Mandatory Access Control (MAC)

Prof. Ravi Sandhu
Executive Director and Endowed Chair

Lecture 8

ravi.utsa@gmail.com
www.profsandhu.com

Denning's Axioms for Information Flow

Denning's Axioms

$$< SC, \rightarrow, \oplus >$$

SC	set of security classes
$\rightarrow \subseteq$ SC X SC	flow relation (i.e., can-flow)
$\oplus$: SC X SC -> SC	class-combining operator

Denning's Axioms

$$< SC, \rightarrow, \oplus >$$

1. SC is finite
2. $\rightarrow$ is a partial order on SC
   (i.e., reflexive, transitive, anti-symmetric)
3. SC has a lower bound L such that L $\rightarrow$ A for all A $\in$ SC
4. $\oplus$ is a least upper bound (lub) operator on SC

Justification for 1 and 2 is stronger than for 3 and 4.
In practice we may have a partially ordered set (poset).

Denning's Axioms Imply

- SC is a universally bounded lattice
- There exists a Greatest Lower Bound (glb) operator $\otimes$ (also called meet)
- There exists a highest security class H

Lattice Structures

Hierarchical Classes

Top Secret

Secret

Confidential

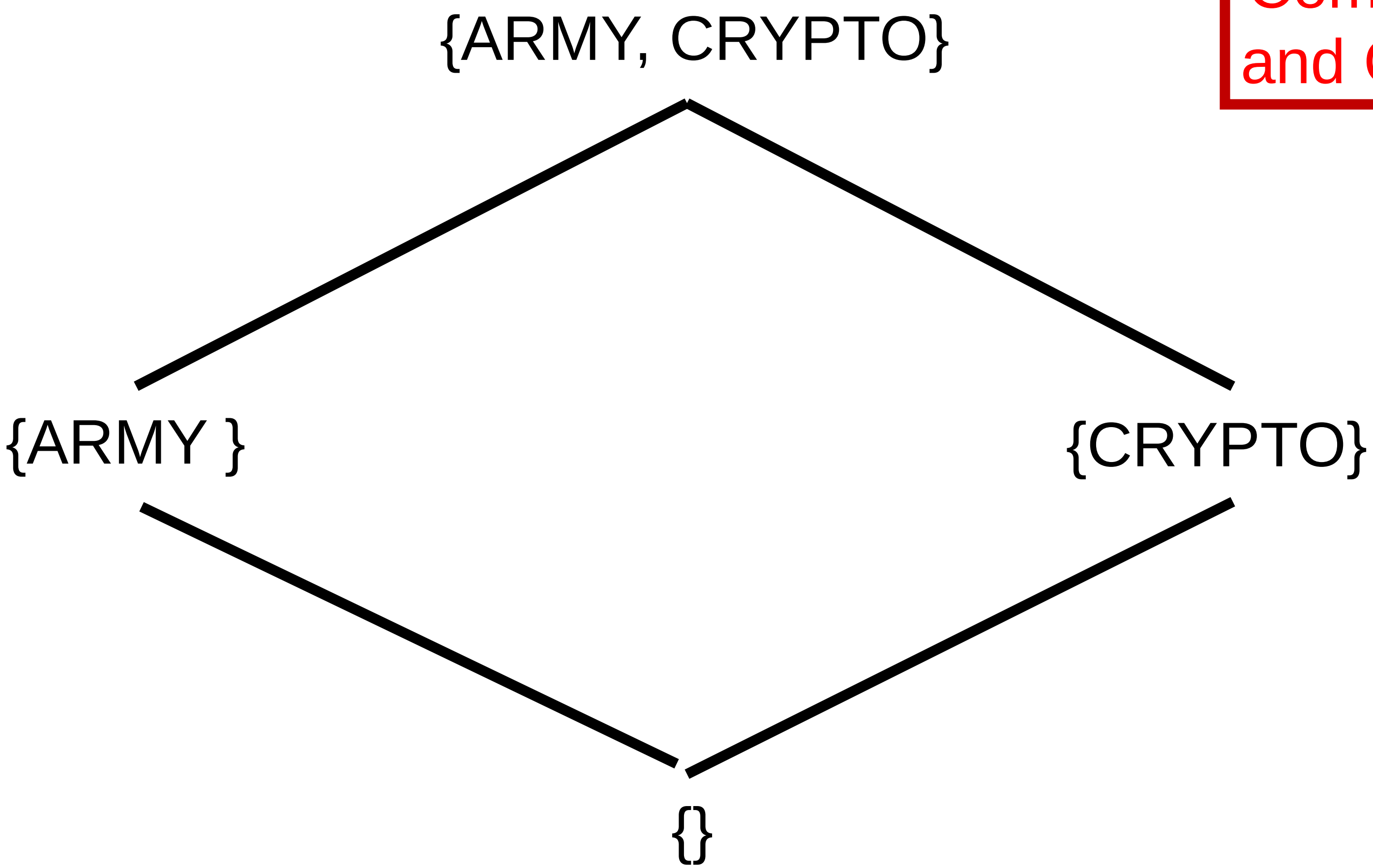
Unclassified

can-flow

reflexive and transitive edges
are implied but not shown

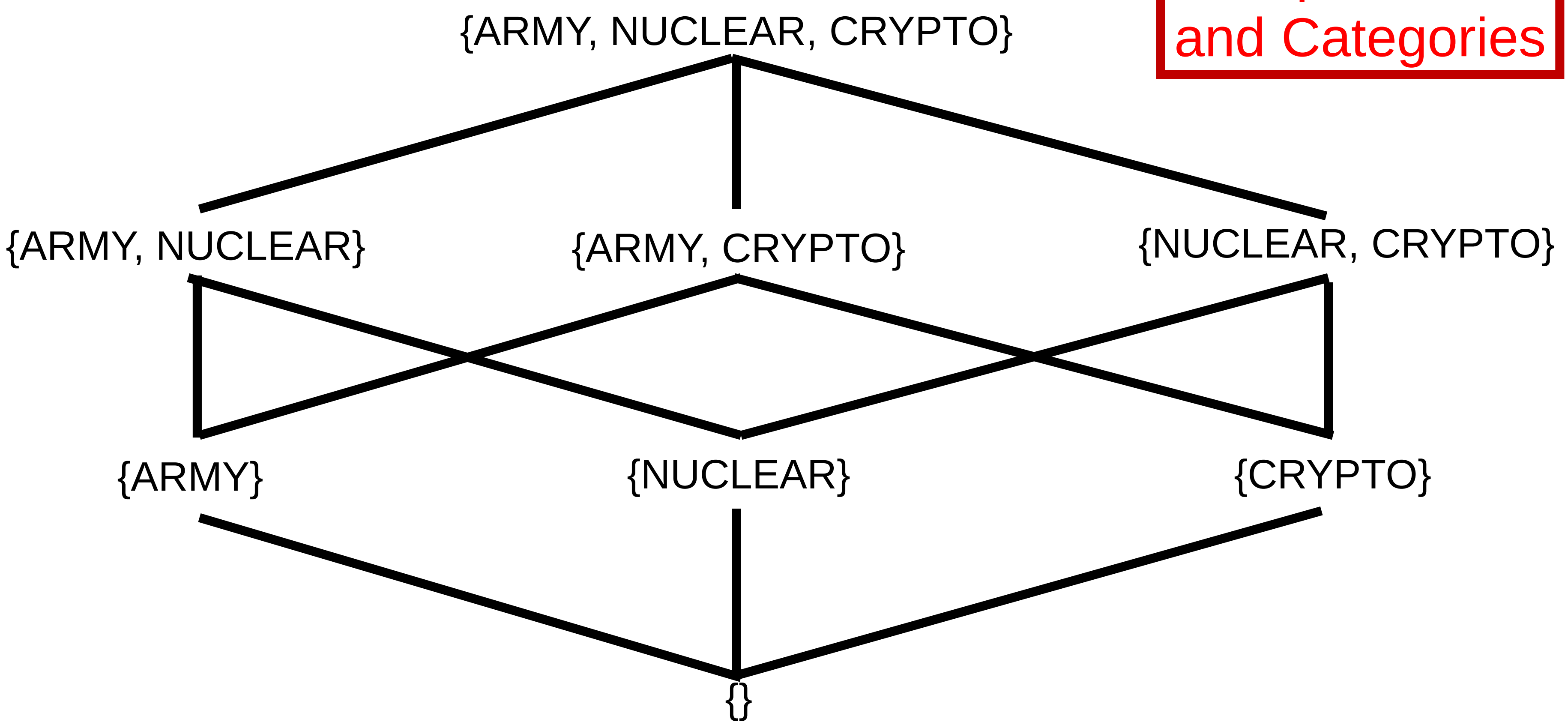
Lattice Structures

Compartments and Categories



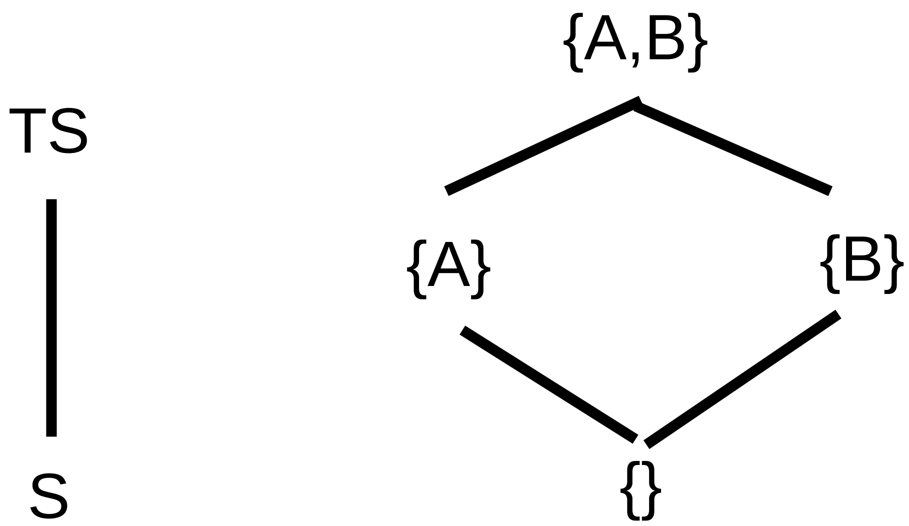
Lattice Structures

Compartments and Categories



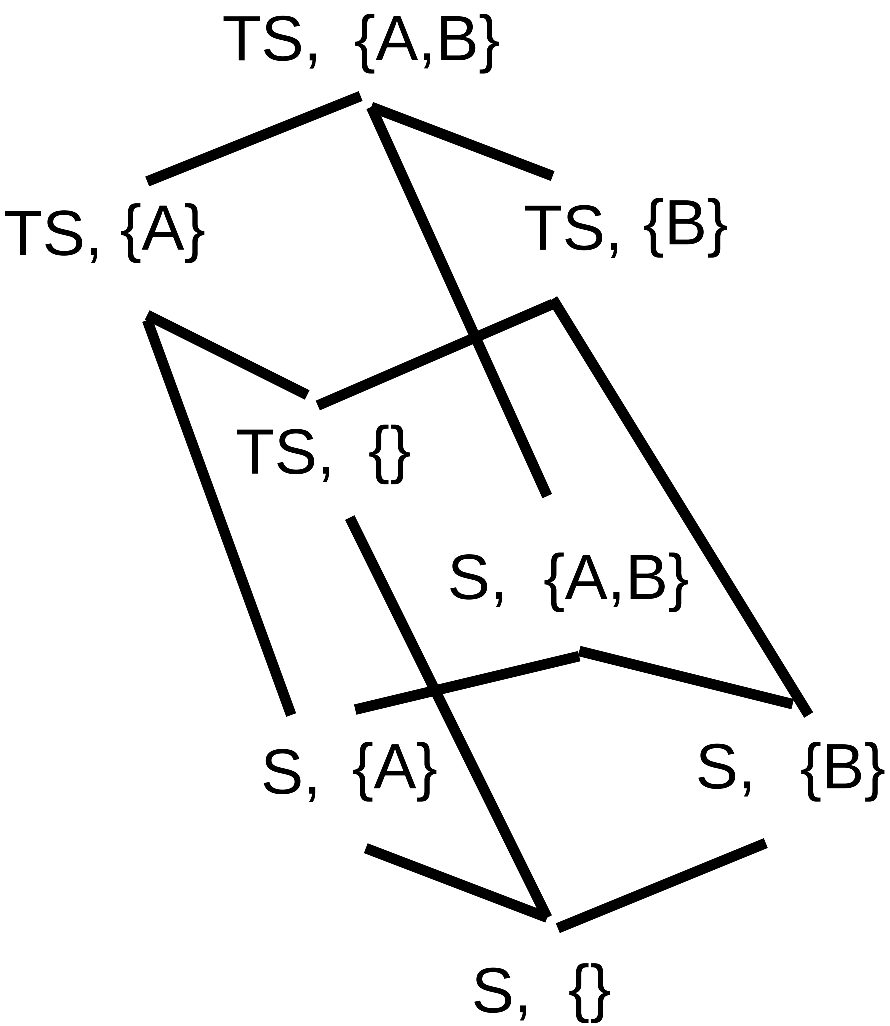
Lattice Structures

Hierarchical Classes with Compartments



product of 2 lattices is a lattice

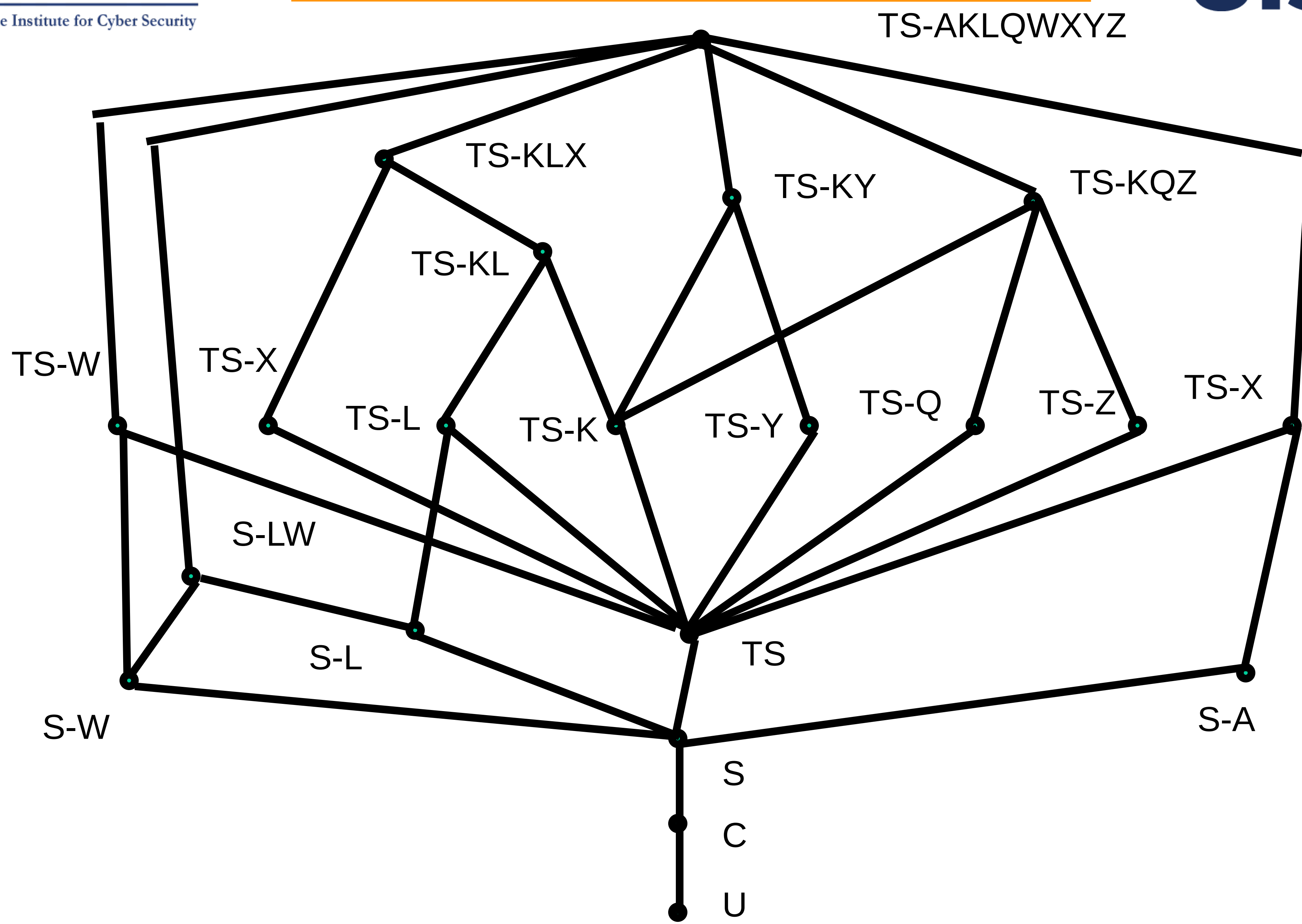
Lattice Structures



Hierarchical Classes with Compartments

product of 2 lattices is a lattice

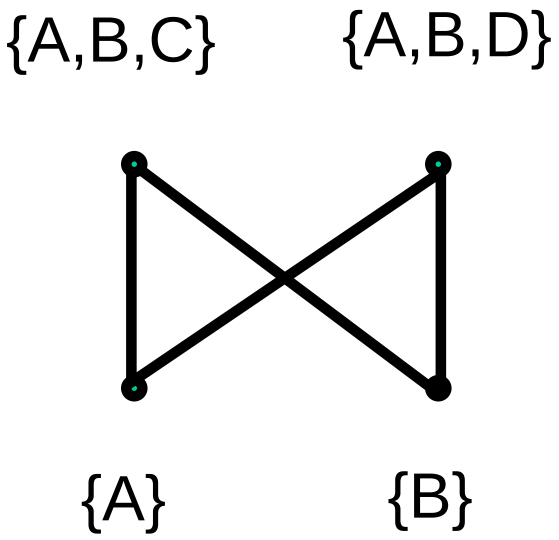
