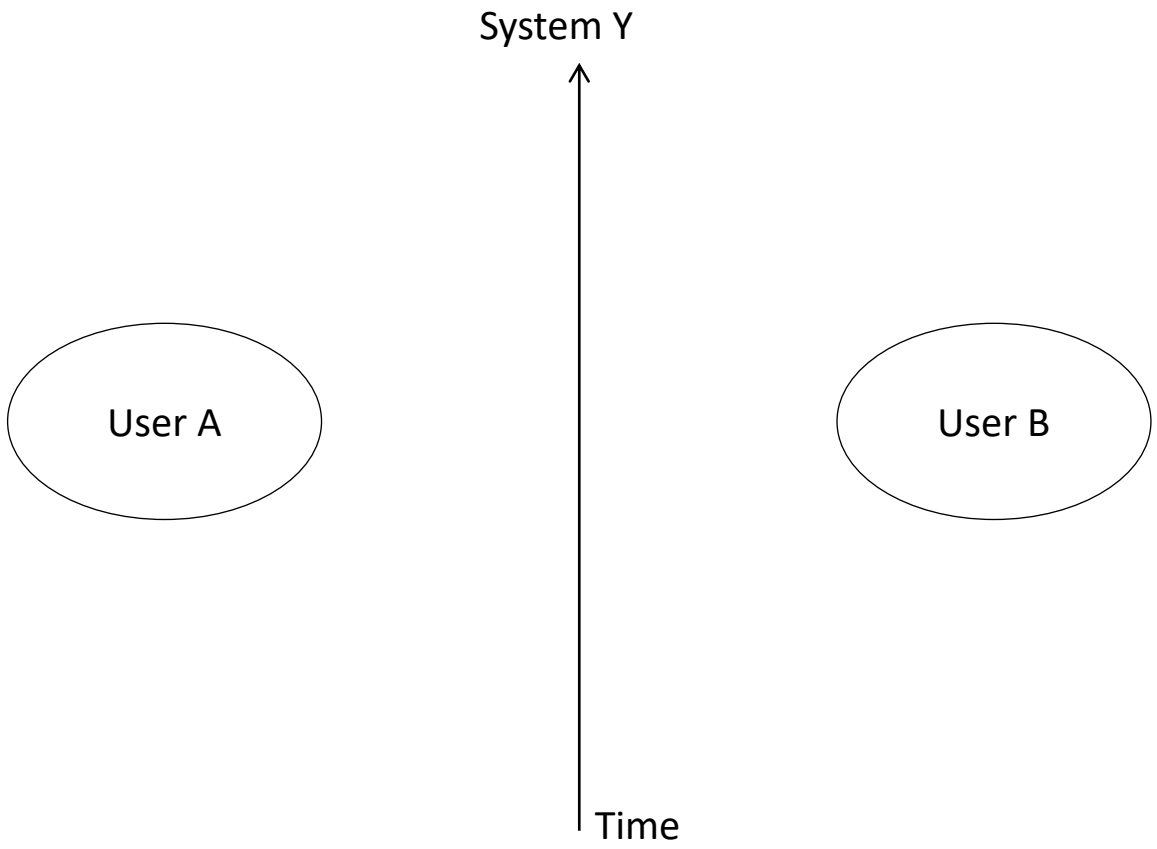
Week 9

Claim: System X' is Deducibility Secure



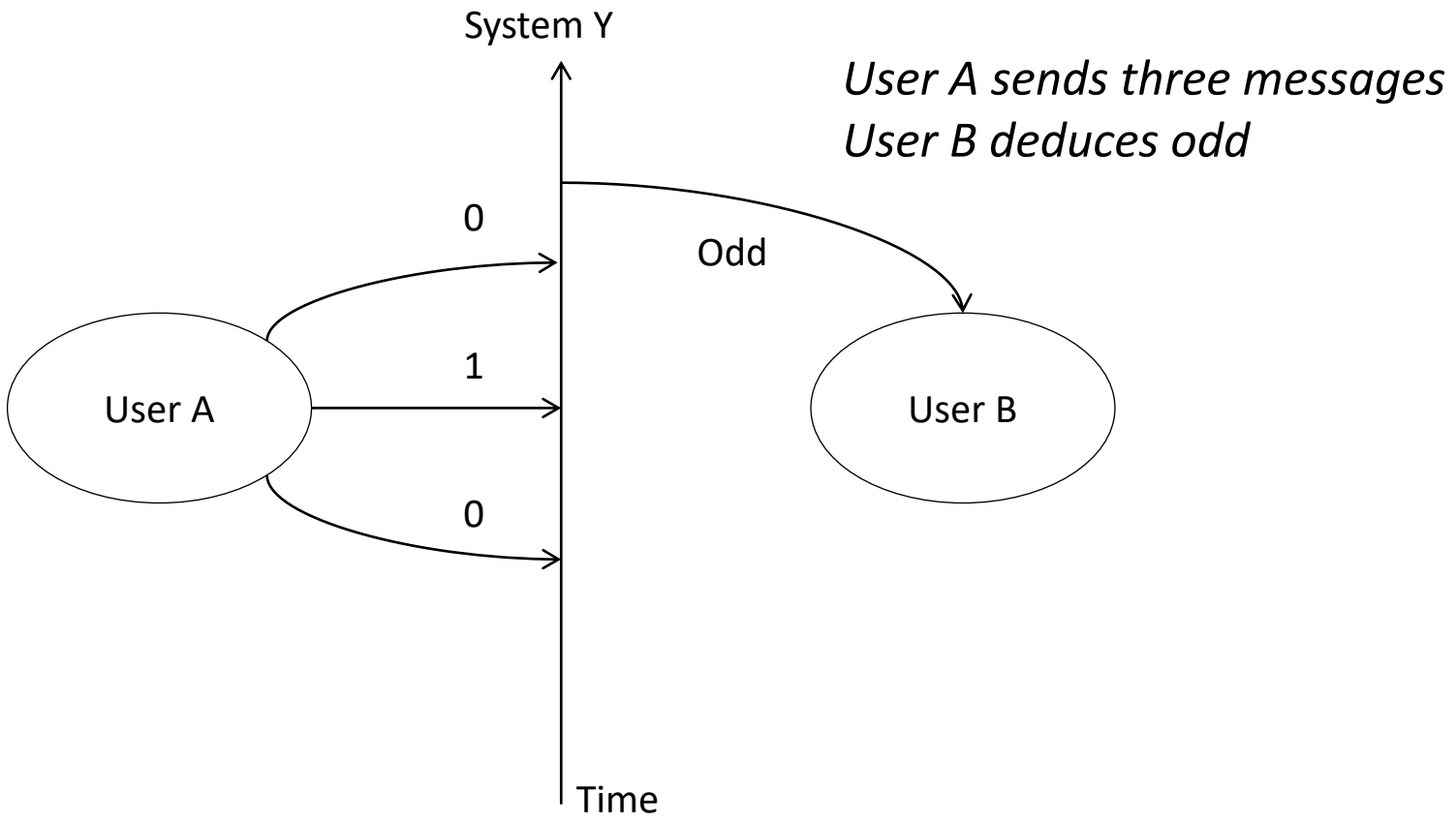
Week 9

New System Y



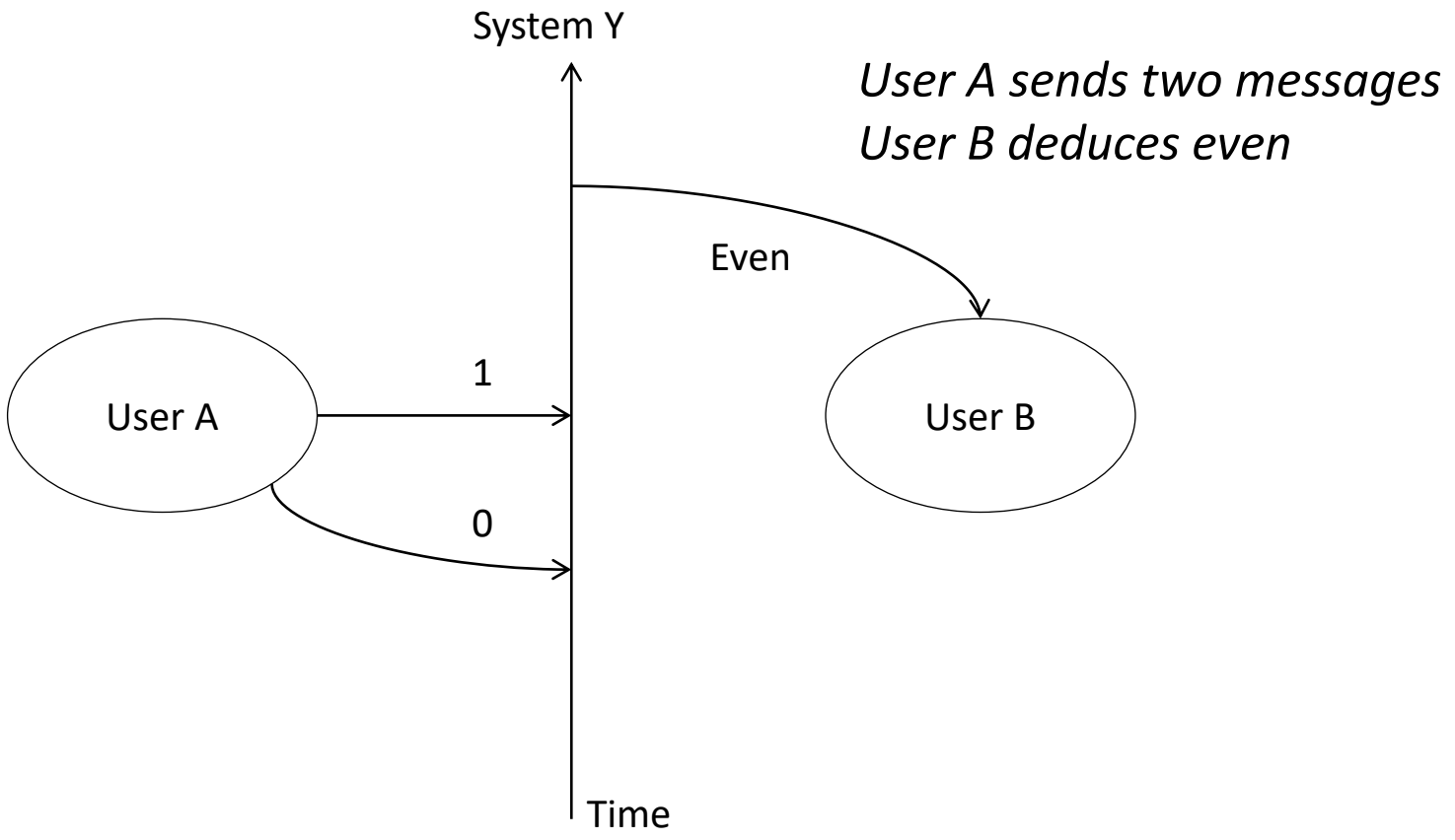
Week 9

User B Reads Parity of User A's Message Count



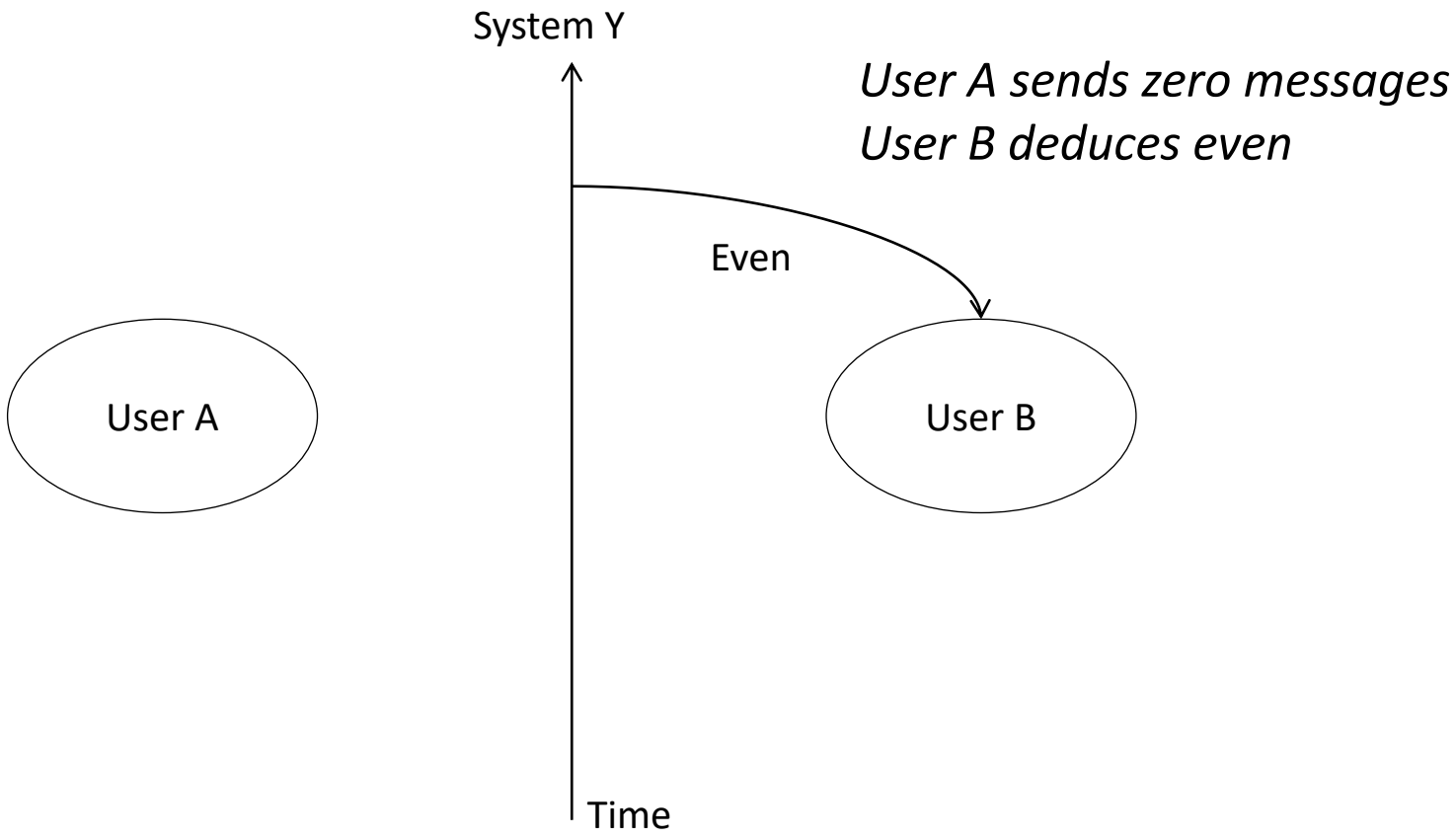
Week 9

