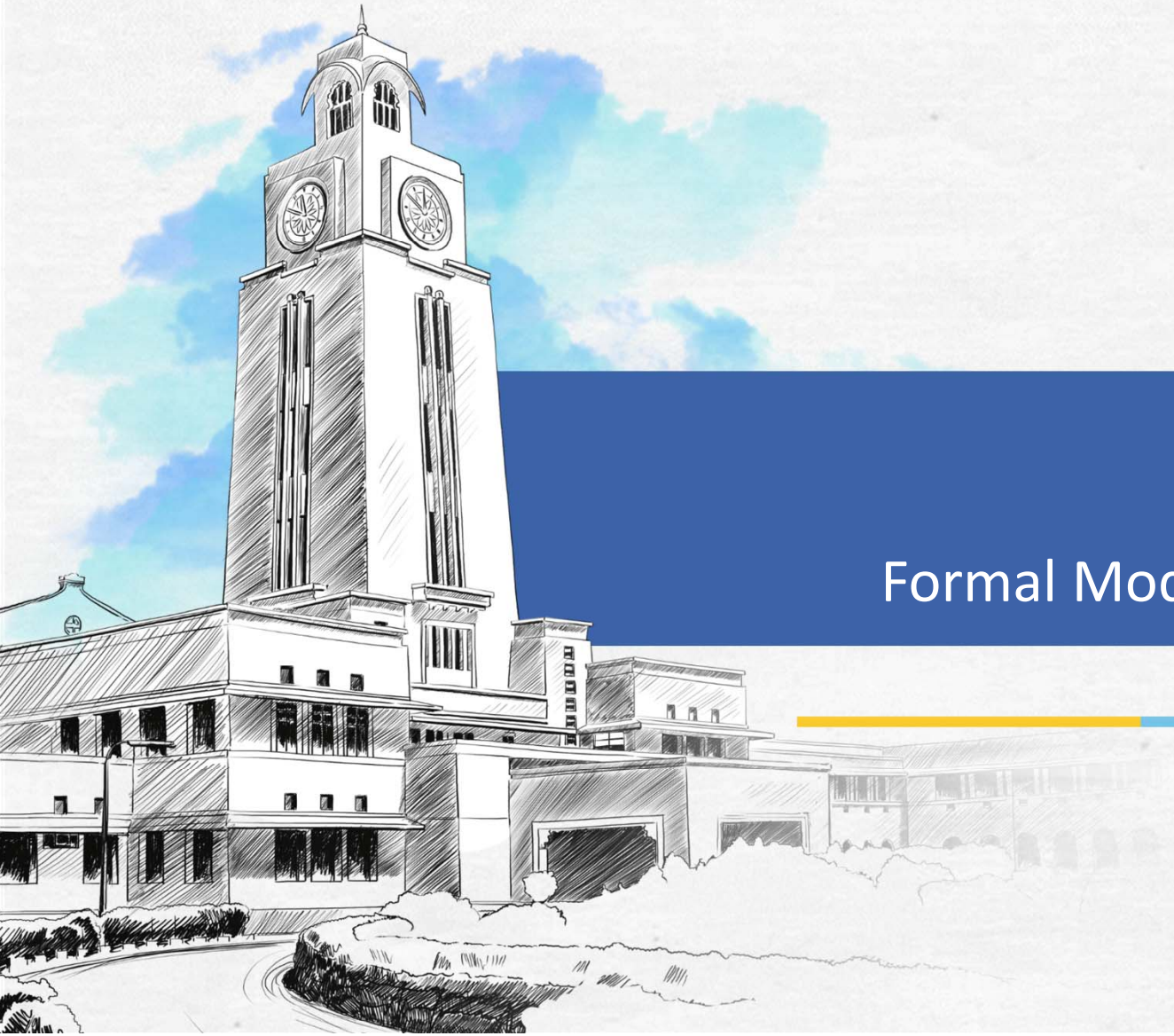

BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Cyber Security

Formal Models of Computer Security

Dr. Ramakrishna Dantu
Associate Professor, BITS Pilani

Disclaimer and Acknowledgement

- The content for these slides has been obtained from books and various other source on the Internet
- I here by acknowledge all the contributors for their material and inputs.
- I have provided source information wherever necessary
- I have added and modified the content to suit the requirements of the course