Smith's Lattice



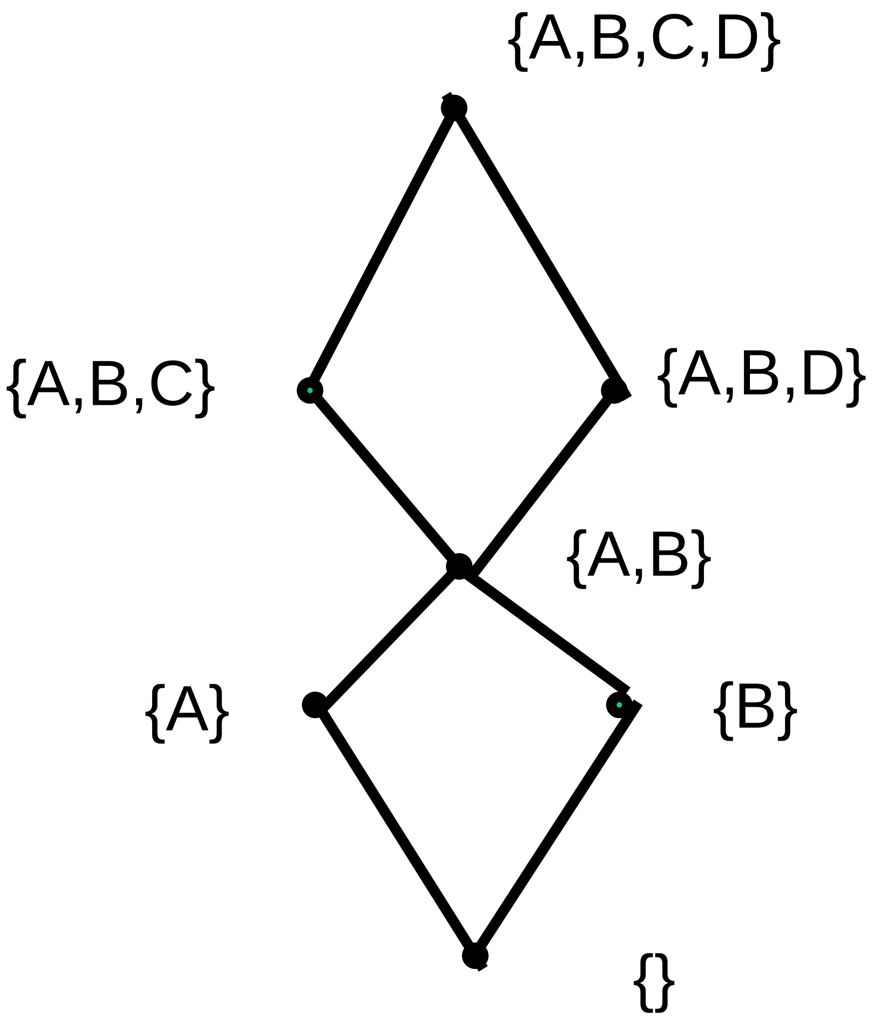
Smith's Lattice

- With large lattices a vanishingly small fraction of the labels will actually be used
  - Smith's lattice: 4 hierarchical levels, 8 compartments
  - number of possible labels = 4*2^8 = 1024
    Only 21 labels are actually used (2%)
- Consider 16 hierarchical levels, 64 compartments which gives 10^20 labels

Extending a POSET to a Lattice



$\Rightarrow$



such extension
is always possible

BLP Model for Confidentiality

BLP Basic Assumptions

- SUB = {S1, S2, ..., Sm}, a fixed set of subjects
- OBJ = {O1, O2, ..., On}, a fixed set of objects
- R = {r, w}, a fixed set of rights
- D, an $m \times n$ discretionary access matrix with $D[i,j] \subseteq R$
- M, an $m \times n$ current access matrix with $M[i,j] \subseteq R$