User B Reads Parity of User A's Message Count



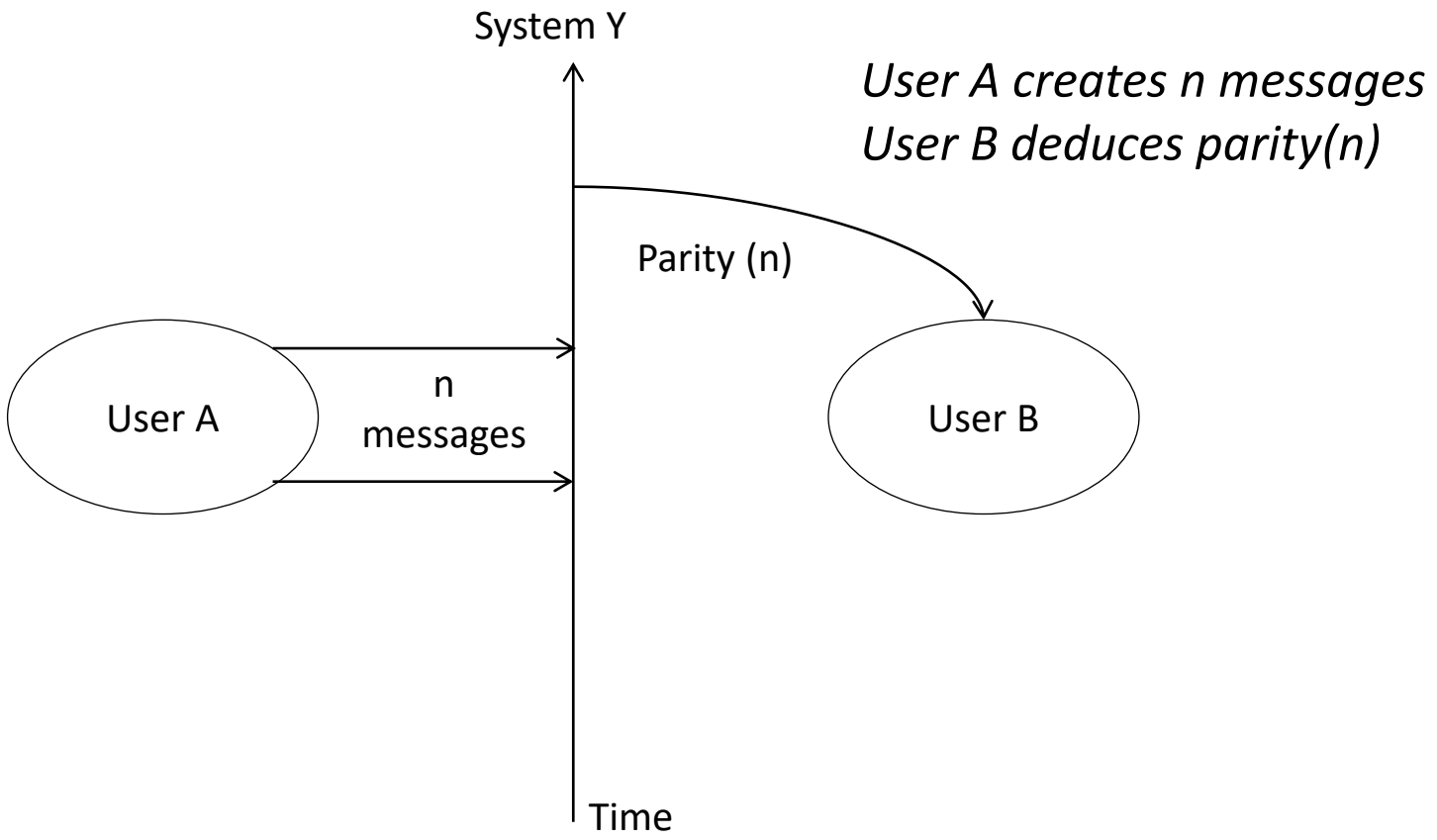
Week 9

User B Reads Parity of User A's Message Count



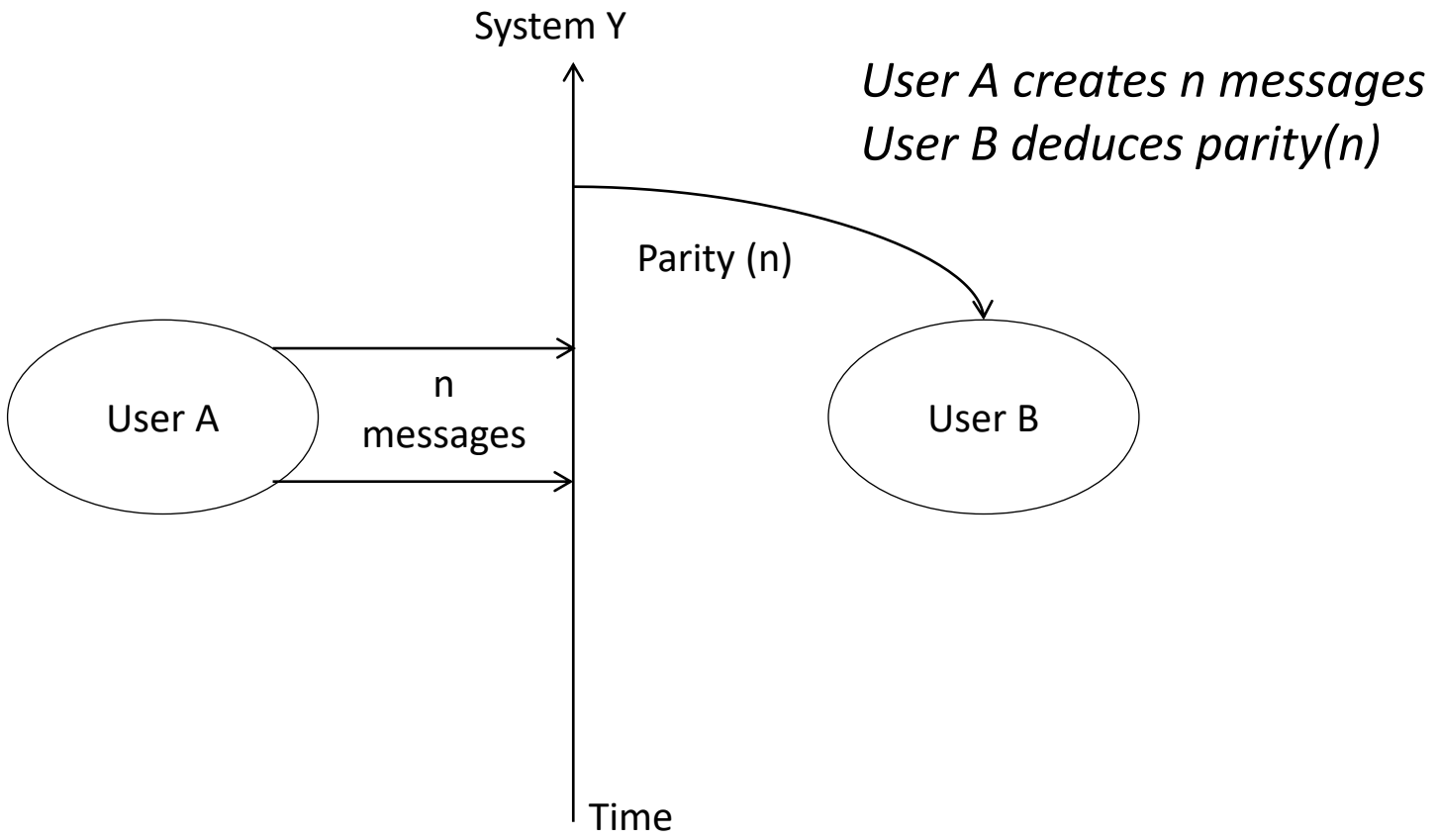
Week 9

System Y: Overt Channel Between Users A and B



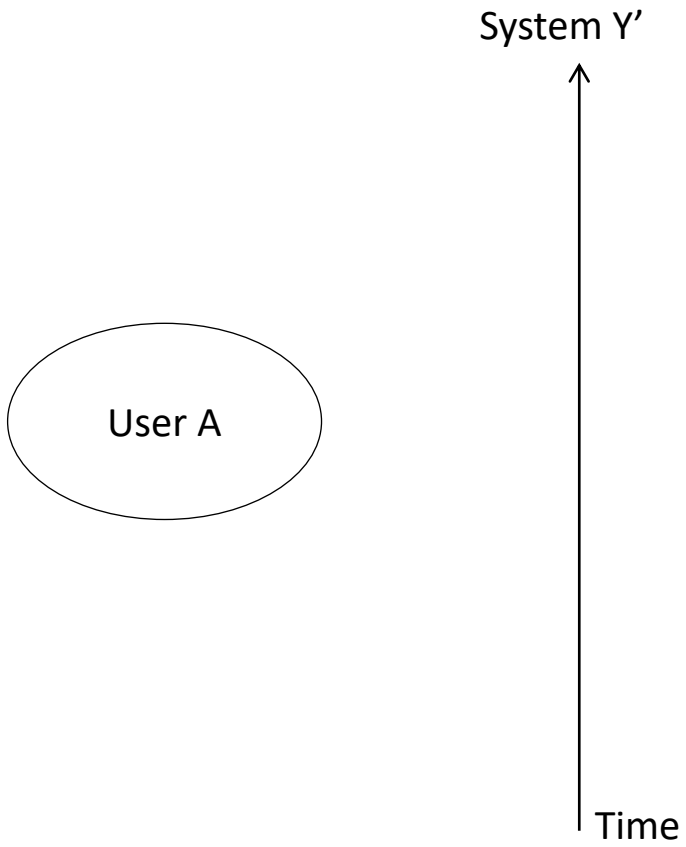
Week 9

