A Complete Generative Label Model for LBAC Models

N.V. Narendra Kumar

IDRBT, Hyderabad, INDIA

(Jointly with Prof. R.K. Shyamasundar, IIT Bombay, INDIA)

Overview

- Security Models
- Information-Flow Security
- Limitations of Current Models
- RW Labels – A Complete Label Model
- Characteristic Properties of B-RWFM
- Expressive Power and Simplicity of RWFM
- Conclusions and Future Work

Security Models

Confidentiality – Bell and LaPadula

- Levels (sensitivity)
- Compartments (need-to-know)
- Label = (level, set of compartments)
- Policy: Information flow lower to higher labels
- Enforcement: No-read-up + No-write-down

Integrity – Biba

- Levels (trustworthiness)
- Compartments (competence)
- Label = (level, set of compartments)
- Policy: Information flow higher to lower labels
- Enforcement: No-read-down + No-write-up

Non-Interference

- Goguen and Meseguer (authority-based): User ‘A’ is said to “not interfere with” user ‘B’ in a system ‘S’ if B’s observation of S is unchanged due to the actions of A

- Volpano and Smith (purely value-based): A program is said to be non-interfering if the value of high-inputs have no impact on the low-outputs

Information-Flow Security

Information-Flow Control (IFC)

- Broadly captures the following: "*information in an object may be allowed to flow into only those objects which have stricter policies*"

- Labels both subjects and objects, and controls the flow of information based on the labels

Information-Flow Control (IFC)

Introduction

- form of mandatory control for security
- subsumes confidentiality and integrity
- security by design
- compositional
- end-to-end security guarantees

Information-Flow Control (IFC)
Basics

- Information-flow control works as follows
  - assign labels to subjects and objects for tracking the flow of information in the system
  - define access rules (read and write) in terms of the can-flow-to relation on labels

- Labels play a crucial role in IFC systems

Denning's Lattice Model

- Flow model = (S, O, L, $\leq$, $\oplus$)
  - Subjects
  - Objects
  - Labels
  - Can-flow-to relation among labels
  - Provides label for aggregate information
- Unifies earlier models and is the basis for TCSEC (precursor to *Common Criteria*)

Brief IFC History

- Bell and LaPadula 1972 – Confidentiality Policy
- Denning 1975 – Lattice Model
- Myers and Liskov 1997-2000 – DLM and DIFC
- Butler Lampson 2011 – Technical Perspective
- Mitchell et al. 2012 – DC Labels

Limitations of Current Models

Drawbacks of State-of-the-art (1)

- 1985 TCSEC (Orange Book)
  - defines the security of a computer system by how well it implements flow control and how good its assurance is
- Despite huge efforts, systems developed had several drawbacks:
  - large TCB, slow, not easy to use, and very limited functionality

Drawbacks of State-of-the-art (2)

- 1997-2000 Myers & Liskov (DLM)
  - only readers for protecting confidentiality and only writers for protecting integrity
  - Concern: *for a proper tracking of any information flow property, it is important to control both reading and writing by subjects*

Drawbacks of State-of-the-art (3)

- 2006-2009 HiStar, Flume and Laminar systems
  - based on the product of confidentiality and integrity
  - Concern: *confidentiality and integrity are not orthogonal properties*
  - Fred Schneider, in his book[#] chapter, clearly brings out the perils of combining confidentiality and integrity policies in this manner

yet to be published,
available at http://www.cs.cornell.edu/fbs/publications/chptr.MAC.pdf

Drawbacks of State-of-the-art (4)

- 2012 Mitchell et al. (DC labels)
  - not easy to derive consistent DC labels for modelling a given end-to-end requirement
  - Concern: *support for downgrading (discretionary control) is orthogonal to the IFC, thus, defeating the purpose of the mandatory controls*

