A Decentralized Model for Information Flow Control Andrew C. Myers and Barbara Liskov, 1997

September 23, 2015

Mikael Elkiær Christensen michrill@student.aau.dk

Department of Computer Science
Aalborg University
Denmark





Mikael Elkiær Christensen

Introduction

What it is not What it is How it differs

DLM Basics

Terminolog

Laueis Kan Dalasia

Example

Code Examp

Advanced

Future Works

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

Department of Computer Science Aalborg University Departs



> Mikael Elkiær Christensen

What it is not

What it is How it differ

Terminology

Labels

Key Principle

Example

Code Example

Advanced

Future Works

It is not:

Department of Computer Science Aalborg University

13



Mikael Elkiær Christensen

Introduction What it is not

What it is

DLM Basics

Torminalas

Labels

Key Princip

Example

Code Examp

Advanced

Future Works

It is not:

► Access Control



> Mikael Elkiær Christensen

Introduction What it is not

What it is

DLM Basics

Labels

Key Principl

Example

Advanced

Future Works

It is not:

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

Department of Computer Science Aalborg University Departs

13



Mikael Elkiær Christensen

Introduction What it is not

What it is

How it differ

DLM Basics

Labels

Key Princip

Example

GUUB EXAII

Advanced

Euturo Works

It is not:

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

This means that DLM will not ensure:



Mikael Elkiær Christensen

Introduction What it is not

What it is

DLM Basic

Terminology

Key Princin

Example

Advanced

....

It is not:

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

This means that DLM will not ensure:

secure communication between applications



Mikael Elkiær Christensen

Introduction What it is not

What it is

DLM Basi

Terminolog

Key Princin

Example

Code Examp

Advanced

Conclusion Euture Works

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



Mikael Elkiær Christensen

Introduction

What it is no

What it is

.....

DLM Basic

Terminology

Labels

Example

Code Examp

Advanced

Euturo Works

Department of Computer Science Aalborg University Denmark

13

It is:



Mikael Elkiær Christensen

Introductio

What it is no What it is

How it differ

DLM Basics

Terminolog

Labelo

Example

Code Exampl

Advanced

Eutura Marka

It is:

► Information Flow Control



Mikael Elkiær Christensen

Introduction What it is n

What it is no What it is

How it diffe

DLM Basics

Terminology

Key Principl

Example

Advanced

. . .

It is:

- ► Information Flow Control
- Decentralized



Mikael Elkiær Christensen

Introductio

What it is no

What it is

How it diffe

DLM Basics

Lahole

Kev Principi

Example

Code Example

Advanced

Euture Works

lt is:

- ► Information Flow Control
- Decentralized

This means that DLM will help ensuring:



Mikael Elkiær Christensen

Introductio

What it is no

What it is

I IOW It GIIIO

DLM Basics

Terminolog

Labeis

Key Princip

Example

Advanced

Eutura Works

lt is:

- ► Information Flow Control
- Decentralized

This means that DLM will help ensuring:

▶ not releasing sensitive data



Mikael Elkiær Christensen

Introduction

What it is no

What it is How it diffe

DLM Basic

Terminology

Labels

Key Principl

Example Code Examp

. .

Advanced

Future Works

It is:

- ► Information Flow Control
- Decentralized

This means that DLM will help ensuring:

- not releasing sensitive data
- not implicitly releasing sensitive data



Mikael Elkiær Christensen

Introduction

What it is no

What it is

HOW IT GITTE

DLM Basic

Terminolog

Koy Princin

Example

Code Examp

Advanced

Conclusion

lt 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



Mikael Elkiær Christensen

Introduction What it is no

What it is How it differs

How it differs

DLM Basics

Terminology Labels

Evennele

Code Example

Advanced

Future Works



Mikael Elkiær Christensen

Introductio What it is no

What it is How it differs

DLM Basics

Termino

Labels

_ .