Formal Models of Computer Security

Agenda

- The CIA Classification:
  - Confidentiality Policies:
    - Bell-LaPadula Model
  - Integrity Policies:
    - The Biba Model
    - Lipner's Integrity Matrix Model
    - Clark-Wilson Integrity Model
    - Trust Models
  - Availability Policies:
    - Deadlock
    - Denial of Service Models

Lipner's Integrity Matrix

Integrity Policies - Recap

Commercial Integrity Constraints

1) Users will not write their own programs, but will use existing production programs and databases.
2) Programmers will develop and test programs on a non-production system
3) If they need access to actual data, they will be given production data via a special process, but will use it on their development system.
4) A special process must be followed to install a program from the development system onto the production system.
5) The special process in requirement 4 must be controlled and audited.
6) The managers and auditors must have access to both the system state and the system logs that are generated.

Lipner's Integrity Matrix

Overview

- Lipner devised his Integrity Matrix Model specifically to handle those concerns/constraints in a commercial environment
- Liper's model combines the elements of Bell La-Padula and Biba models to provide confidentiality and integrity
- Does it in two steps
  - Bell-LaPadula component first (Confidentiality)
  - Add in Biba component (Integrity)

Lipner's Integrity Matrix

Lipner's Use of Bell-LaPaluda Model

- Confidentiality levels (higher to lower):
  - Audit Manager (AM):
    - system audit and management functions are at this level.
  - System Low (SL):
    - any process can read information at this level
- Five confidentiality categories:
  - Development (D):
    - programs under development and testing, but not yet in production use
  - Production Code (PC):
    - production processes and programs
  - Production Data (PD):
    - data covered by the integrity policy
  - System Development (SD):
    - system programs under development, but not yet in production use
  - Software Tools (T):
    - programs provided on the production system not related to the sensitive or protected data

Lipner's Integrity Matrix

Users and Security Levels

- Lipner's assignment of users to security levels based on their jobs
  - Ordinary users
    - can execute (read) production code but cannot alter it
    - can alter and read production data
    - cannot execute category T (Software Tools), so they cannot write their own programs
    - hence, their clearance is (SL, {PC, PD})
  - Application Developers
    - need access to tools for developing their programs
    - do not have read/write access to PD (Production Data), so cannot access production data
    - If they need production data, the data must first be downgraded to D (this requires sys admins)
    - hence, application programmers have (SL, {D, T}) clearance

Lipner's Integrity Matrix

Users and Security Levels

- Lipner's assignment of users to security levels based on their jobs
  - System Programmers
    - System programmers develop system programs and, like application programmers, use tools to do so
    - hence, system programmers should have clearance (SL, {SD, T})
  - System managers and Auditors
    - need access to all logs but cannot change levels of objects
    - their clearance is (AM, {D, PC, PD, SD, T})
  - System controllers
    - need to install code
    - must have the ability to downgrade code once it is certified for production, so other entities cannot write to it
    - their clearance is (SL, {D, PC, PD, SD, T}) with the ability to downgrade programs

Lipner's Integrity Matrix

Users and Security Levels

Subjects	Description	Security Level
Ordinary users	Will use production code to modify production data	(SL, { PC, PD })
Application developers	Develop programs and need access to tools for developing their programs	(SL, { D, T })
System programmers	Develop system programs and, use tools to do so	(SL, { SD, T })
System managers and auditors	Need high clearance to be able to access all logs	(AM, { D, PC, PD, SD, T })
System controllers	Must have the ability to downgrade code once it is certified for production, so other entities cannot write to it	(SL, {D, PC, PD, SD, T}) and downgrade privilege