RW Labels – A Complete Label Model

Recasting Denning's Lattice

1. Make the labels explicit

2. Incorporate the flow semantics into labels

Make the Labels Explicit

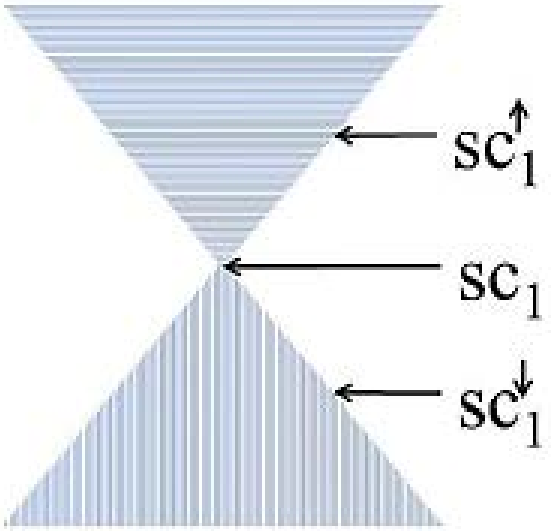
$$sc^{\uparrow} \triangleq \{sc' \in SC \mid sc \leqslant sc'\}$$

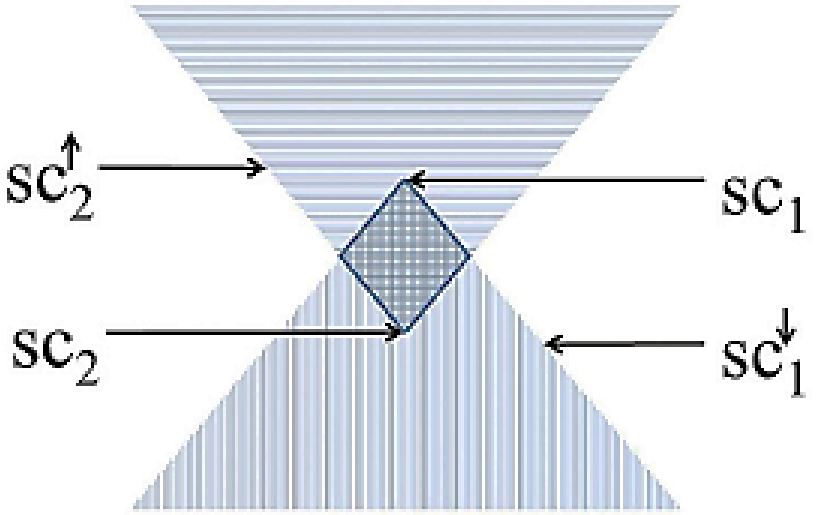
$$sc^{\downarrow} \triangleq \{sc' \in SC \mid sc' \leqslant sc\}$$

Theorem 1: Given a Denning's flow model $DFM = (S, O, SC, \leqslant, \oplus)$, let $DFM' = (S, O, SC', \leqslant', \oplus')$ where $SC' = 2^{SC} \times 2^{SC}$, $(A_1, B_1) \leqslant' (A_2, B_2) \triangleq ((A_1 \supseteq A_2) \wedge (B_1 \subseteq B_2))$, and $(A_1, B_1) \oplus' (A_2, B_2) \triangleq (A_1 \cap A_2, B_1 \cup B_2)$. The function $f : SC \rightarrow SC'$ defined by $f(sc) = (sc^{\uparrow}, sc^{\downarrow})$ is such that for any two elements sc_1 and sc_2 of SC,

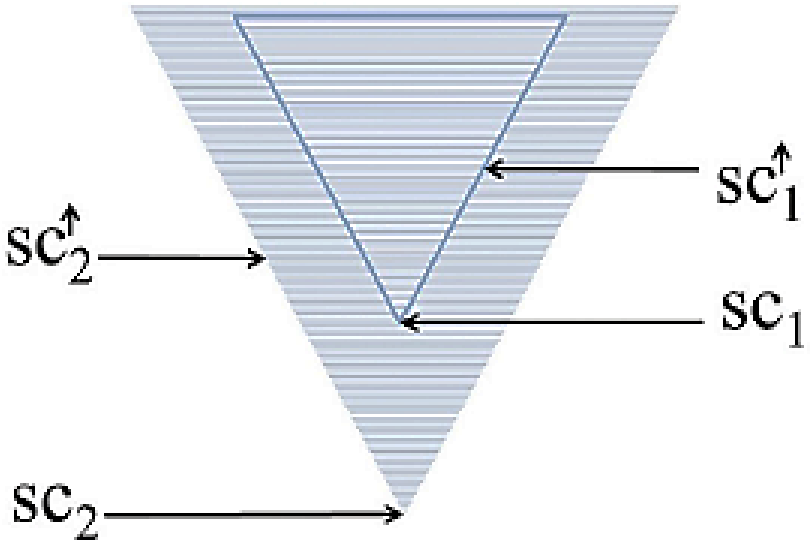
$$sc_1 \leqslant sc_2 \text{ if and only if } f(sc_1) \leqslant' f(sc_2).$$

