Safety and Security with AADL: Using lattices to model data flow

ERICK MARTINEZ

Safety and security with AADL: Using lattices to model data flow

Erick Martinez-Rosales, Dr. Eugene Vasserman

Department of Computer Science
College of Engineering
Kansas State University

Introduction

The Architecture Analysis and Design Language (AADL) is a model-based engineering language that is used today to design and build safety-critical systems. It is used to check safety features and faults within a system to make sure that the system is suitable for deployment. Unfortunately, security is often an afterthought in safety-critical design. In AADL, a language designed for safety modeling, security modeling is a non-trivial task.

The objective of this research is to create a program or system that can be used by safety engineers so that when they design systems both safety and security can be implemented equally.

A lattice structure is similar to a hierarchy, a form of organization depending on level or power. The difference is that lattices are multilateral structures as well, not only do they need specific level or power requirements but they also have to belong to a certain group or label in order to be comparable. This helps with reasoning about system guarantees related to performance, bandwidth, timing, etc.

We are attempting to build a lattice data type in AADL, making it possible to assign labels and properties to specific parts of a system so that information can move within the system with only well-defined interactions, i.e. data with label *A* will not interfere with data labeled *B*.

Method

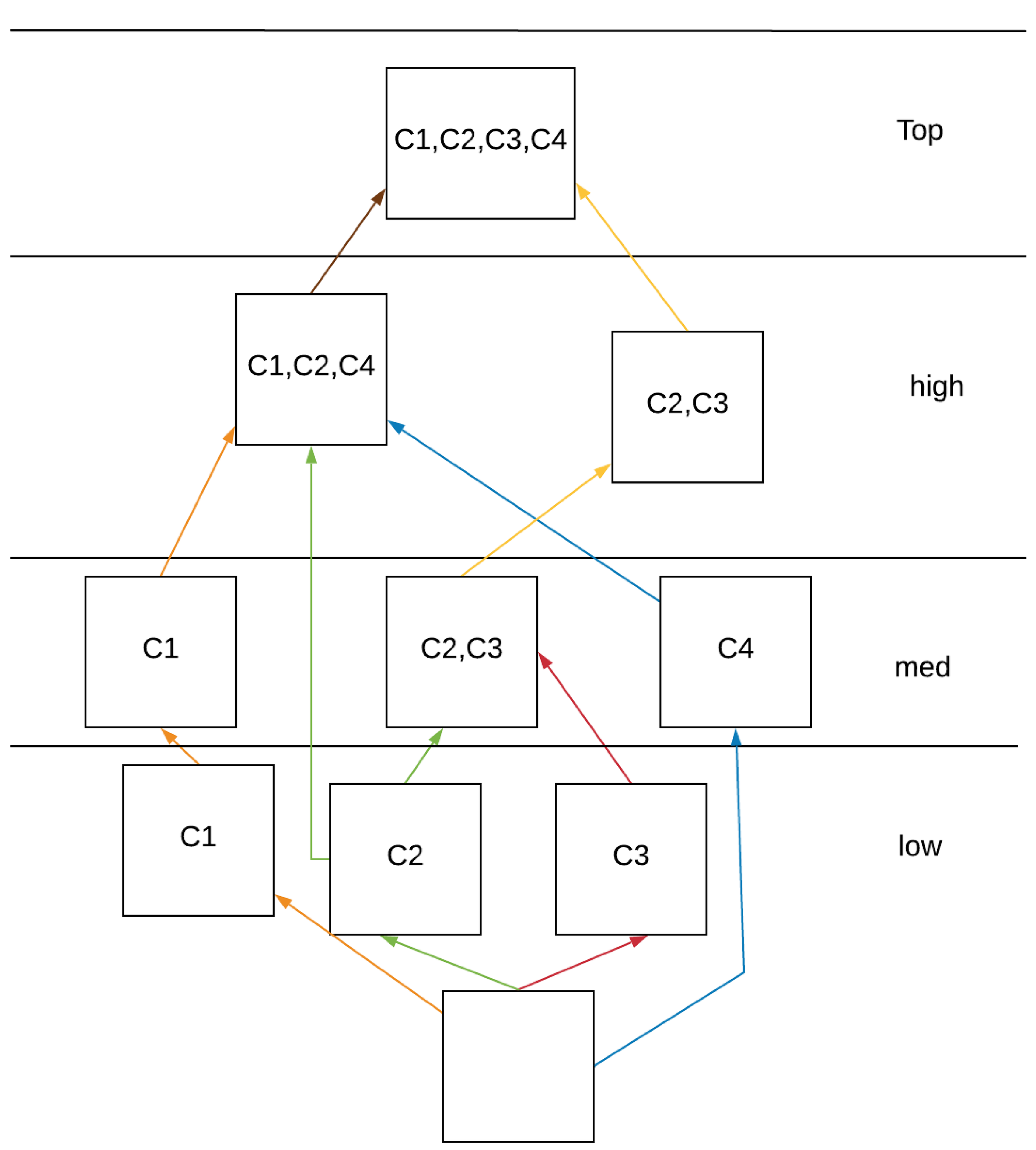
In this work, using AADL and the Error Modeling Annex version 2 (EMv2) we will model a system that uses the lattice structure of labels and properties in the lattice model below.

The levels low, med, high, and top are used to identify the level of importance of each node in the lattice.

The labels C1-C4 are used to identify the data classification type (or partition/compartment).