Claim: System Y is Not Deducibility Secure



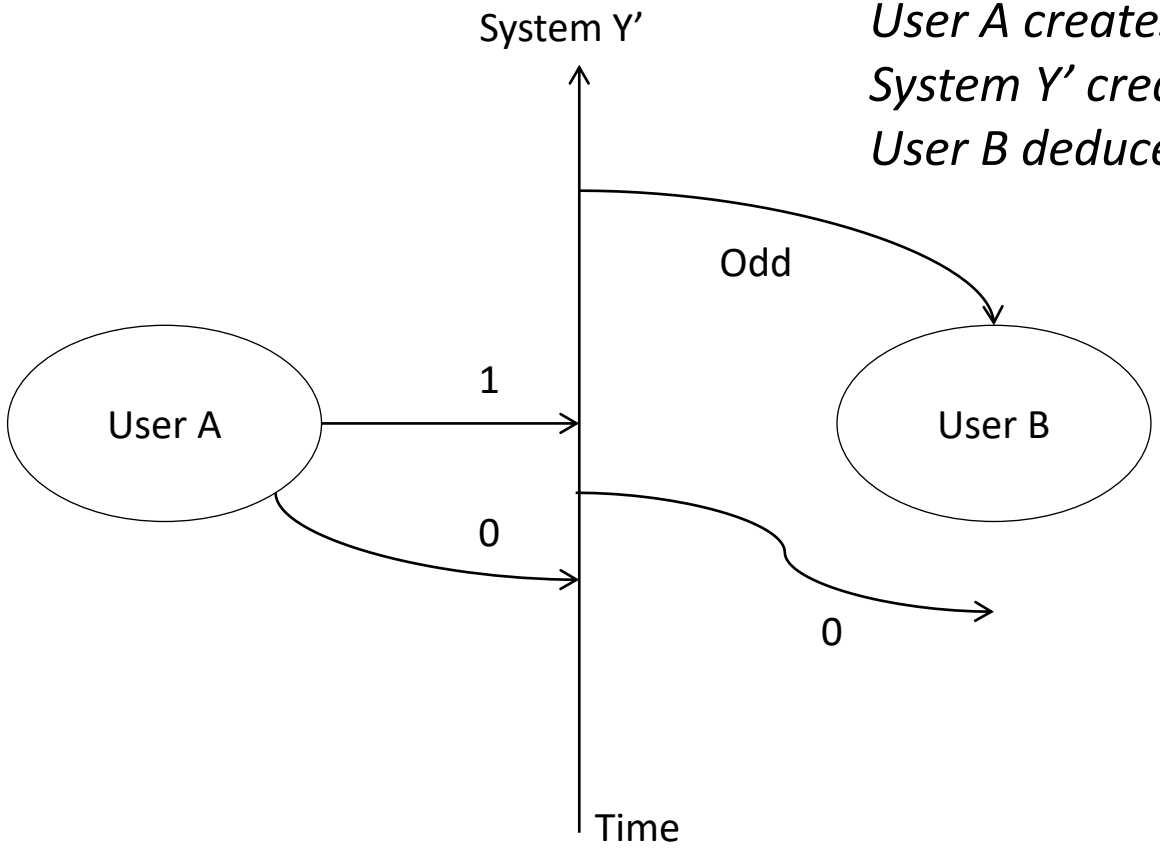
Week 9

New System Y’



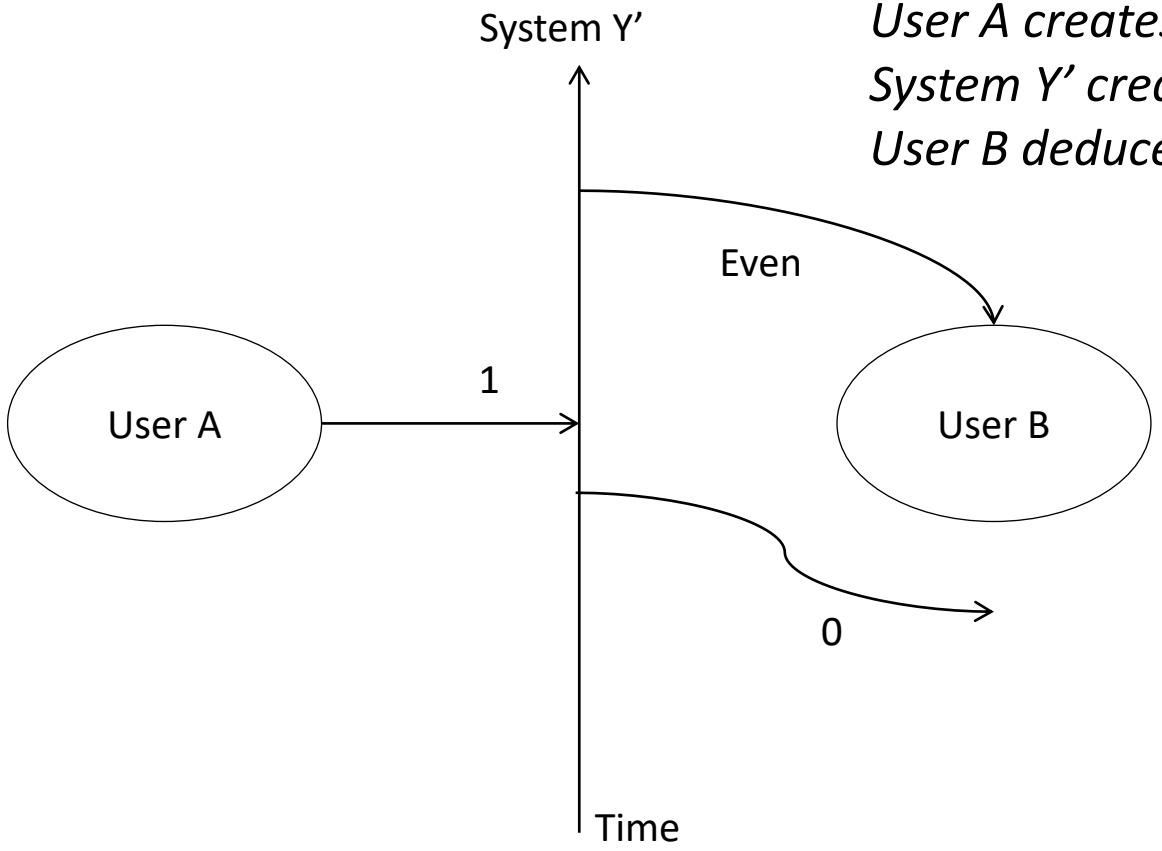
Week 9

User B Reads Parity of User A and System Y' Message Count



Week 9

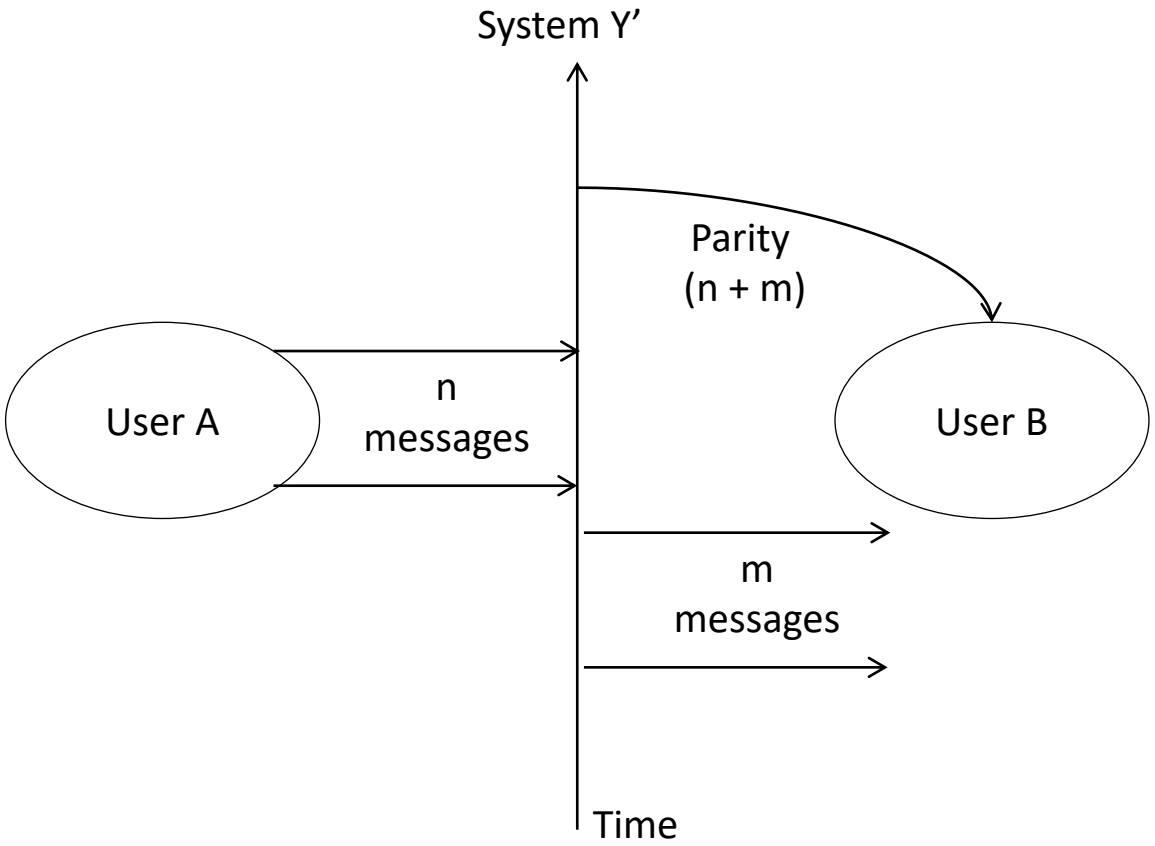
User B Reads Parity of User A and System Y' Message Count