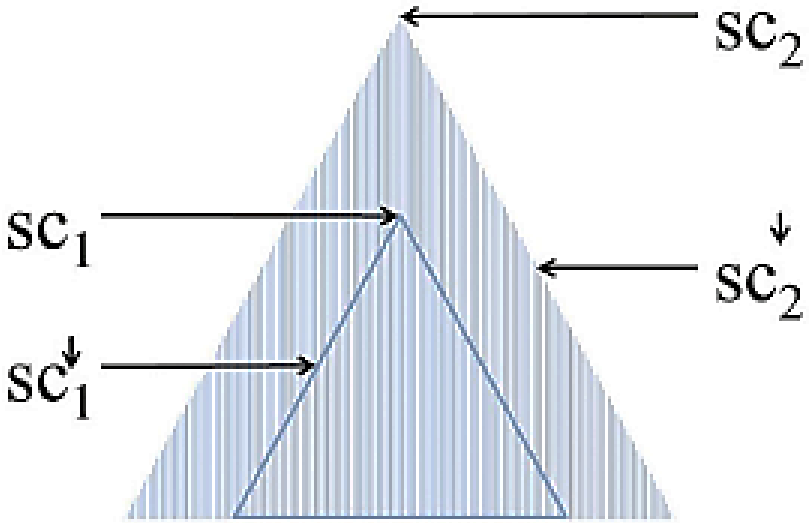
Intuitions (1)





Intuitions (2)





Incorporate the Flow Semantics into Labels

Definition 1 [Readers and Writers]**:** Given a Denning's flow model $DFM = (S, O, SC, \leqslant, \oplus)$ together with a policy $\lambda : S \cup O \rightarrow SC$, we define the readers, denoted sc^R, and writers, denoted sc^W, of a security class $sc \in SC$ as:

- $sc^R \triangleq \{s \in S \mid \lambda(s) \in sc^{\uparrow}\}$,
- $sc^W \triangleq \{s \in S \mid \lambda(s) \in sc^{\downarrow}\}$.

Theorem 2: Given a Denning's flow model $DFM = (S, O, SC, \leqslant, \oplus)$ together with a policy $\lambda : S \cup O \rightarrow SC$, let $DFM'' = (S, O, SC'', \leqslant'', \oplus'')$ where $SC'' = 2^S \times 2^S$, $(A_1, B_1) \leqslant'' (A_2, B_2) \triangleq ((A_1 \supseteq A_2) \wedge (B_1 \subseteq B_2))$, and $(A_1, B_1) \oplus'' (A_2, B_2) \triangleq (A_1 \cap A_2, B_1 \cup B_2)$, and $\lambda'' : S \cup O \rightarrow SC''$ be such that $\lambda''(e) \triangleq (\lambda(e)^R, \lambda(e)^W)$, for $e \in S \cup O$. For any subject $s \in S$ and entity $e \in S \cup O$,

1. $\lambda(s) \leqslant \lambda(e)$ if and only if $\lambda''(s) \leqslant'' \lambda''(e)$, and
2. $\lambda(e) \leqslant \lambda(s)$ if and only if $\lambda''(e) \leqslant'' \lambda''(s)$.

Recasting Algorithm

Input Denning's model $DFM = (S, O, SC, \leqslant, \oplus)$ and policy $\lambda : S \cup O \rightarrow SC$

Output Flow model $DFM_1 = (S, O, SC_1, \leqslant_1, \oplus_1)$ and policy $\lambda_1 : S \cup O \rightarrow SC_1$