We will test the lattice-based data flow analysis to identify possible data flow and clearance or property violations in a modified GPS system.

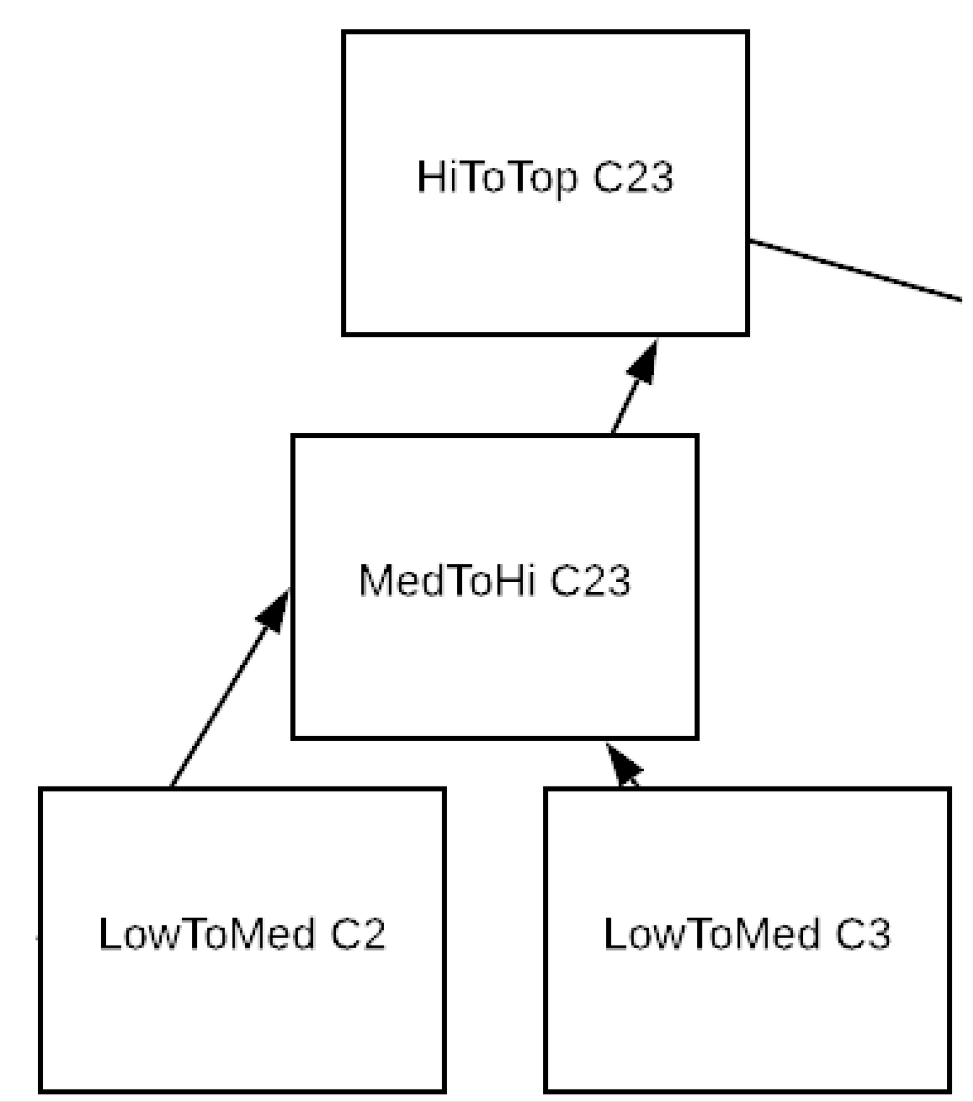
LATTICE DATA FLOW MODEL



Results

```
process implementation sendInfoToH
    subcomponents
        med: thread med23;
        Hi: thread High23;
    connections
        mtos: port med.class23Inf
end sendInfoToHi.i23;
```

```
em implementation integration.i
subcomponents
    cpu: processor Core;
    sendInfoT124: process sendInfoToTop
    sendInfoT23: process sendInfoToTop.
    sendInfoS1: process sendInfoToSec.i
    sendInfoS2: process sendInfoToSec.i
    sendInfoS23: process sendInfoToSec.
    sendInfoS4: process sendInfoToSec.i
    sendInfoM1: process sendInfoToMed.i
    sendInfoM2: process sendInfoToMed.i
```



we can use threads and processes to navigate the lattice structure to follow the flow of the information.

Putting the partitions in this manner will allow us to identify if the partition the information is going to belongs to the same category.

Conclusions

We found a way to construct a lattice data type from a combination of other AADL-native data types and structures. Using lattices, we can study a system's behavior in terms of data flow and management under normal as well as adversarial/malicious operating conditions.

This project will be integrated as part of a larger security-focused effort to augment the reasoning within AADL, to create a platform that will allow safety engineers to design a safety-critical systems that will enforce both safety and security properties.

References

1. Anderson, Ross. Security Engineering: a Guide to Building Dependable Distributed Systems, Second Edition. Wiley, 2008.

2. Sandhu, Ravi S. *Lattice Based Access Control Models.* IEEE Computer, Volume 26, Number 11, Pages 9-19, November 1993.

3. Sandhu, Ravi S. *Role Hierarchy and Constraints for Lattice-Based Access Controls.* European Symposium on Research in Computer Security, September. 1996.

Kansas LSAMP
Kansas Louis Stokes Alliance for Minority Participation