- E.g.,: Ordinary users have security level of System Low (SL) under the categories of Production Code and Production Data
- E.g.,: System Programmers have security level of System Low (SL) under the categories of System Development and Software Tools

Lipner's Integrity Matrix

Users and Security Levels

Security Level → Categories↓	**Audit Manager (AM)**	**System Low (SL)**
Development (D)	System managers and auditors	Application Developers; System Controller
Production Code (PC)	System managers and auditors	Ordinary Users; System Controller
Production Data (PD)	System managers and auditors	Ordinary Users; System Controller
System Development (SD)	System managers and auditors	System Programmers; System Controller
Software Tools (T)	System managers and auditors	Application Developers; System Programmers; System Controller

Lipner's Integrity Matrix

Objects and Classifications

- Objects are assigned to security levels/categories based on who should access them
- Objects that might be altered have two categories:
  - that of the data itself and that of the program that may alter it
- For example:
  - Ordinary user needs to execute (read) production code,
    - so this is labeled (SL, {PC})
    - This is based on simple security policy of the Bell-LaPadula Model
  - Ordinary users should be able to write production data,
    - so this is labeled (SL, {PC, PD})
    - This is based on *-property of the Bell-LaPadula Model

Objects	Security Level
Development code/test data	(SL, { D, T })
Production code	(SL, { PC })
Production data	(SL, { PC, PD })
Software tools	(SL, { T })
System programs	(SL, ∅)
System programs in modification	(SL, { SD, T })
System and application logs	(AM, { *appropriate* })