User A creates 1 message
System Y' creates 1 message
User B deduces parity(2) = even

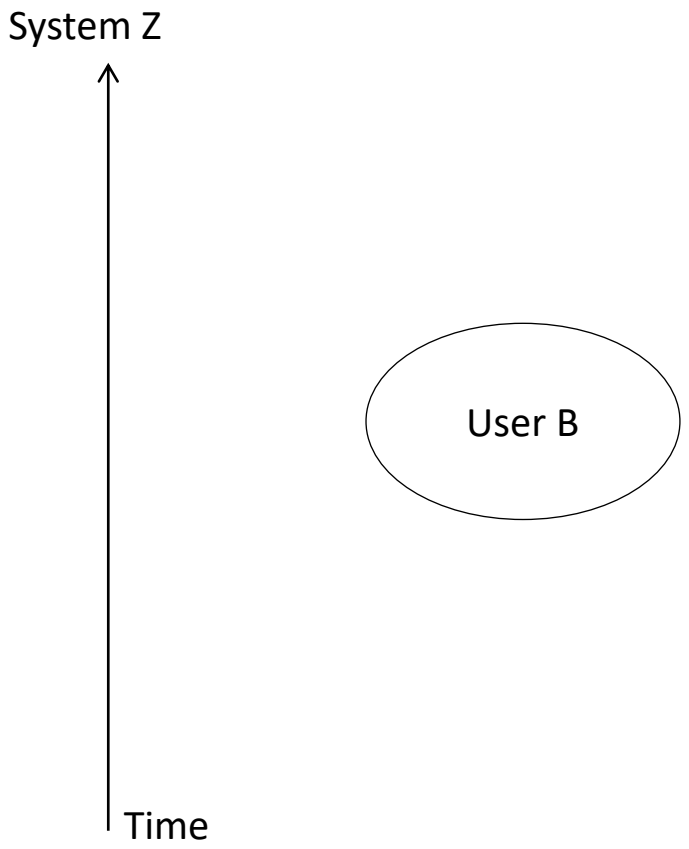
Week 9

Claim: System Y' is Deducibility Secure



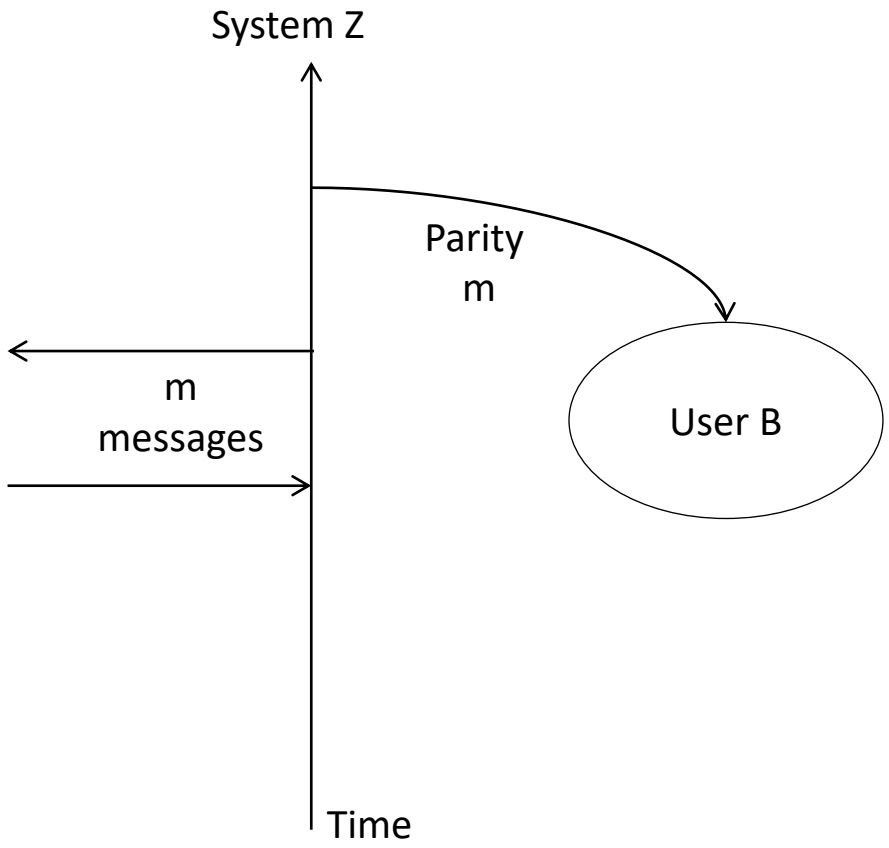
Week 9

New System Z



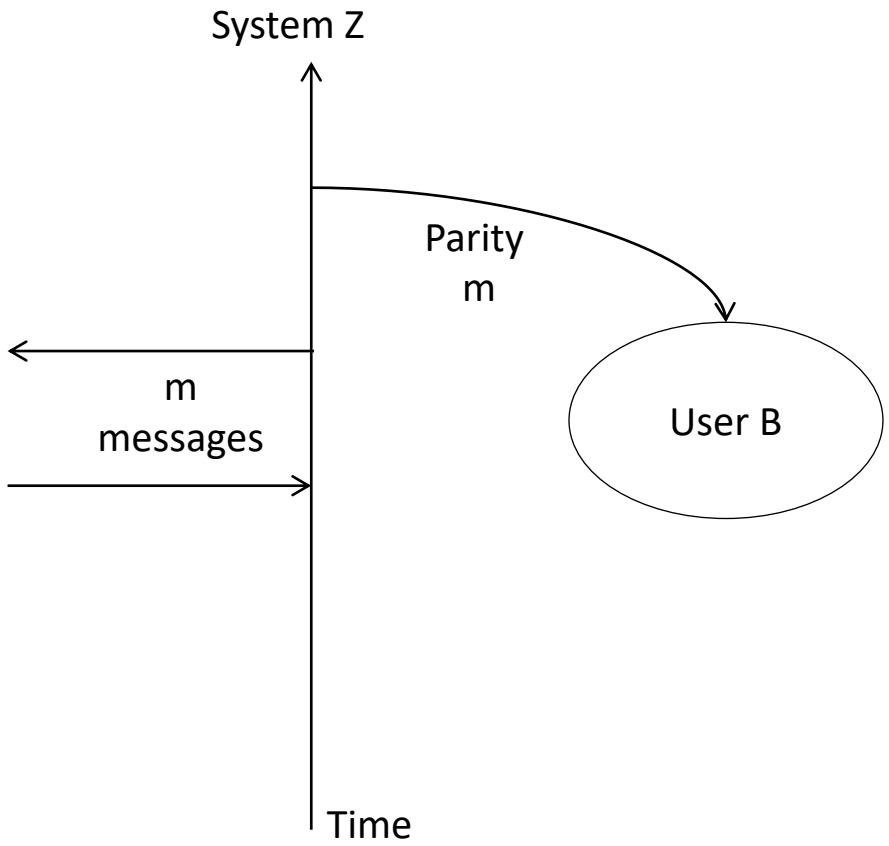
Week 9