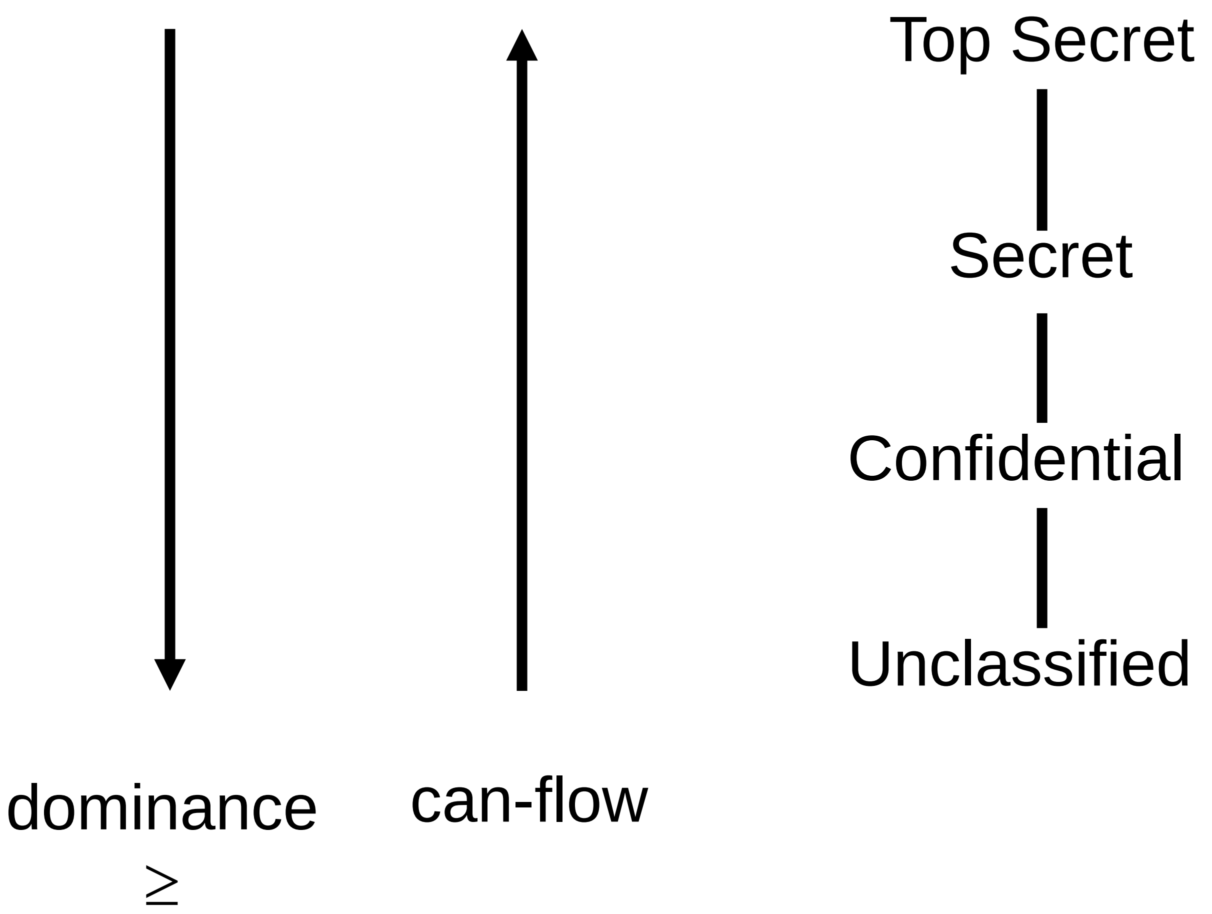
BLP Model (Liberal ★-Property)

- Lattice of confidentiality labels $\Lambda = \{\lambda 1, \lambda 2, ..., \lambda p\}$
- Static assignment of confidentiality labels λ: SUB $\cup$ OBJ $\rightarrow \Lambda$
- M, an m $\times$ n current access matrix with
  - $r \in M[i,j] \Rightarrow r \in D[i,j] \wedge \lambda(Si) \geq \lambda(Oj)$ simple security
  - $w \in M[i,j] \Rightarrow w \in D[i,j] \wedge \lambda(Si) \leq \lambda(Oj)$ liberal ★-property

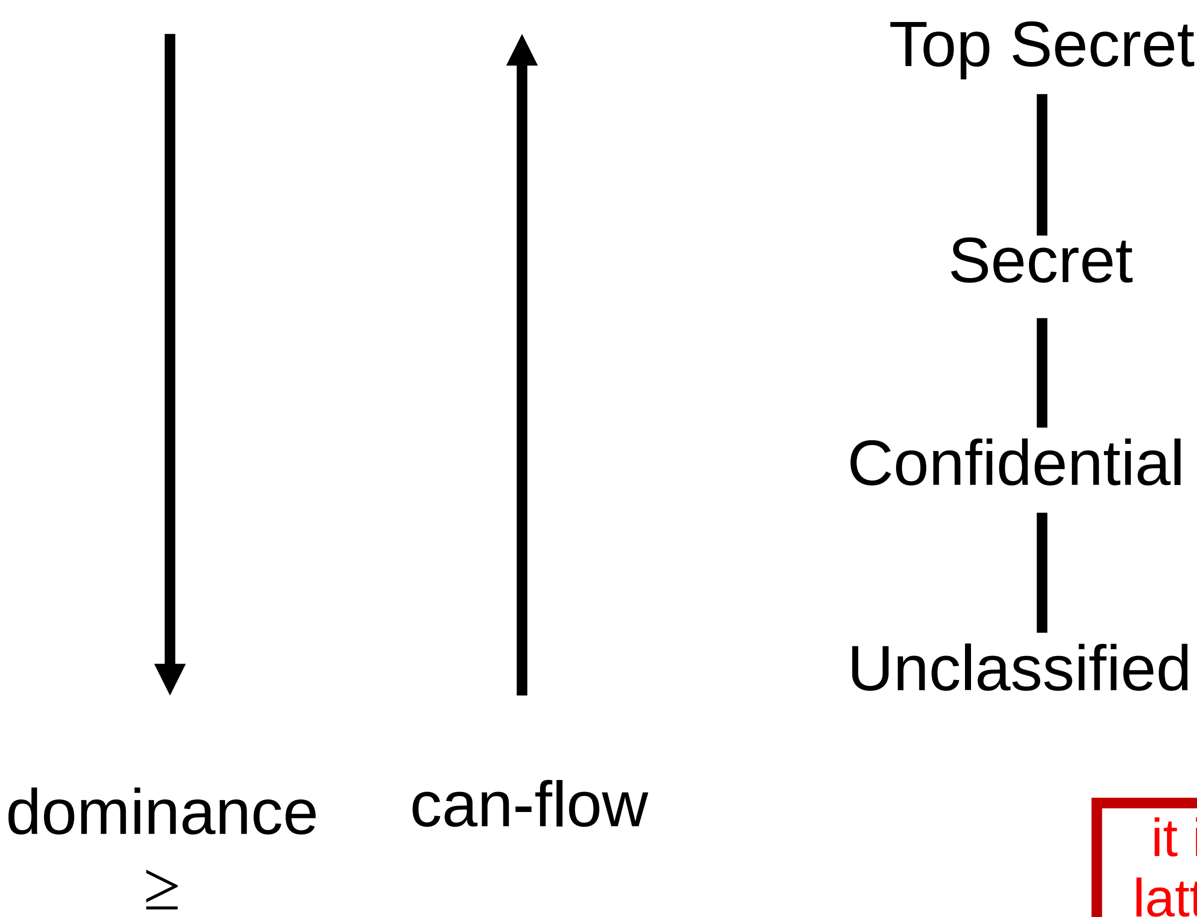
BLP Model (Strict ★-Property)

- Lattice of confidentiality labels $\Lambda = \{\lambda 1, \lambda 2, ..., \lambda p\}$
- Static assignment of confidentiality labels λ: SUB $\cup$ OBJ $\rightarrow \Lambda$
- M, an m $\times$ n current access matrix with
  - $r \in M[i,j] \Rightarrow r \in D[i,j] \wedge \lambda(Si) \geq \lambda(Oj)$ simple security
  - $w \in M[i,j] \Rightarrow w \in D[i,j] \wedge \lambda(Si) = \lambda(Oj)$ strict ★-property

