A Decentralized Model for Information Flow Control

Andrew C. Myers and Barbara Liskov, 1997

September 23, 2015

Mikael Elkiær Christensen
michri11@student.aau.dk

Department of Computer Science
Aalborg University
Denmark

AALBORG UNIVERSITY
DENMARK

Introduction

The result of this paper is a model for controlling information flow: Decentralized Label Model (DLM).

Introduction
What it is not

It is not:

Introduction
What it is not

It is not:

- Access Control

Introduction
What it is not

It is not:

- Access Control
- Authentication, Authorization, Confidentiality, Integrity.

Introduction
What it is not

It is not:

- Access Control
- Authentication, Authorization, Confidentiality, Integrity.

This means that DLM will not ensure:

Introduction
What it is not

It is not:

- Access Control
- Authentication, Authorization, Confidentiality, Integrity.

This means that DLM will not ensure:

- secure communication between applications

Introduction
What it is not

It is not:

- Access Control
- Authentication, Authorization, Confidentiality, Integrity.

This means that DLM will not ensure:

- secure communication between applications
- limited access to data once released

Introduction
What it is

It is:

Introduction
What it is

It is:

- Information Flow Control

Introduction
What it is

It is:

- Information Flow Control
- Decentralized

Introduction
What it is

It is:

- Information Flow Control
- Decentralized

This means that DLM will help ensuring:

Introduction
What it is

It is:

- Information Flow Control
- Decentralized

This means that DLM will help ensuring:

- not releasing sensitive data

Introduction
What it is

It is:

- Information Flow Control
- Decentralized

This means that DLM will help ensuring:

- not releasing sensitive data
- not implicitly releasing sensitive data

Introduction
What it is

It is:

- Information Flow Control
- Decentralized

This means that DLM will help ensuring:

- not releasing sensitive data
- not implicitly releasing sensitive data
- not giving away hints of inner workings

DLM differs from previous solutions as it is:

Introduction
How it differs

DLM differs from previous solutions as it is:

- decentralized

Introduction
How it differs

DLM differs from previous solutions as it is:

- decentralized
- less restrictive of allowed computations

Introduction
How it differs

DLM differs from previous solutions as it is:

- decentralized
- less restrictive of allowed computations
- not completely disallowing inter-application communication

Introduction
How it differs

DLM differs from previous solutions as it is:

- decentralized
- less restrictive of allowed computations
- not completely disallowing inter-application communication
- meant to extend current programming languages with data flow annotations

DLM Basics

DLM provides both static and dynamic checking of data flow.

DLM Basics
Terminology

Principals represent users and other authoritative entities.

DLM Basics
Terminology

Principals represent users and other authoritative entities.

Values are entities computations can manipulate.

DLM Basics
Terminology

Principals represent users and other authoritative entities.

Values are entities computations can manipulate.

Slots are value-holders (e.g. variables, objects, and other storage locations).

DLM Basics
Terminology

Principals represent users and other authoritative entities.

Values are entities computations can manipulate.

Slots are value-holders (e.g. variables, objects, and other storage locations).

Input channels are read-only sources that allow information to enter the system.

DLM Basics
Terminology

Principals represent users and other authoritative entities.

Values are entities computations can manipulate.

Slots are value-holders (e.g. variables, objects, and other storage locations).

Input channels are read-only sources that allow information to enter the system.

Output channels are information sinks that transmit information outside the system.

Department of Computer Science
Aalborg University
Denmark

DLM Basics
Terminology

Principals represent users and other authoritative entities.

Values are entities computations can manipulate.

Slots are value-holders (e.g. variables, objects, and other storage locations).

Input channels are read-only sources that allow information to enter the system.

Output channels are information sinks that transmit information outside the system.

Labels are attached to values, slots or channels (more to follow).

DLM Basics
Labels

Department of Computer Science
Aalborg University
Denmark

DLM Basics
Operations

Department of Computer Science
Aalborg University
Denmark

Example

Department of Computer Science
Aalborg University
Denmark

Example
Code Example

Department of Computer Science
Aalborg University
Denmark

Advanced

Conclusion

Department of Computer Science
Aalborg University
Denmark

Conclusion
Future Works

Questions?

AALBORG UNIVERSITY
DENMARK