Bell LaPadula Model

Access Modes

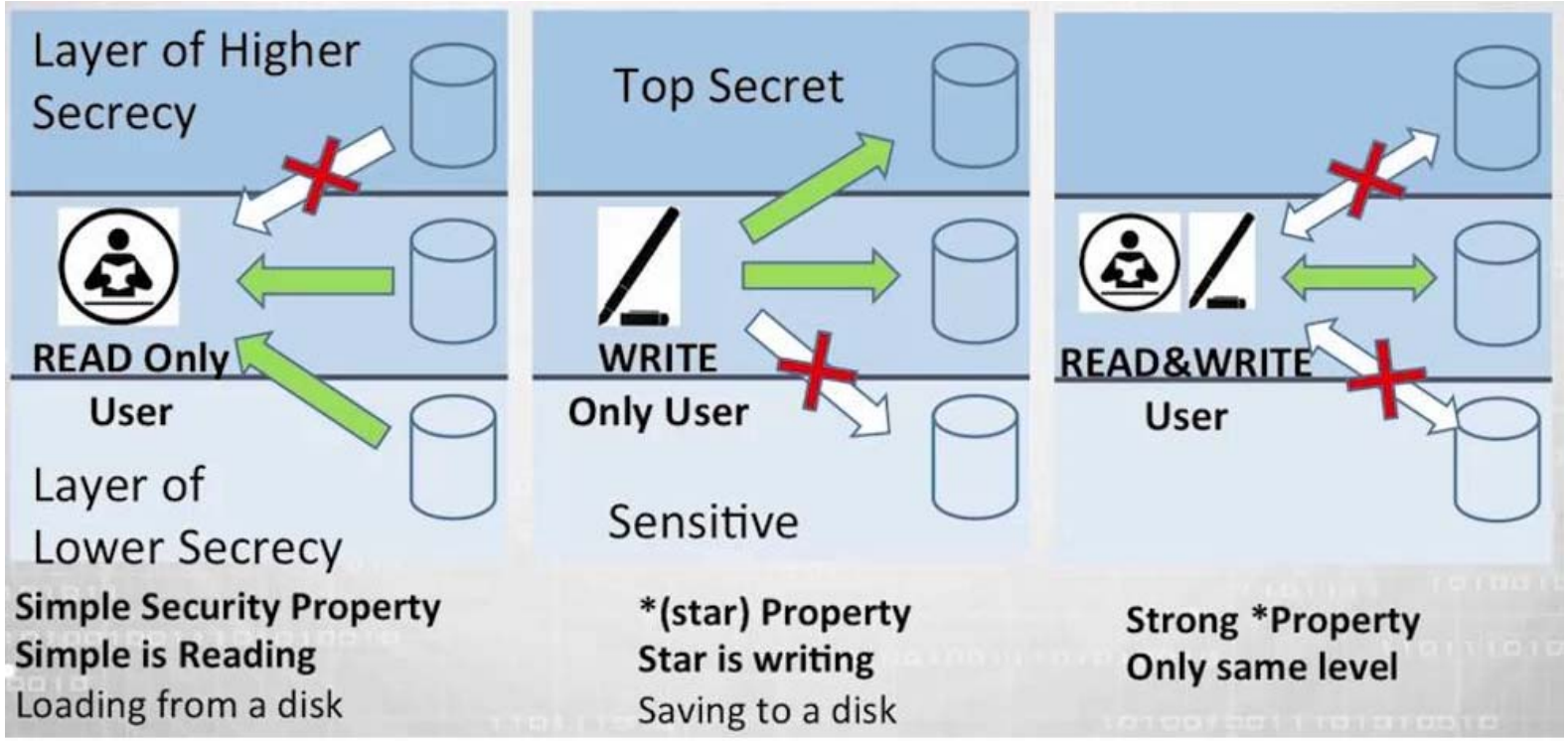

Image Source: Skillset.com

Lipner's Integrity Matrix

Subjects/Objects and Clearance/Classifications

Subjects	Clearance	Objects	Classification
Ordinary users	(SL, { PC, PD })	Development code/test data	(SL, { D, T })
Application developers	(SL, { D, T })	Production code	(SL, { PC })
System programmers	(SL, { SD, T })	Production data	(SL, { PC, PD })
System managers and auditors	(AM, { D, OC, OD, SD, T })	Software tools	(SL, { T })
System controllers	(SL, {D, PC, PD, SD, T}) and downgrade privilege	System programs	(SL, ∅)
		System programs in modification	(SL, { SD, T })
		System and application logs	(AM, { *appropriate* })

Here downgrade means the ability to move software (objects) from development to production

Lipner's Integrity Matrix

Original Requirements – Review

Requirements	How the Requirement is met?
Users will not write their own programs, but will use existing production programs and databases.	Users have no access to T, so cannot write their own programs
Programmers will develop and test programs on a non-production system; if they need access to actual data, they will be given production data via a special process, but will use it on their development system.	Applications programmers have no access to PD, so cannot access production data; if needed, it must be put into D, requiring the system controller to intervene
A special process must be followed to install a program from the development system onto the production system.	Installing a program requires downgrade procedure (from D to PC), so only system controllers can do it
The special process in requirement 3 must be controlled and audited.	Control: only system controllers can downgrade Audit: any such downgrading must be logged
The managers and auditors must have access to both the system state and the system logs that are generated.	System management and audit users are in AM and so have access to system state and logs

Lipner's Integrity Matrix

Problem

- The model is too inflexible in special-purpose software
  - For example, a program for repairing an inconsistent or erroneous production database cannot be application-level software
  - System managers cannot run programs for repairing inconsistent or erroneous production database
    - System managers at AM, production data at SL
- So to remedy these problems, Lipner integrates his model with Biba's model

Lipner's Integrity Matrix

Adding Biba

- Three integrity classifications (highest to lowest)
  - ISP (System Program):
    - for system programs
  - IO (Operational):
    - production programs, development software
  - ISL (System Low):
    - users get this on log in

ISP > IO > ISL

- Two integrity categories
  - ID (Development):
    - development entities
  - IP (Production):
    - production entities

Lipner's Integrity Matrix

Simplify Bell-LaPadula (Confidentiality)

- In the original model, the security category T (tools) allowed
  - application developers and system programmers to use the same programs without being able to alter those programs
- The revised model now distinguishes two integrity categories:
  - Development (ID): development entities
  - Production (IP): production entities
  - They serve the purpose of the security tools (T) category, which is eliminated from the model
- Production code and production data is collapsed into a single category (called SP)