BLP vis a vis Lattices

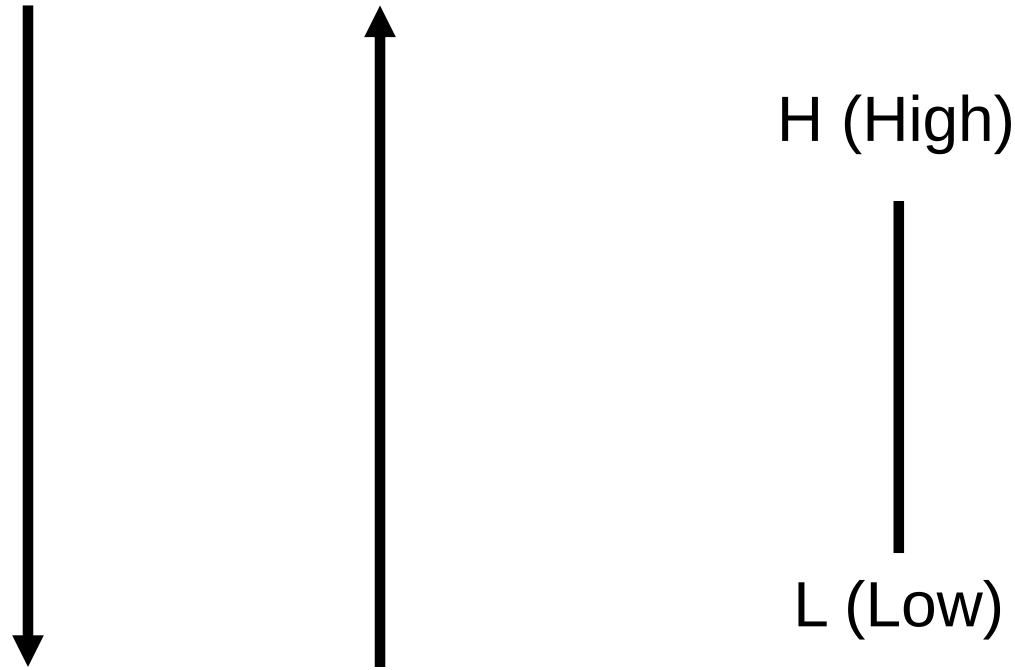


BLP vis a vis Lattices



it is risky to visualize
lattices as total orders
but it is ok sometimes

BLP vis a vis Lattices



dominance
$\geq$

can-flow

often 2 levels suffice to make the main point

★-Property

- Applies to subjects not to users
  - Users are trusted (must be trusted) not to disclose secret information outside of the computer system
  - A user can login (create a subject) with any label dominated by the user's clearance
  - Subjects are not trusted because they may have Trojan Horses embedded in the code they execute
- ★-property prevents deliberate leakage and does not address
  - inference
  - covert channels
- Simple-security and ★-Property do not account for
  - encryption

Biba Model
for
Integrity

BLP Revisited

HS (High Secrecy)

LS (Low Secrecy)

dominance

$\geq$

can-flow

Biba Inverted Flow

HI (High Integrity)

LI (Low Integrity)

dominance

$\geq$

can-flow

Biba and BLP Aligned: BLP Style

HS (High Secrecy) LI (Low Integrity)

LS (Low Secrecy) HI (High Integrity)

dominance
$\geq$

can-flow

One-directional flow is the key point
No need for opposite directions for confidentiality and integrity

Biba and BLP Aligned: Biba Style

LS (Low Secrecy) HI (High Integrity)

HS (High Secrecy) LI (Low Integrity)

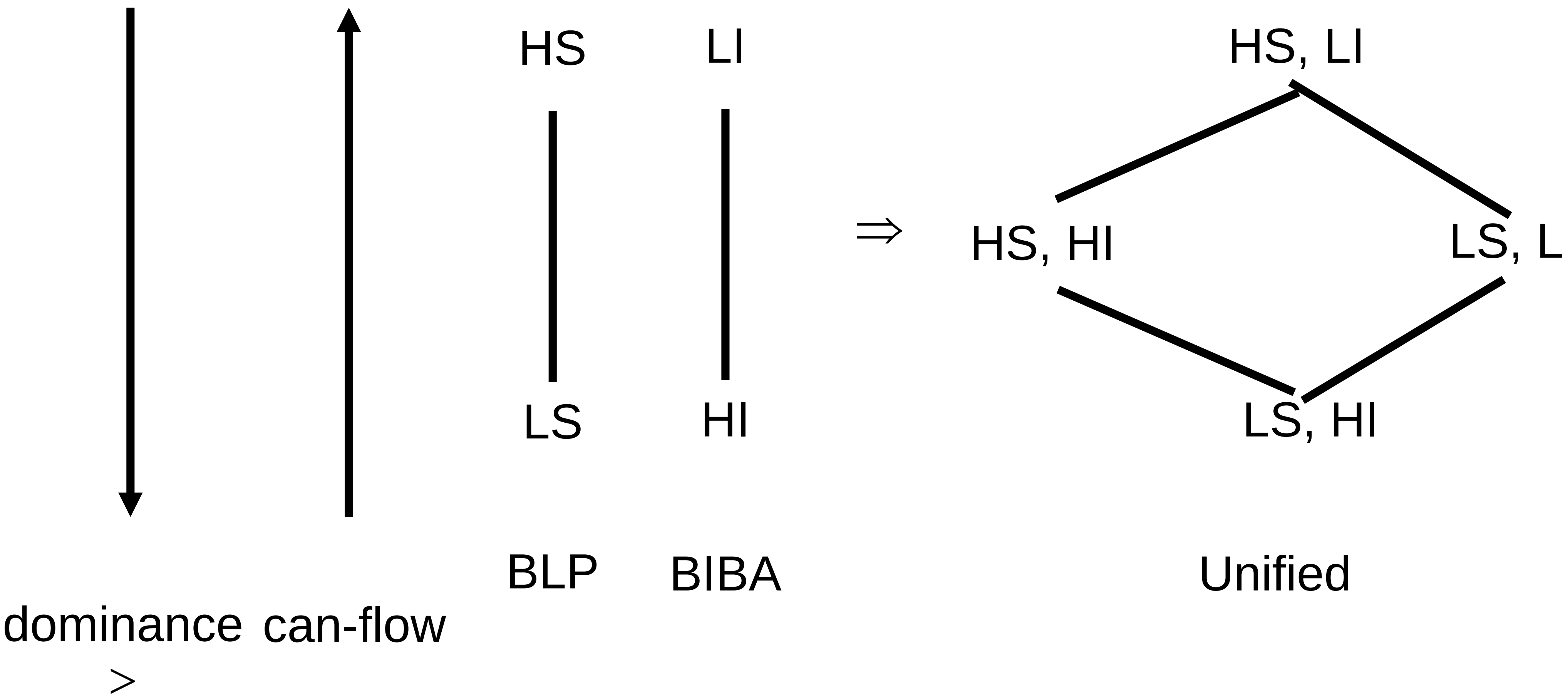
dominance

$\geq$

can-flow

One-directional flow is the key point
No need for opposite directions for confidentiality and integrity