User B Reads Parity of Message Count Between System Z and User A



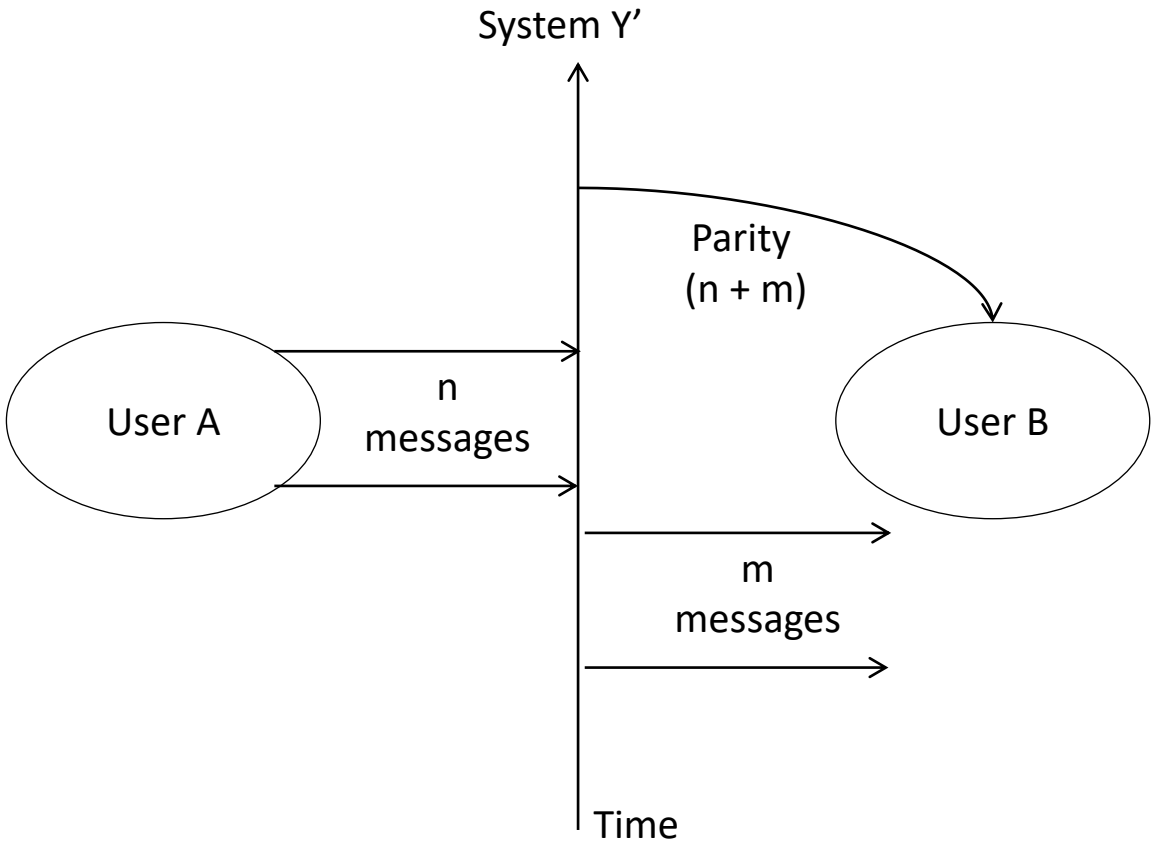
Week 9

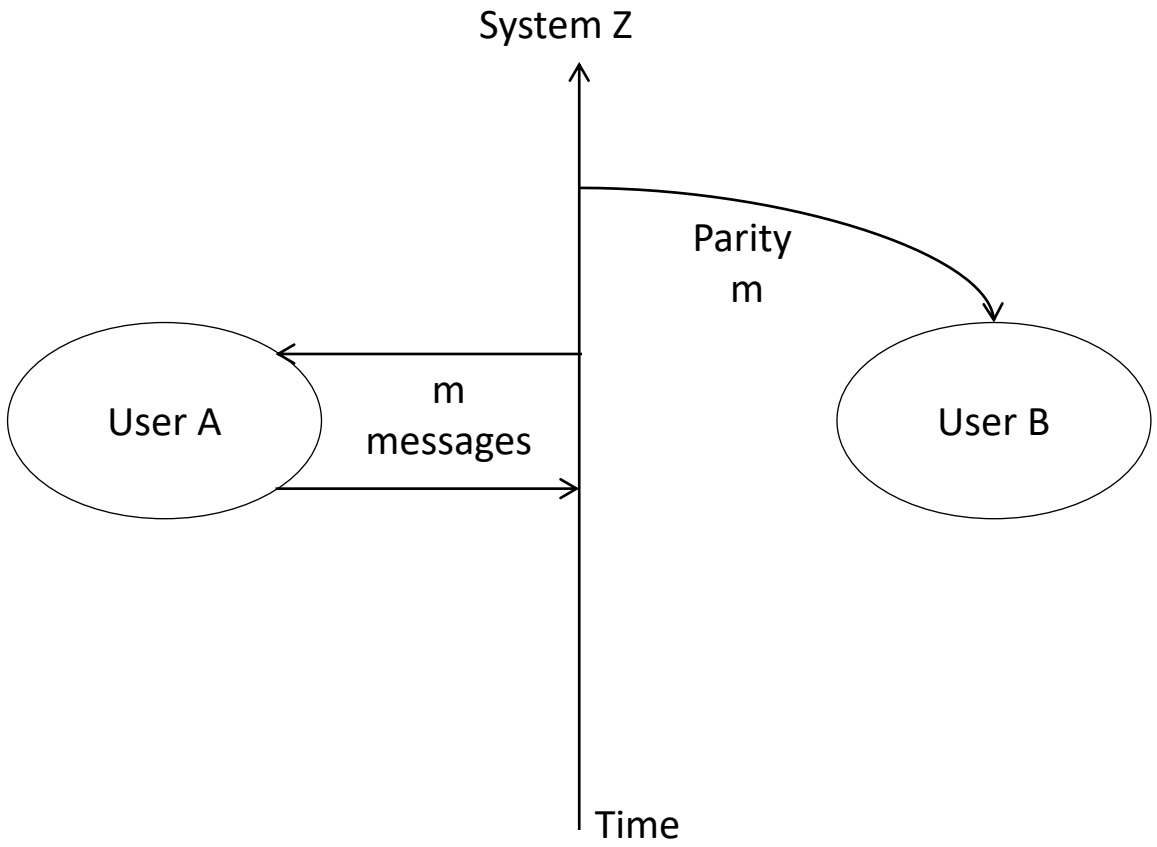
Claim: System Z is Deducibility Secure (with respect to B)



Week 9

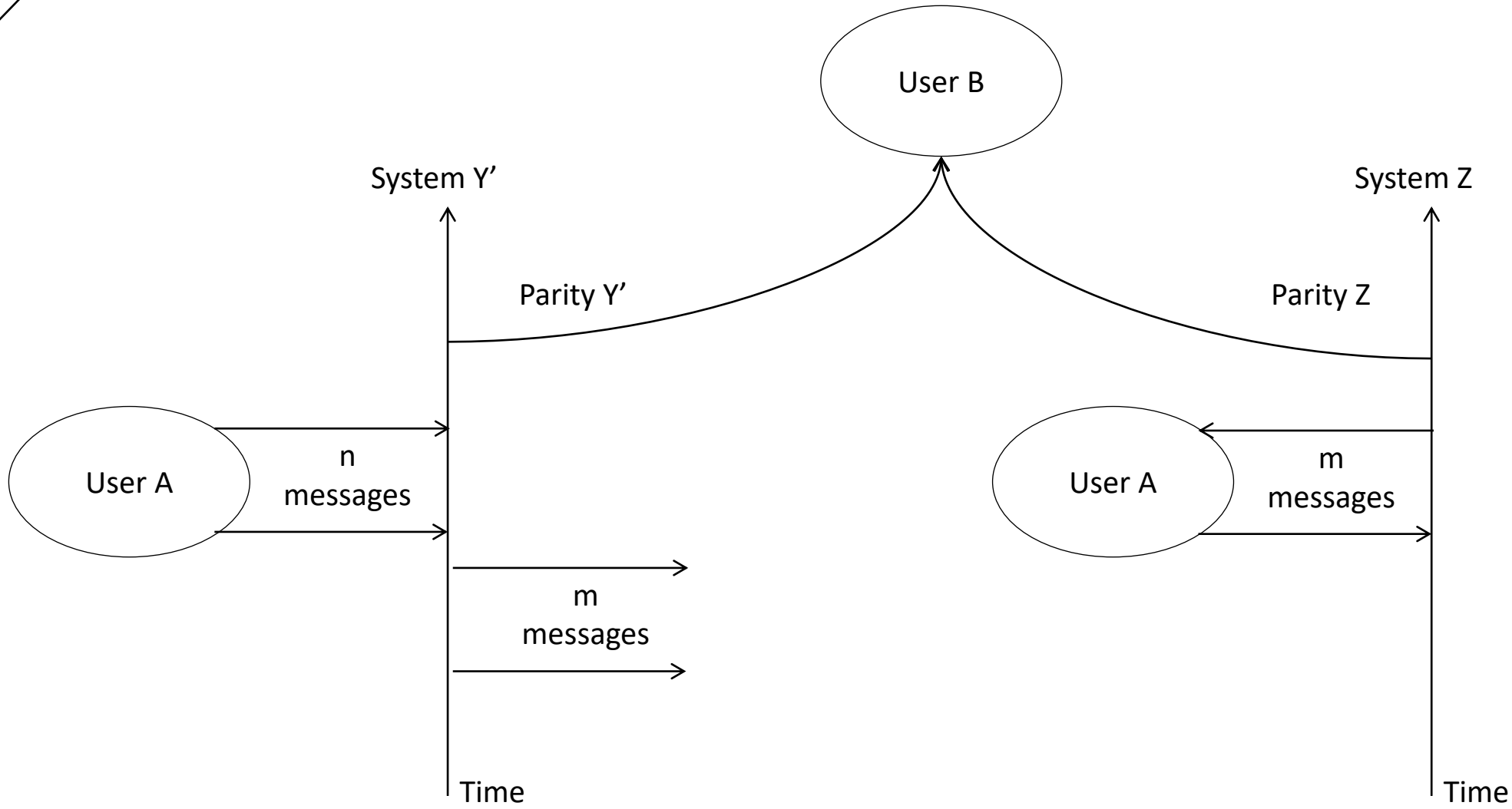
Two Separate Deducibility Secure Systems