Code Examp

Advanced

Future Works

DLM differs from previous solutions as it is:

decentralized



Mikael Elkiær Christensen

Introduction What it is no

What it is

How it differs

DLM Basics

Terminol

Labels

Rey Filicip

Example Code Examp

Code Examp

Advanced

Future Works

- decentralized
- less restrictive of allowed computations



Mikael Elkiær Christensen

How it differs

Advanced

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



Mikael Elkiær Christensen

How it differs

Advanced

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



Mikael Elkiær Christensen

Introductio

What it is

How it diffe

DLM Basics

Terminology

Labels

Example

Code Examp

Advanced

Future Works

DLM provides both static and dynamic checking of data flow.

Department of Computer Science Aalborg University



Mikael Elkiær Christensen

Introduction What it is not What it is

DLM Basics

Terminology

Key Principle

Example
Code Example

Advanced

Future Works

Principals represent users and other authoritative entities (e.g. groups or roles).



Mikael Elkiær Christensen

Introduction
What it is no

What it is How it differ

DLM Basics

Terminology

Key Principle

Example Code Example

Advanced

Euturo Worke

Principals represent users and other authoritative entities (e.g. groups or roles).

Values are entities computations can manipulate.



Mikael Elkiær Christensen

Introduction
What it is no

What it is How it differ

DLM Basics

Terminology

Key Principl

Example Code Examp

Advanced

Future Works

Principals represent users and other authoritative entities (e.g. groups or roles).

Values are entities computations can manipulate.

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



Mikael Elkiær Christensen

Introduction
What it is no

What it is How it differs

DLM Basics Terminology

Labels

Key Principl

Example Code Examp

Code Examp

Advanced

Future Works

Principals represent users and other authoritative entities (e.g. groups or roles).

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.



Mikael Elkiær Christensen

What it is no What it is

DLM Basics

Terminology

Labels Kou Bringini

Example

Code Example

Advanced

Future Works

Principals represent users and other authoritative entities (e.g. groups or roles).

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.



Mikael Elkiær Christensen

What it is no
What it is

How it differs

DLM Basics Terminology

Labels

Example Code Examp

Advanced

Conclusion
Future Works

Principals represent users and other authoritative entities (e.g. groups or roles).

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).

13



Mikael Elkiær Christensen

What it is not What it is

DLM Basics Terminology

Labels

Key Principl

Example Code Example

Advanced

Future Works

A label ${\bf L}$ is a set of owners, where each owner denotes its readers, e.g.:

$$\{o_1: r_1, r_2; o_2: r_2, r_3\}$$

where o_1, o_2, r_1, r_2, r_3 are principals.



Mikael Elkiær Christensen

What it is not What it is

Terminol

Lahels

Key Principl

Example Code Example

Advanced

Conclusion Future Works A label **L** is a set of owners, where each owner denotes its readers, e.g.:

$$\{o_1: r_1, r_2; o_2: r_2, r_3\}$$

where o_1, o_2, r_1, r_2, r_3 are principals.

The effective reader set of a label is the intersection of every reader, for L it is $\{r_2\}$.



Mikael Elkiær Christensen

Introduction

What it is not What it is How it differs

DLM Basics

Terminolog

Key Principles

Example Code Example

Advanced

Future Works

► Labels are comparable:

▶ $L_1 \sqsubseteq L_2$ signifies that L_2 is at least as restrictive as L_1 .



Mikael Elkiær Christensen

Introductio

What it is not What it is How it differs

Lahale

Key Principles

Example

Advanced

ravanoca

► Labels are comparable:

- $\blacktriangleright \ L_1 \sqsubseteq L_2 \ \text{signifies that} \ L_2 \ \text{is at least as restrictive as} \ L_1.$
- ► Labels can be joined:
 - L₁ ⊔ L₂ results in a join of owners and intersection of readers.



Mikael Elkiær Christensen

Introductio

What it is How it differs

DLM Basic

Labels

Key Principles

Code Example

Advanced

Future Works

- ► Labels are comparable:
 - ▶ $L_1 \sqsubseteq L_2$ signifies that L_2 is at least as restrictive as L_1 .
- ▶ Labels can be joined:
 - L₁ ⊔ L₂ results in a join of owners and intersection of readers.
- Principals can act for other principals.



Mikael Elkiær Christensen

Introductio

What it is How it differs

Terminolo

Kev Principles

Example

Code Example

Advanced

Future Works

- ► Labels are comparable:
 - ▶ $L_1 \sqsubseteq L_2$ signifies that L_2 is at least as restrictive as L_1 .
- ▶ Labels can be joined:
 - L₁ ⊔ L₂ results in a join of owners and intersection of readers.
- Principals can act for other principals.
- Relabeling can be done, further restricting or declassifying.