BLP-Biba Unified Lattice: BLP Style



BLP versus Biba

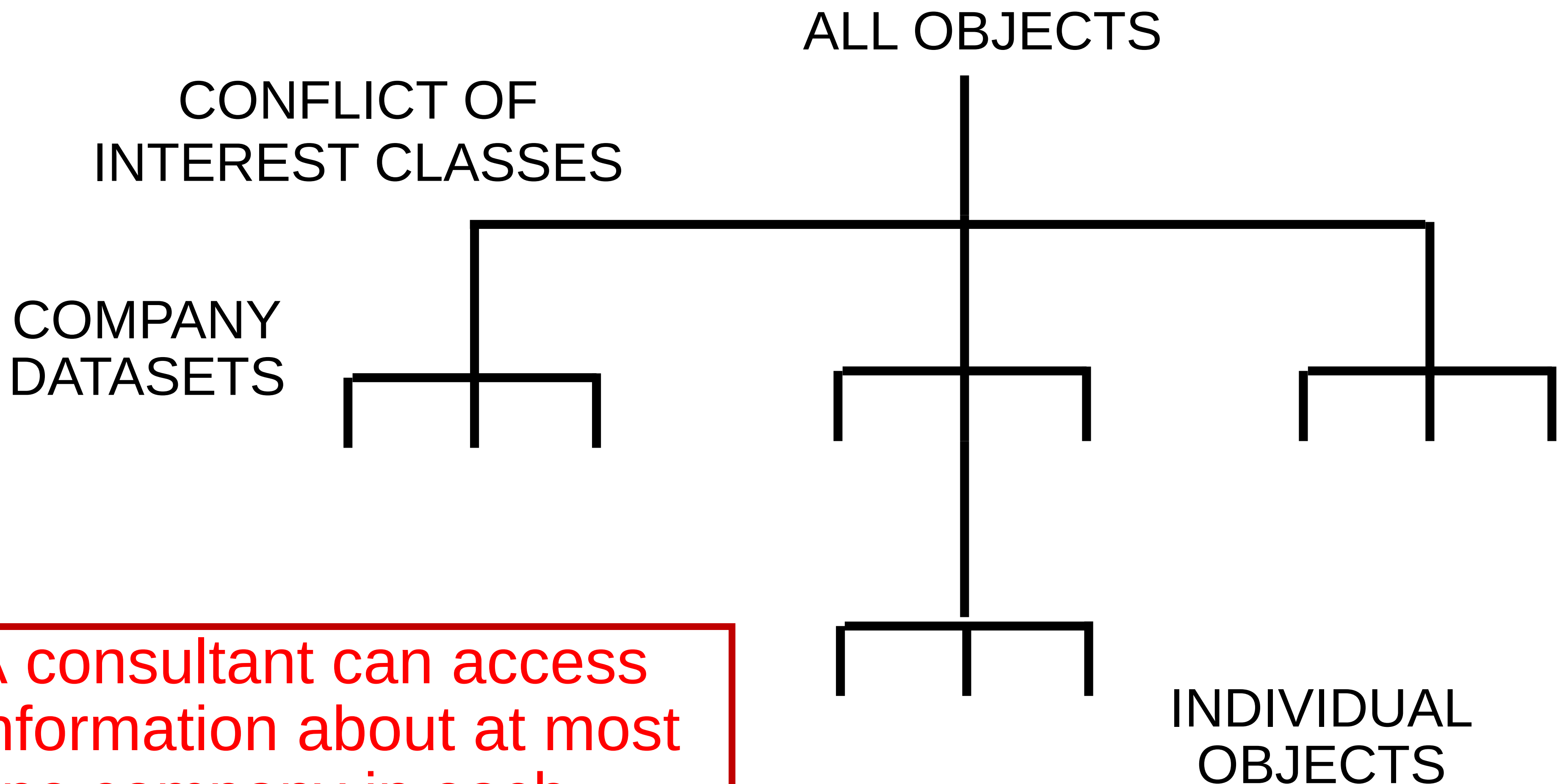
- BLP and Biba are fundamentally equivalent and interchangeable
- Lattice-based access control is a mechanism for enforcing one-way information flow, which can be applied to confidentiality or integrity goals
- We will use the BLP formulation:
  - high confidentiality, low integrity at the top
  - low confidentiality, high integrity at the bottom

The Chinese Wall Lattice for Separation of Duty

Chinese Wall Policy

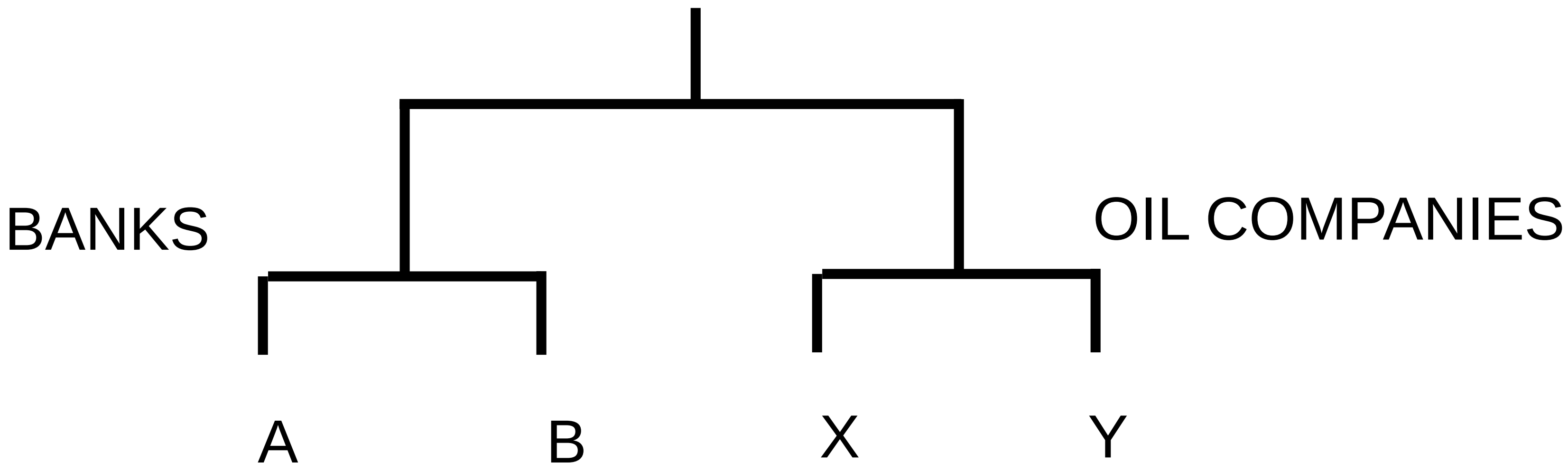
- A commercial security policy for separation of duty driven confidentiality
- Mixture of free choice (discretionary) and mandatory controls
- Requires some kind of dynamic labelling