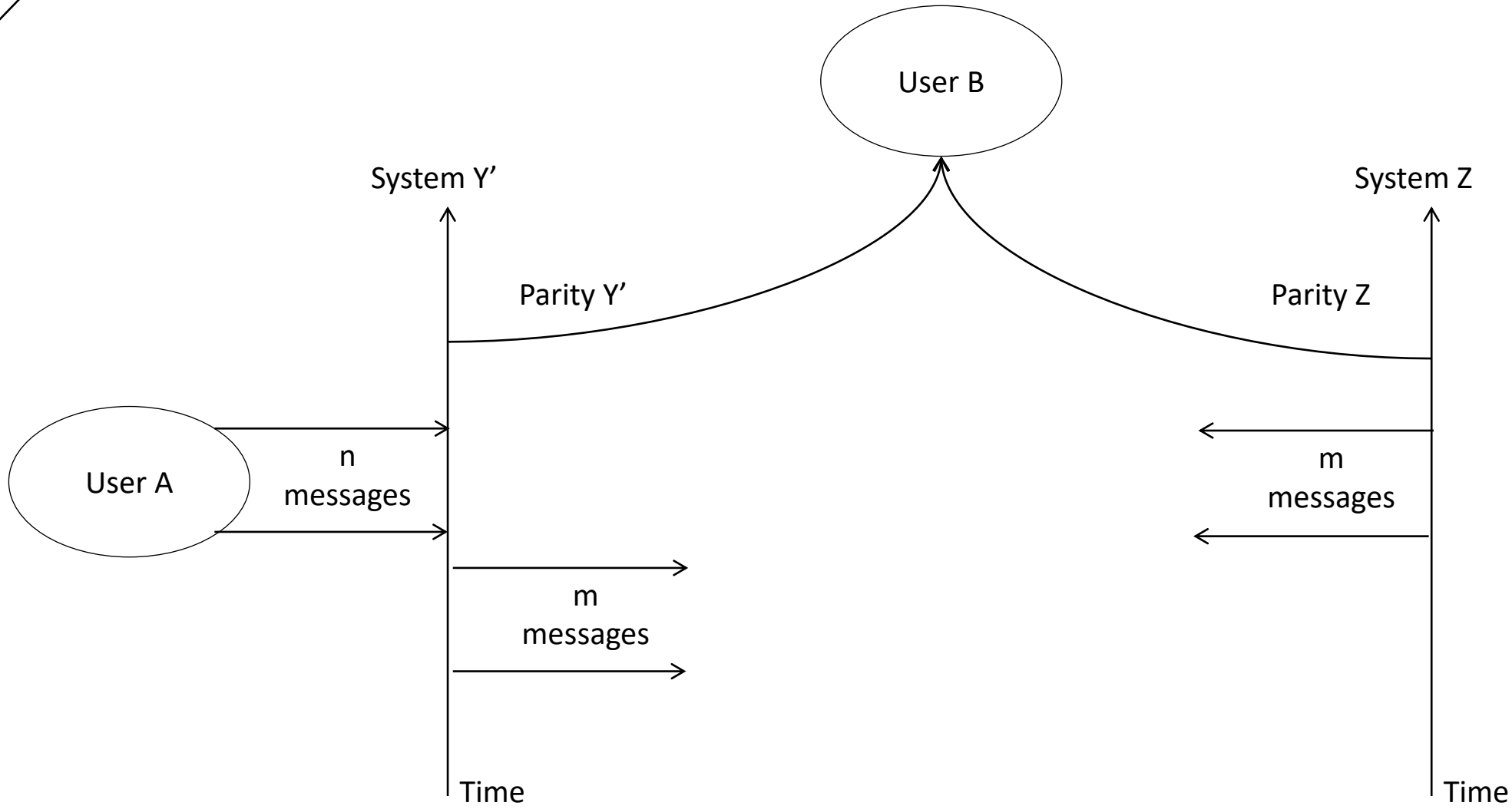
Week 9

User B Reads Both Parities



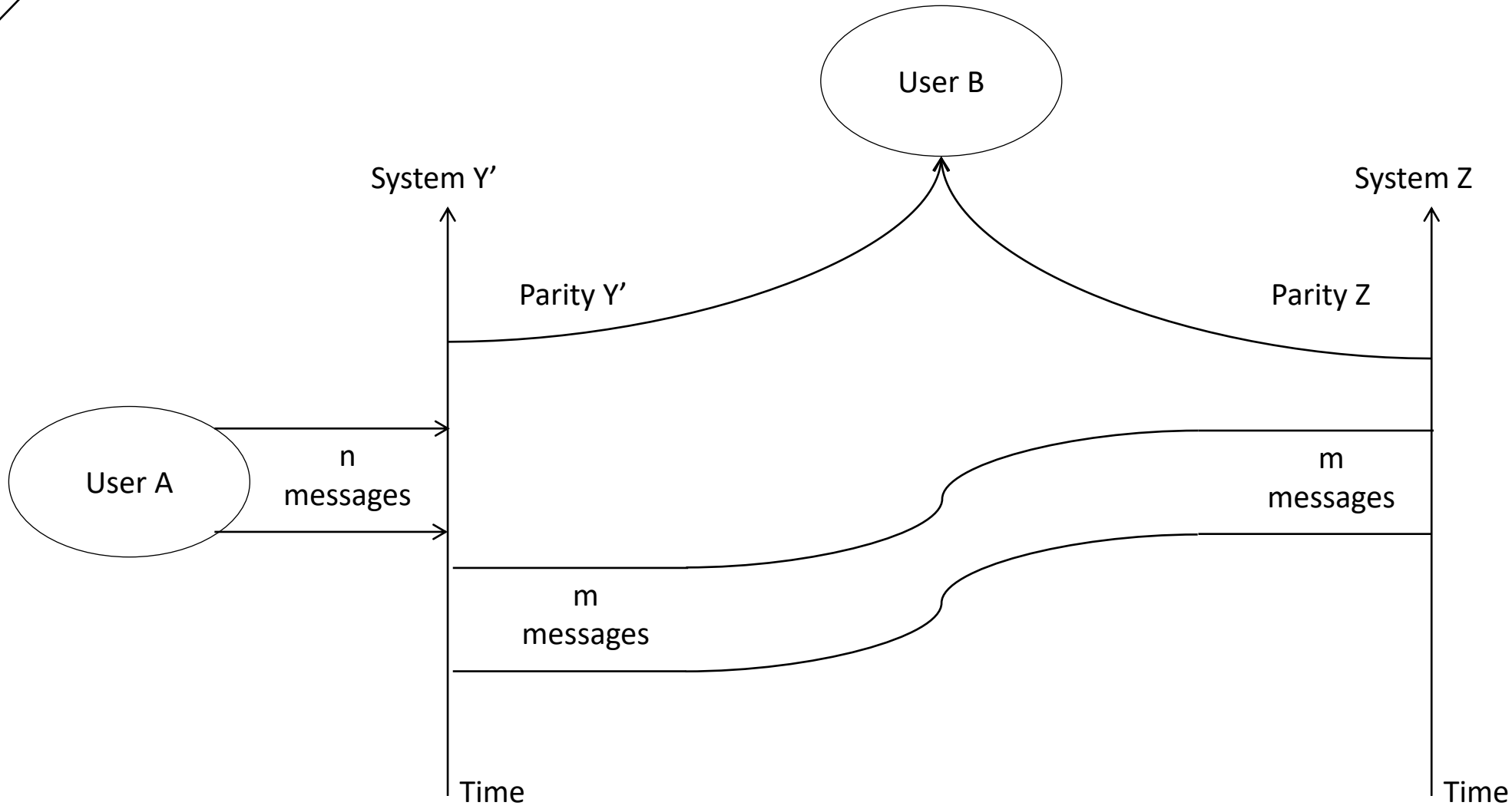
Week 9

User B Reads Both Parities



Week 9

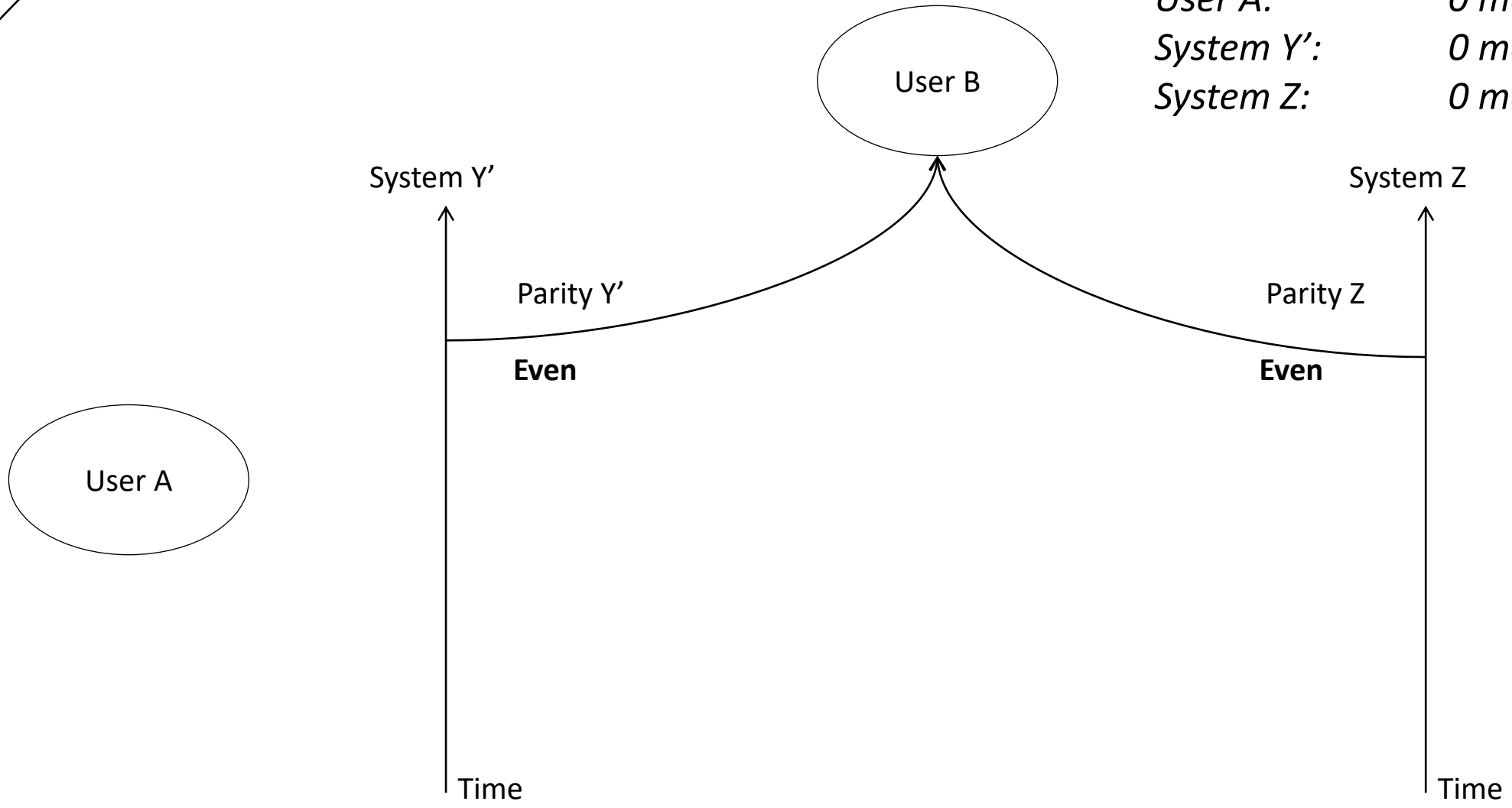
Hook Up Messages in Obvious Way – System (Y', Z)



Week 9

User A Sends Zero Messages: Parities Equal

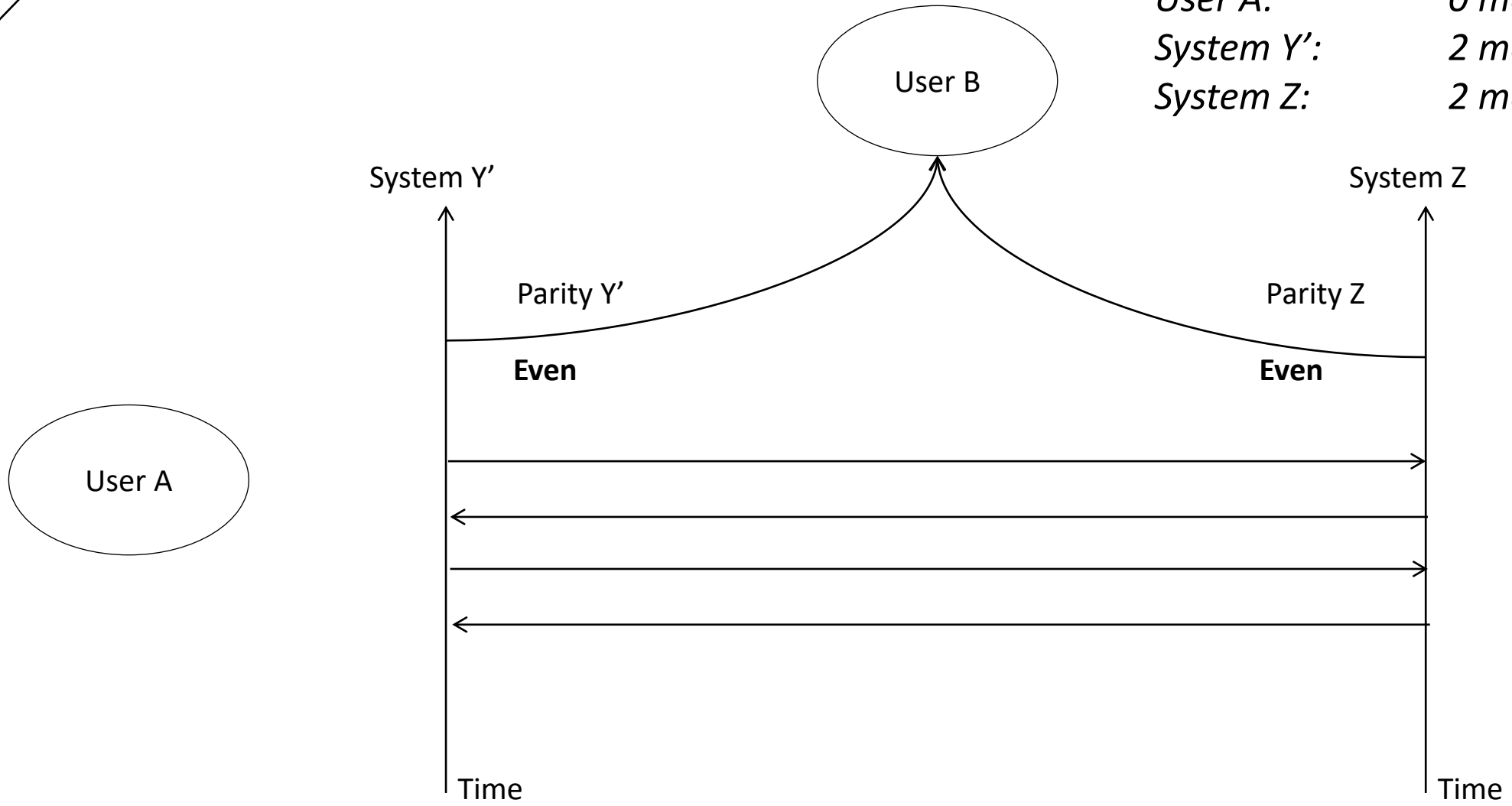
User A: *0 messages*
System Y': *0 messages*
System Z: *0 messages*