Procedure

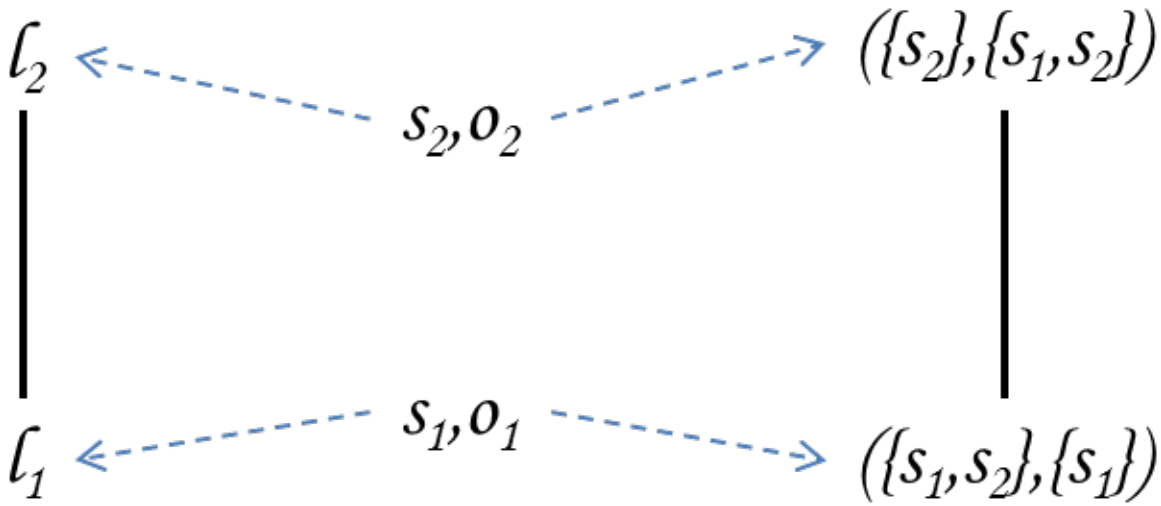
$SC_1 = 2^S \times 2^S$

$(A_1, B_1) \leqslant_1 (A_2, B_2) \triangleq [(A_1 \supseteq A_2) \wedge (B_1 \subseteq B_2)]$

$(A_1, B_1) \oplus_1 (A_2, B_2) \triangleq (A_1 \cap A_2, B_1 \cup B_2)$

$\lambda_1(e) = (\{s \in S \mid \lambda(e) \leqslant \lambda(s)\}, \{s \in S \mid \lambda(s) \leqslant \lambda(e)\})$, where $e \in S \cup O$

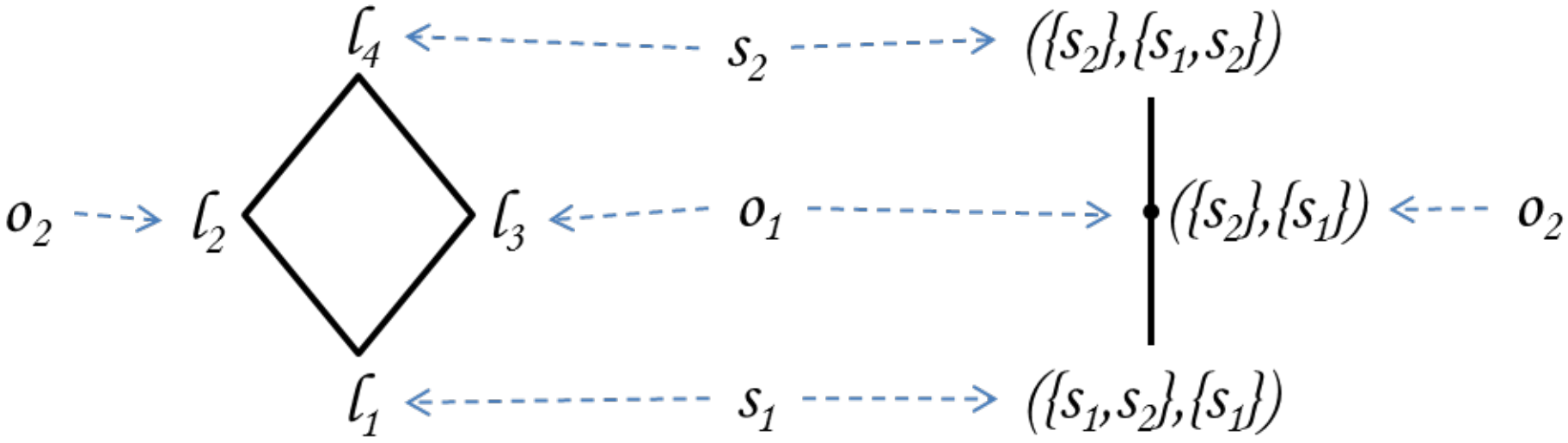
Example – 1



Denning's Policy

Readers-Writers Policy

Example – 2



Denning's Policy

Readers-Writers Policy

RW Labels

- Structure = (R,W)
  - R: authorized readers of the information
  - W: authorized writers of the information
- Information-flow allowed only when readers decrease and writers increase
- [Sound w.r.t. Denning] RW labels form a bounded lattice; defines B-RWFM