Chinese Wall Policy

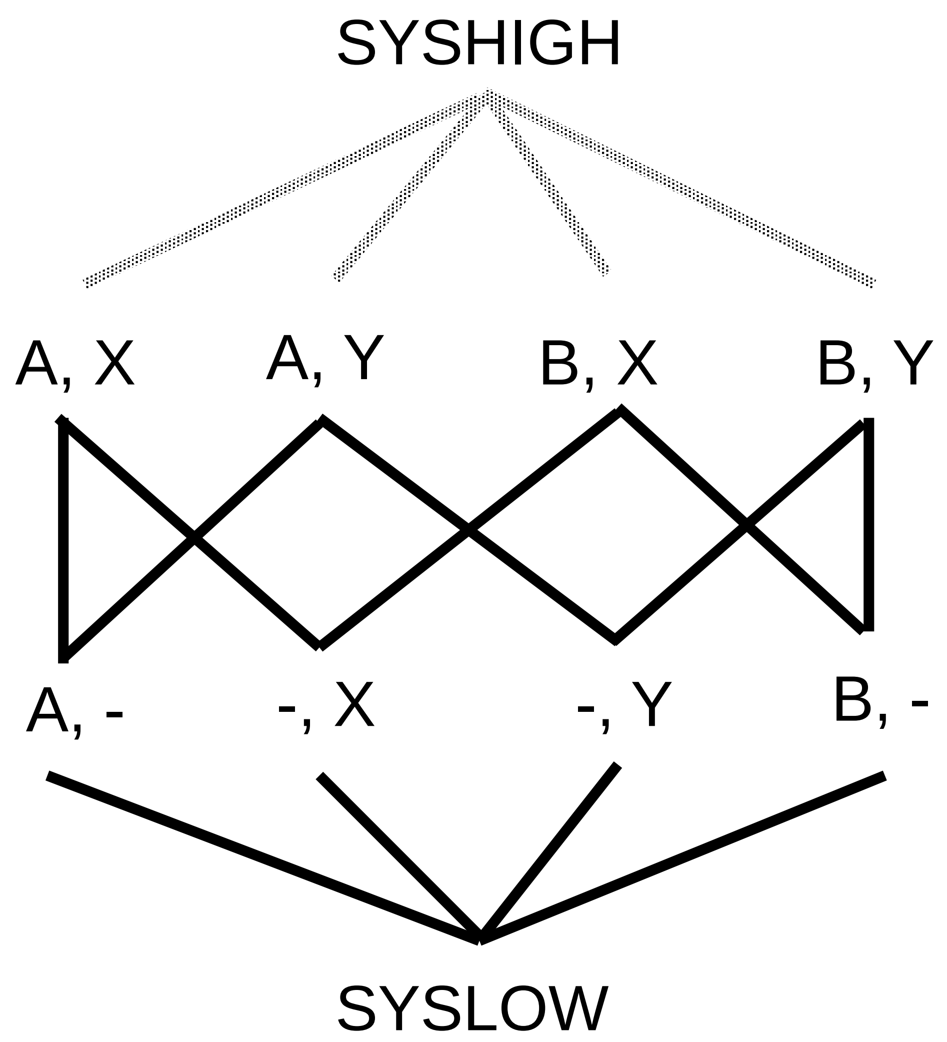


A consultant can access information about at most one company in each conflict of interest class

Chinese Wall Example



Chinese Wall Lattice



Conclusion

MAC or LBAC or BLP (or Biba)

- BLP enforces one-directional information flow in a lattice of security labels

Enforcement

- BLP can enforce one-directional information flow policies for
  - Confidentiality
  - Integrity
  - Separation of duty
  - Combinations thereof