Week 9

User A Sends Even Number Messages: Parities Equal

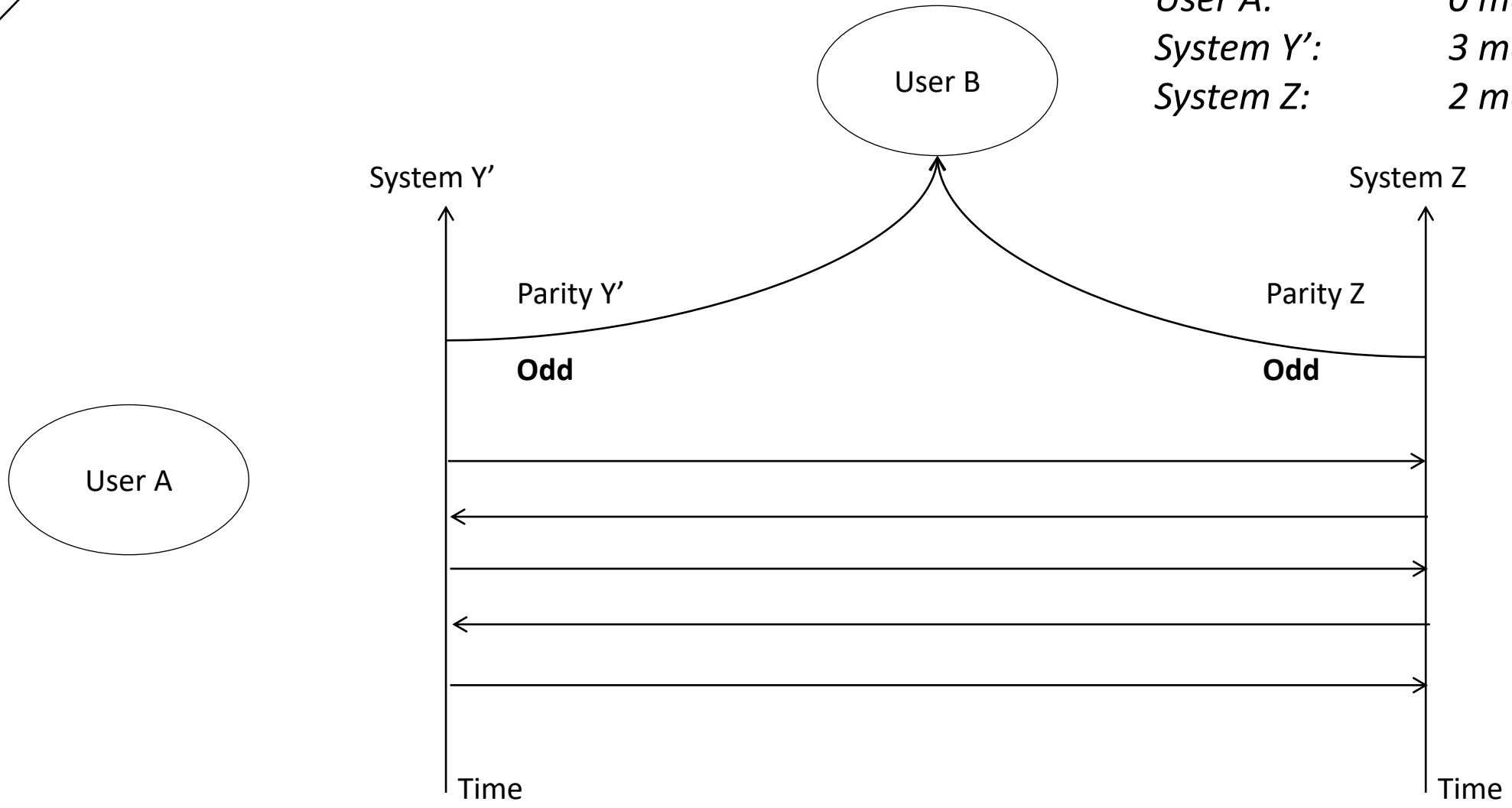
User A: *0 messages*
System Y': *2 messages*
System Z: *2 messages*



Week 9

User A Sends Zero Messages: Parities Equal

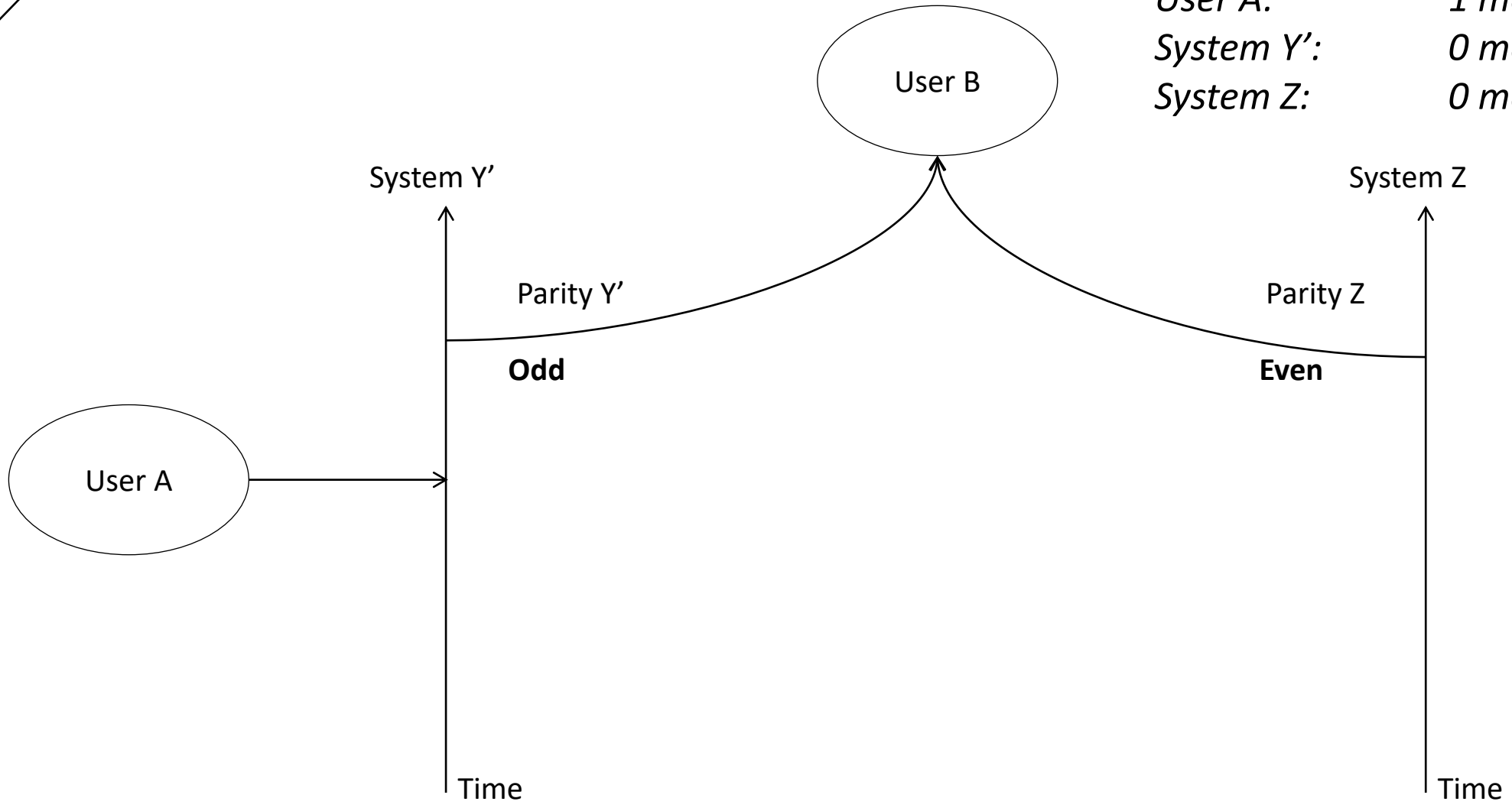
User A: *0 messages*
System Y': *3 messages*
System Z: *2 messages*



Week 9

User A Sends One Message: Parities Different

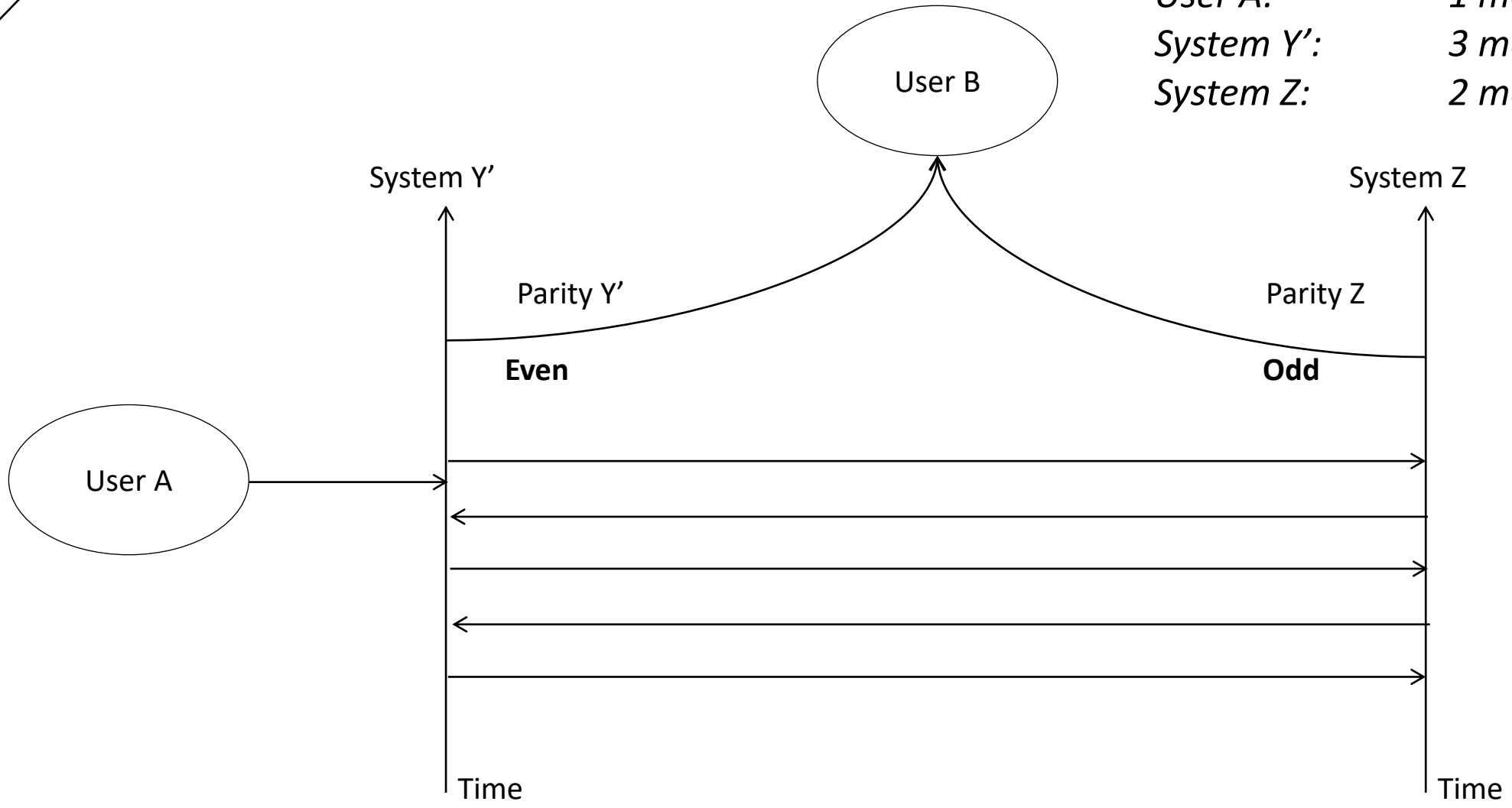
User A: *1 message*
System Y': *0 messages*
System Z: *0 messages*