Characteristic Properties of B-RWFM

B-RWFM

Definition 6 [Access Rules in B-RWFM]**:** Given a B-RWFM, and functions R and W describing a labelling,

- a subject s is allowed to read an object o if $R(o) \supseteq R(s)$, $W(o) \subseteq W(s)$, and
- a subject s is allowed to write an object o if $R(s) \supseteq R(o)$ and $W(s) \subseteq W(o)$.

Theorem 4 [Completeness of B-RWFM]**:** Given a Denning's flow model $DFM = (S, O, SC, \oplus, \leqslant)$ and a policy $\lambda : S \cup O \rightarrow SC$, there exists a labelling, $\lambda_B : S \cup O \rightarrow SC_B$, in the basic readers-writers flow model that enforces the same policy i.e.,

(i) s is permitted to read o by Denning's policy if and only if it is permitted by the basic readers-writers policy.
(ii) s is permitted to write o by Denning's policy if and only if it is permitted by the basic readers-writers policy.

Properties of B-RWFM

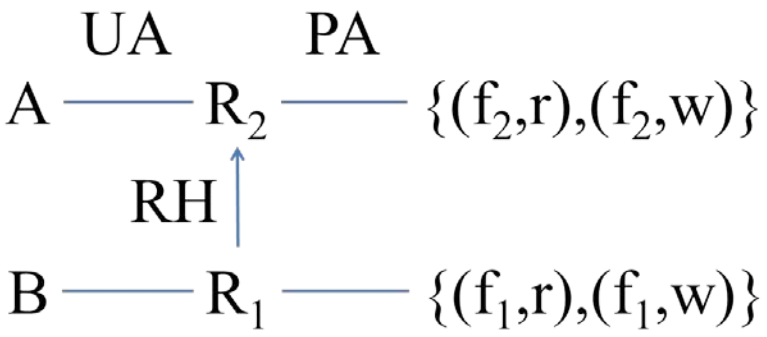
- $s \in R(s)$, and $s \in W(s)$
- $\forall s,o$:
  - $R(o) \supseteq R(s) \Rightarrow W(o) \subseteq W(s)$
  - $W(s) \subseteq W(o) \Rightarrow R(s) \supseteq R(o)$
- $\forall s,o$:
  - $s \in R(o) \Leftrightarrow R(o) \supseteq R(s)$
  - $s \in W(o) \Leftrightarrow W(s) \subseteq W(o)$
- Flow rules reduce to simple access checks !!

Information Dominance of Subjects

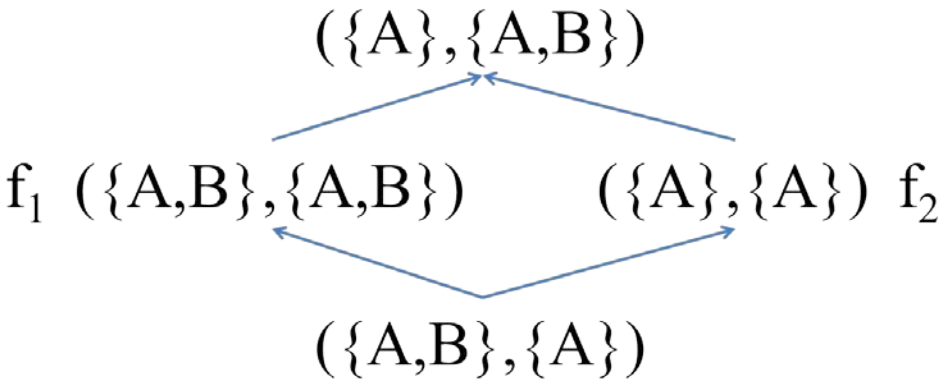
- s_1 read-dominates s_2: $s_1 \in R(s_2)$
- s_1 write-dominates s_2: $s_1 \in W(s_2)$
- s_1 information-dominates s_2:
  - $s_1 \in R(s_2) \wedge s_2 \in W(s_1)$
- [Principal Hierarchy] s_1 dominates s_2:
  - $s_1 \in R(s_2) \wedge s_1 \in W(s_2)$
  - In the flow context $\Rightarrow$ s1, s2 have the same label !
- Information-dominance better suited for flow