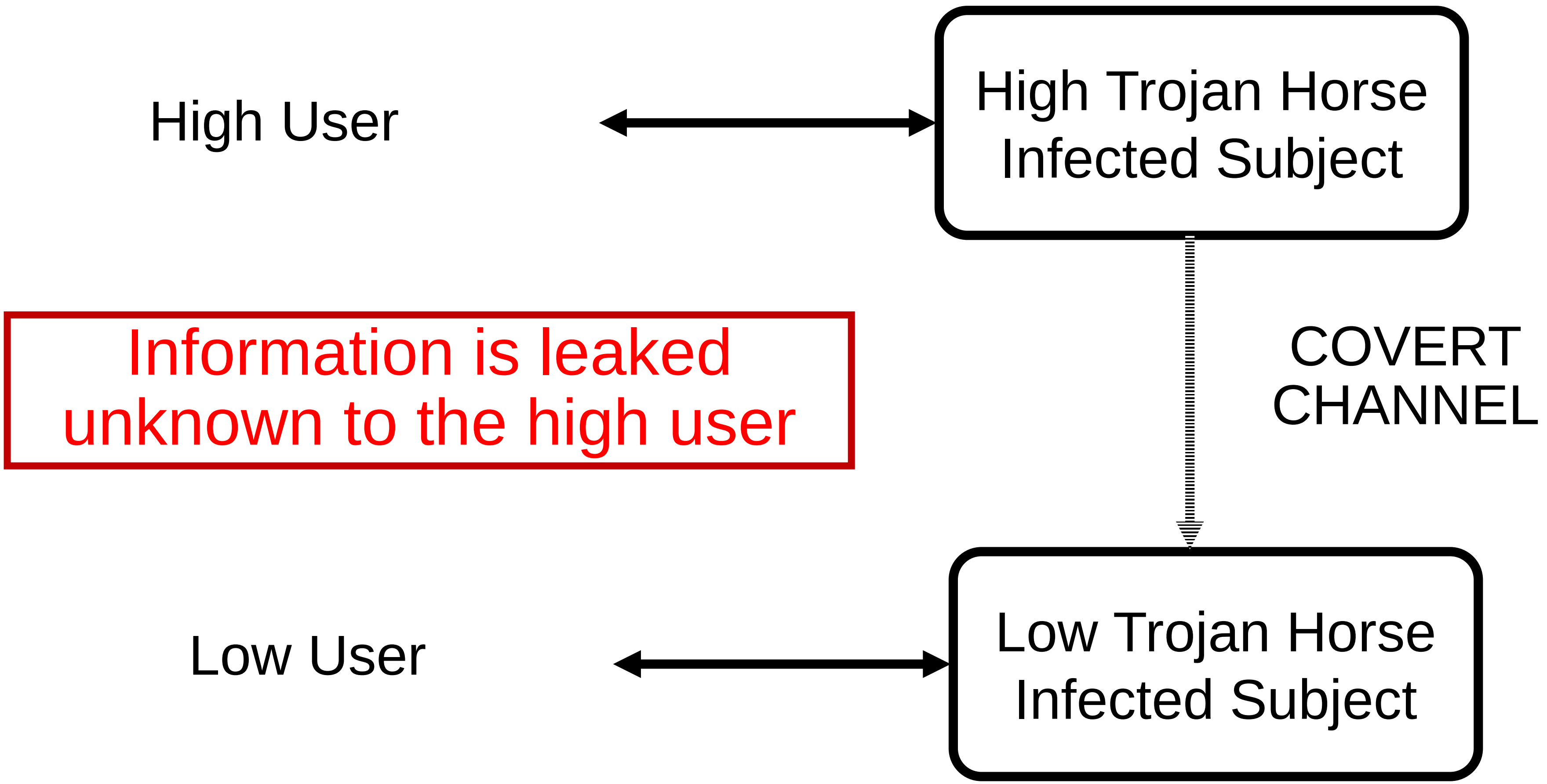
Policy

Covert Channels

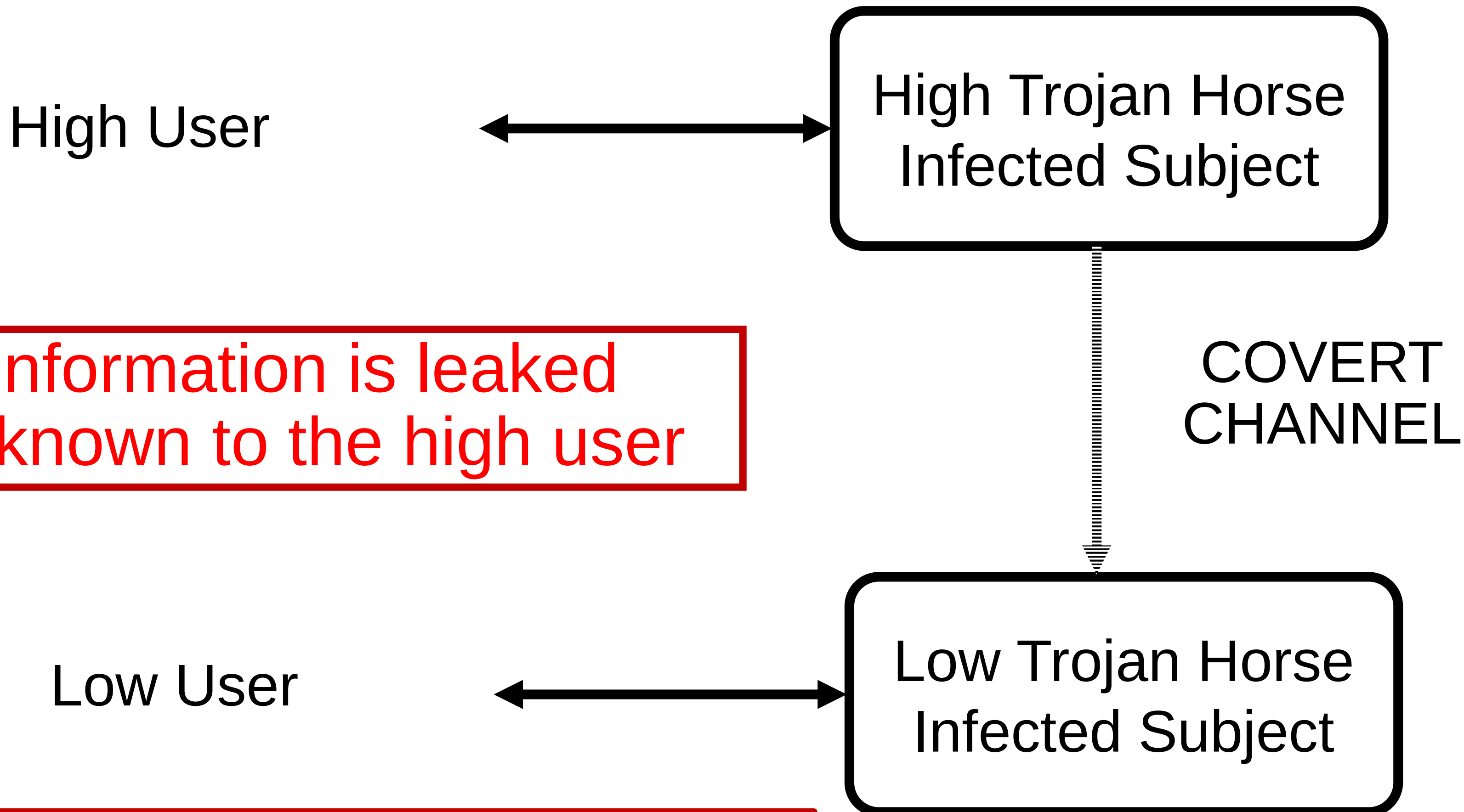
Covert Channels

- A covert channel is a communication channel based on the use of system resources not normally intended for communication between subjects (processes)

Covert Channels



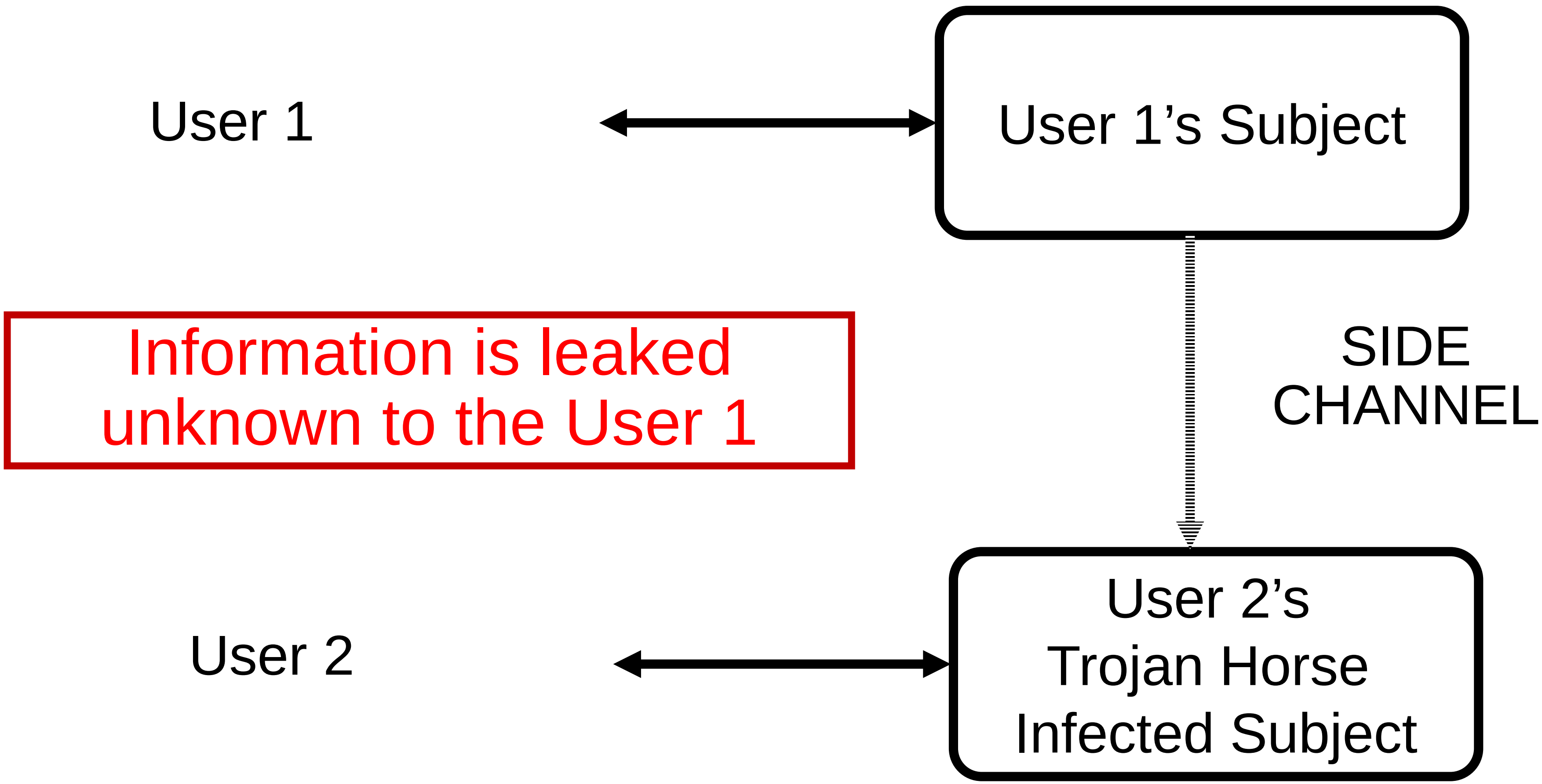
Covert Channels



Information is leaked unknown to the high user

★-property prevents overt leakage of information and does not address covert channels

Side Channels



Covert Channels versus Side Channels

- Covert channels require a cooperating sender and receiver
- Side channels do not require a sender but nevertheless information is leaked to a receiver

- Identify the channel
  - close the channel or slow it down
  - detect attempts to use the channel
  - tolerate its existence

Storage Channels

- Also known as Resource Exhaustion Channels
- Given 5GB pool of dynamically allocated memory
  - HIGH PROCESS (sender)
    bit = 1 $\Rightarrow$ request 5GB of memory
    bit = 0 $\Rightarrow$ request 0GB of memory
  - LOW PROCESS (receiver)
    request 5GB of memory
    if allocated then bit = 0 otherwise bit = 1

Timing Channels

- Also known as Load Sensing Channels
- Given 5GB pool of dynamically allocated memory
  - HIGH PROCESS (sender)
    bit = 1 $\Rightarrow$ enter computation intensive loop
    bit = 0 $\Rightarrow$ go to sleep
  - LOW PROCESS (receiver)
    perform a task with known computational requirement
    if completed promptly then bit = 0 otherwise bit = 1