Mikael Elkiær Christensen

What it is no

How it differ

DLM Basics

Labels

Example

Codo Even

Code Exampl

Advanced

Future Works

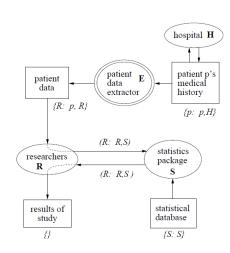


Figure 1: Medical Study Scenario



Mikael Elkiær Christensen

Introduction

What it is

DLM Basic

Terminolog

Labelo

.

Code Example

Advanced

Conclusion

```
check_password (db: array[pinfo{\bot}]{\bot},
                 user: string \{\bot\},
                 password: string{client: chkr})
   returns (ret: bool{client: chkr})
   % Return whether password is the password of user
   i: int {chkr: chkr} := 0
   match: bool {client: chkr;
                 chkr: chkr} := false
   while i < db.length() do
     if db[i].names = user &
       db[i].passwords = password then %
          match := true
                                         % {client: chkr;
     end
                                         % chkr: chkr}
     i := i + 1
   end
   ret := false
                                              % ⊥
   if_acts_for(check_password, chkr) then
     ret := declassify(match, {client: chkr}) % \( \pm$
   end
```

pinfo = record [names, passwords: string{chkr: chkr}]

Figure 6: Annotated password checker

end check_password



Advanced

Decentralized Label Model

Mikael Elkiær Christensen

DLM Basics

Advanced

- Label polymorphism
- Run-time labels (1b type)
- ► Protected types (protected[T])
- ▶ Inferred labels



Decentralized Label Model

Mikael Elkiær Christensen

Introduction
What it is no

How it differs

DLM Basics

Terminology Labels

Key Principle

Example

Code Exampl

Advanced

Conclusion

► Decentralized Label Model



Decentralized Label Model

Mikael Elkiær Christensen

DLM Basics

Advanced

Conclusion

Decentralized Label Model

► Control of information flow

Department of Computer Science Aalborg University Denmark

13



Decentralized Label Model

Mikael Elkiær Christensen

Introduction What it is no

What it is How it differs

DLM Basics Terminology

Terminolog

Key Princip

Example

Code Exampl

Advanced

Conclusion Future Works Decentralized Label Model

- Control of information flow
- Static and dynamic label checking

Department of Computer Science Aalborg University Denmark



Decentralized Label Model

Mikael Elkiær Christensen

Introductio What it is no

What it is How it differs

DLM Basics Terminology

Labels Vou Bringin

Example

Code Examp

Advanced

Conclusion

Future Works

- Decentralized Label Model
- Control of information flow
- Static and dynamic label checking
- Possible to extend existing programming languages

Department of Computer Science Aalborg University Departs



Mikael Elkiær Christensen

Introduction What it is no

What it is How it differs

DLM Basics

Terminology Labels Key Principles

Example

Code Examp

Advanced

Conclusion Future Works ► Actual implementation (JIF – dead)



Mikael Elkiær Christensen

Introductio What it is no What it is

DLM Basics

Labels

Kev Principle

Evample

Code Examp

Advanced

Future Works

► Actual implementation (JIF – dead)

Support for user-defined data abstractions



Mikael Elkiær Christensen

What it is not What it is

DLM Basics

Labels

Key Principle

Example

Code Examp

Advanced

Conclusion

► Actual implementation (JIF – dead)

- Support for user-defined data abstractions
- Formal proofs



Mikael Elkiær Christensen

What it is not What it is What it is

DLM Basics Terminology

Labels

Example

Code Examp

Advanced

Conclusion Future Works

- ▶ Actual implementation (JIF dead)
- Support for user-defined data abstractions
- Formal proofs
- ▶ Network systems



Mikael Elkiær Christensen

DLM Basics

Advanced

Future Works

▶ Actual implementation (JIF – dead)

- Support for user-defined data abstractions
- ▶ Formal proofs
- Network systems
- Threading

Questions?