Expressive Power and Simplicity of RWFM

RBAC Encoded in B-RWFM



a. RBAC configuration



b. Induced lattice

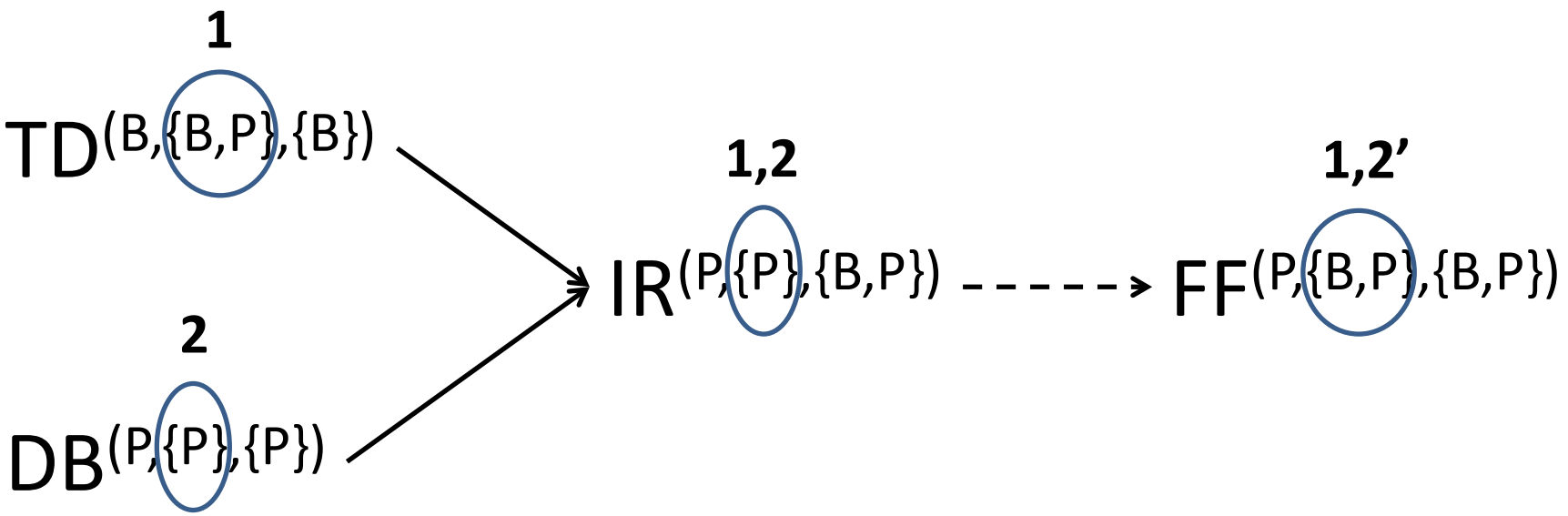
Example

WebTax – Description

- Bob provides his tax-data to a professional tax preparer, who computes Bob's final tax form using a private database of rules for minimizing the tax payable and returns the final form to Bob
- Security requirements
  1. Bob requires that his tax-data remains confidential
  2. Preparer requires that his private database remains confidential

Example

WebTax – Illustration



TD	Tax-data	IR	Intermediate results
DB	Database of tax optimization rules	FF	Final tax form
→	Flows-to	- ->	Downgraded-to

Example

WebTax – Comparison

	DLM	DC	RWFM
TD	{B: B}	(B, B)	(B, {B,P}, {B})
DB	{P: P}	(P, P)	(P, {P}, {P})
IR	{B: B; P: P}	(B∧P, B∨P)	(P, {P}, {B,P})
FF	{B: B}	(B, B∨P)	(P, {B,P}, {B,P})

Conclusions and Future Work

Conclusions

- Derived an intuitive and semantic label model complete w.r.t. Denning's model
- Illustrated the advantages of the RWFM model in the specification and enforcement of information flow properties
- Demonstrated that RWFM succinctly subsumes several security models

Ongoing Work

- Compare the security models
- Design a robust downgrading rule for RWFM
- Use RWFM to
  - Identify a basis for language-based security
  - Build a B1-secure (labelled security) general-purpose OS

Future Work

- Differentiate language and system security
- Design a protocol description language

Thank You