Lipner's Integrity Matrix

Simplify Bell-LaPadula (Confidentiality)

- This gives rise to the following three confidentiality categories:
  - Production (SP):
    - Production code (PC) and data (PD)
  - Development (SD):
    - Same as previous category Development (D)
  - System Development (SSD):
    - Same as previous category System Development (SD)

	Original	New
Subjects		
Development	D	SD
Production Code (PC):	PC	SP
Production Data (PD):	PD	SP
System Development (SD):	SD	SSD
Software Tools (T):	T	Eliminated

Lipner's Integrity Matrix

Security and integrity levels for subjects

Subjects	Security Level	Integrity Level
Ordinary users	(SL, { SP })	(ISL, { IP })
Application developers	(SL, { SD })	(ISL, { ID })
System programmers	(SL, { SSD })	(ISL, { ID })
System managers and auditors	(AM, { SP, SD, SSD })	(ISL, { IP, ID})
System controllers	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID})
Repair	(SL, { SP })	(ISL, { IP })

ISP > IO > ISL

Lipner's Integrity Matrix

Security and integrity levels for subjects

- The integrity classes are chosen to allow modification of data and programs as appropriate
- For Example:
  - Ordinary users should be able to modify production data, so users of that class must have write access to integrity category IP
  - App developers should have write access to integrity category ID

Subjects	Security Level	Integrity Level
Ordinary users	(SL, { SP })	(ISL, { IP })
Application developers	(SL, { SD })	(ISL, { ID })
System programmers	(SL, { SSD })	(ISL, { ID })
System managers and auditors	(AM, { SP, SD, SSD })	(ISL, { IP, ID})
System controllers	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID})
Repair	(SL, { SP })	(ISL, { IP })

- Table shows the integrity levels and security categories of users

ISP > IO > ISL

Lipner's Integrity Matrix

Comparison of Old and New Security Levels

	Original	New	New
Subjects	**Confidentiality Level**	**Confidentiality Level**	**Integrity Level**
Ordinary users	(SL, { PC, PD })	(SL, { SP })	(ISL, { IP })
Application developers	(SL, { D, T })	(SL, { SD })	(ISL, { ID })
System programmers	(SL, { SD, T })	(SL, { SSD })	(ISL, { ID })
System managers and auditors	(AM, { D, OC, OD, SD, T })	(AM, { SP, SD, SSD })	(ISL, { IP, ID})
System controllers	(SL, {D, PC, PD, SD, T}) and downgrade privilege	(SL, { SP, SD }) and downgrade privilege	(ISP, { IP, ID})
Repair	Not available	(SL, { SP })	(ISL, { IP })

Here downgrade means the ability to move software (objects) from development to production

ISP > IO > ISL

Lipner's Integrity Matrix

Objects and Classifications

- The final step is to select integrity classes for objects
- Consider the objects Production Code and Production Data
- Ordinary users must be able to:
  - write production data, but not production code
- By placing:
  - Production Data in integrity class (ISL, {IP}) and
  - Production Code in integrity class (IO, {IP})

  an ordinary user cannot alter production code but can alter production data (IO > ISL)
- Similar analysis leads to the levels shown in the next table

Lipner's Integrity Matrix

Security and integrity levels for objects

Objects	Security Level	Integrity Level
Development code/test data	(SL, { SD })	(ISL, { IP})
Production code	(SL, { SP })	(IO, { IP })
Production data	(SL, { SP })	(ISL, { IP })
Software tools	(SL, ∅)	(IO, { ID })
System programs	(SL, ∅)	(ISP, { IP, ID })
System programs in modification	(SL, { SSD })	(ISL, { ID })
System and application logs	(AM, { *appropriate* })	(ISL, ∅)
Repair	(SL, {SP})	(ISL, { IP })

ISP > IO > ISL

Thank You!