Week 9

User A Sends One Message: Parities Different

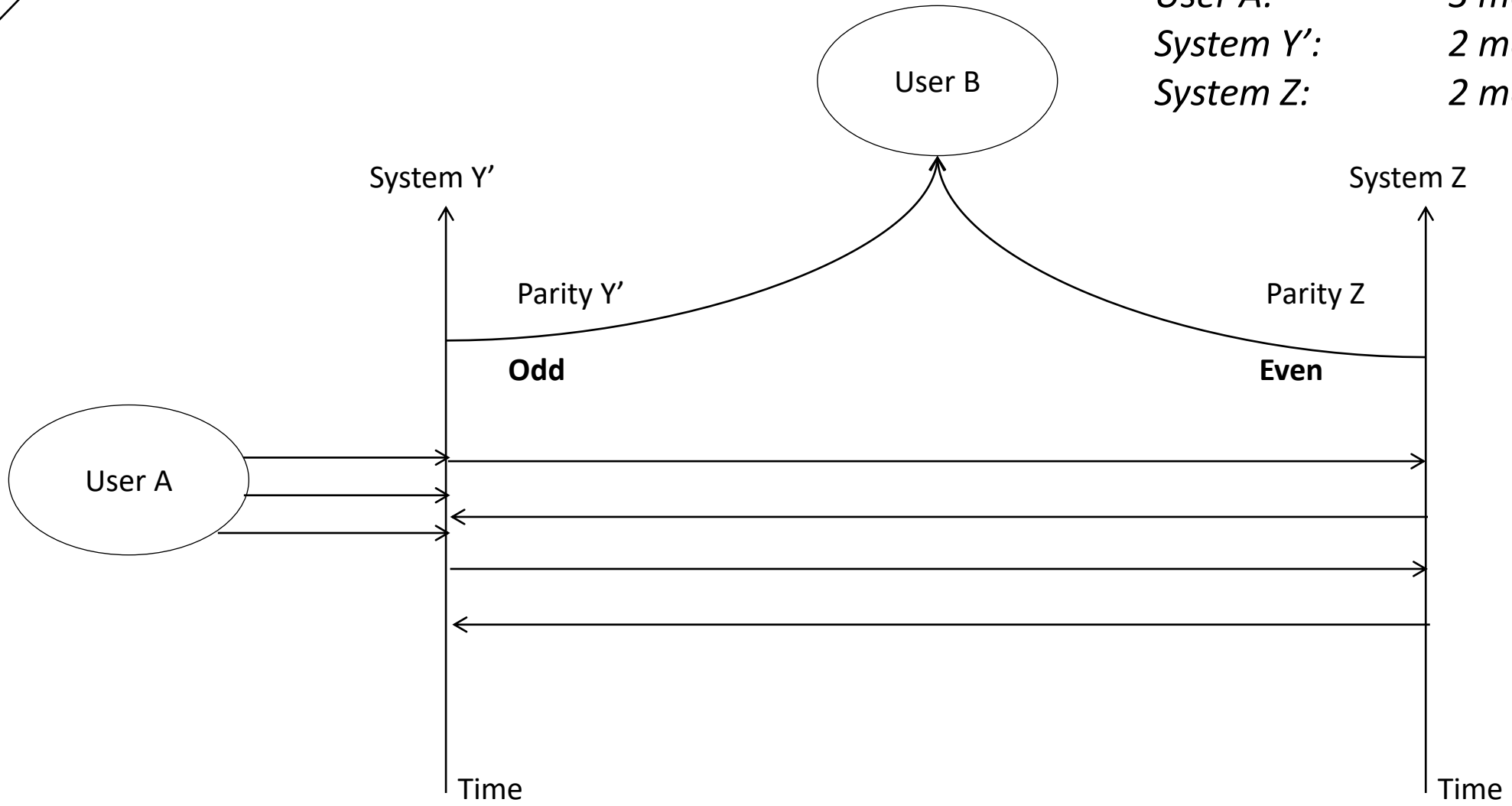
User A: *1 message*
System Y': *3 messages*
System Z: *2 messages*



Week 9

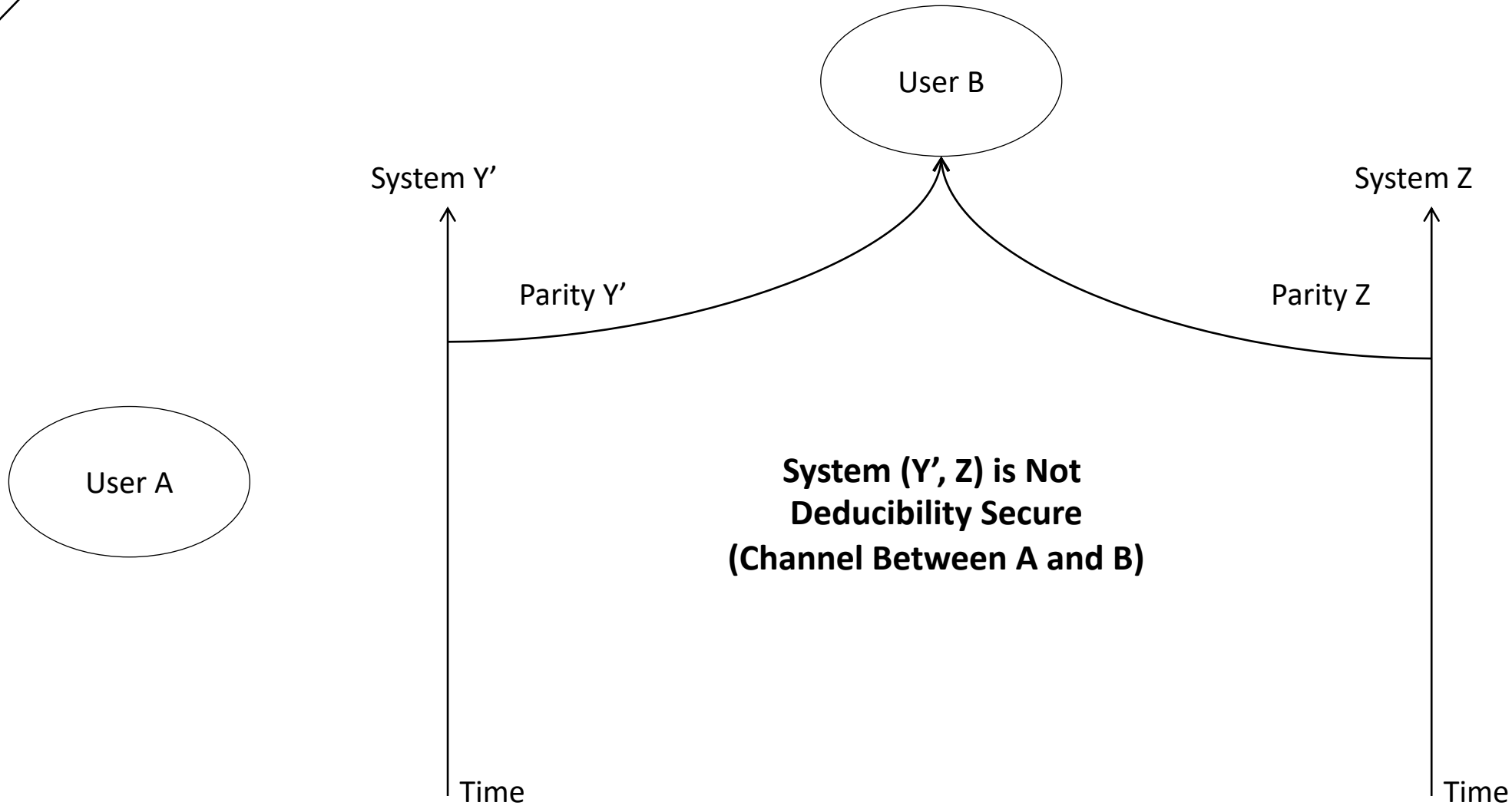
User A Sends Odd Number of Messages: Parities Different

User A: *3 messages*
System Y': *2 messages*
System Z: *2 messages*

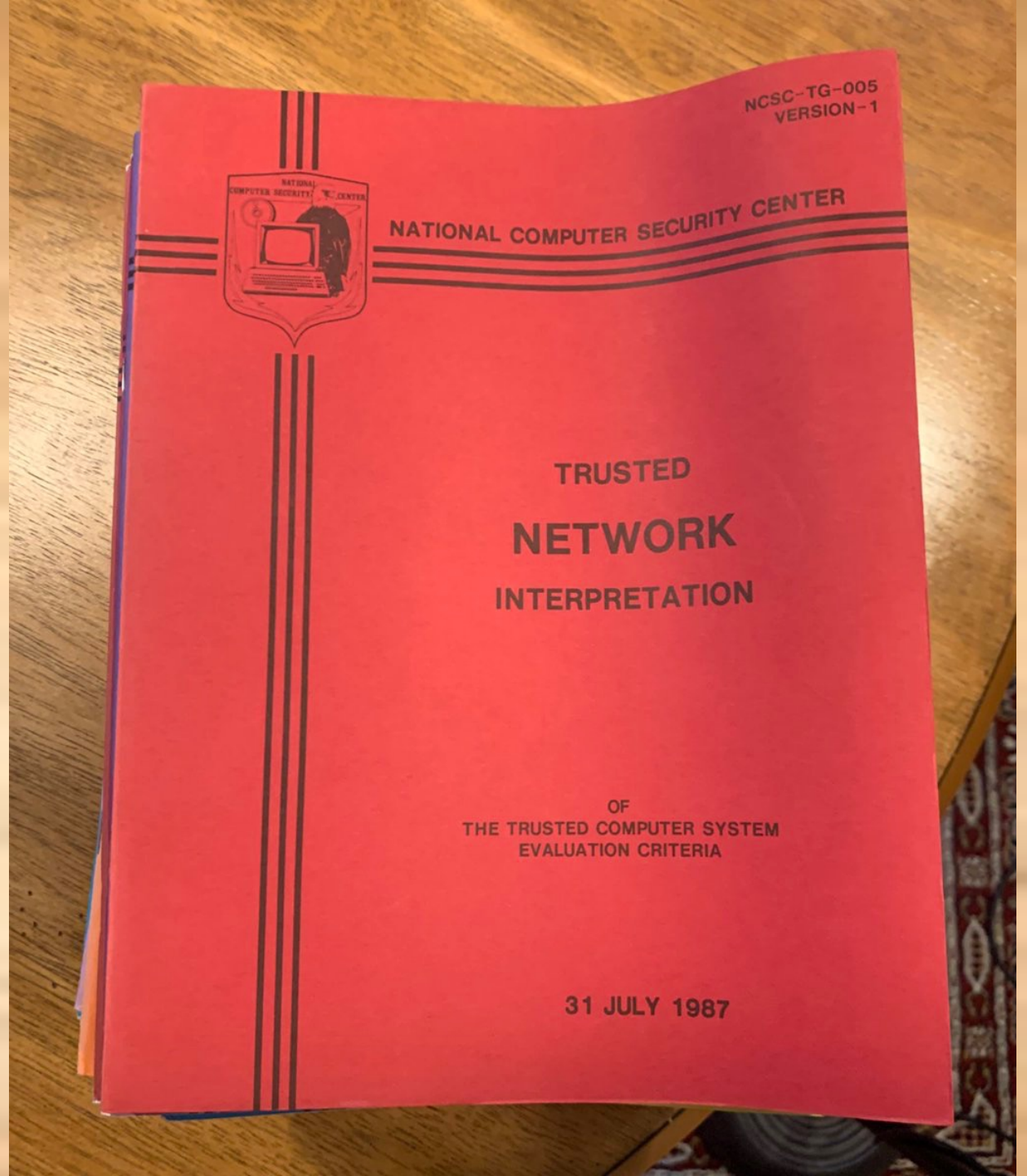


Week 9

Claim: Deducibility Security Does Not Compose



Week 9


NCSC-TG-005
VERSION-1
NATIONAL COMPUTER SECURITY CENTER
TRUSTED
NETWORK
INTERPRETATION
OF
THE TRUSTED COMPUTER SYSTEM
EVALUATION CRITERIA
31 JULY 1987