

Automated Assurance of Security Policy Enforcement

Julien Delange



Copyright 2016 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

DM-0004074



Software Security is Putting Our Lives at Risk!

Jeep hack (July 2015)

- Remote control of a car, 1.4M recalls
- <https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/>

Airplanes, ATC control system (2012)

- No encryption or authentication between planes and ground stations
- Sniffing packets or injecting forged data (what is the impact with the autopilot?)
- <https://www.youtube.com/watch?v=CXv1j3GbgLk>

Medical devices

- Remote access to infusion pumps without authentication
- Hundreds of devices are **insecure by design**
- <https://securityintelligence.com/news/do-no-harm-medical-device-vulnerabilities-put-patients-at-risk/>



Towards an AADL Security Modeling Guide

Security is being a lot of attention recently – for a reason!

- **Fact:** Our world is becoming software intensive. It is everywhere.
- **Consequence:** It introduces vulnerabilities.
- **Impact:** New threats that needs to be handled

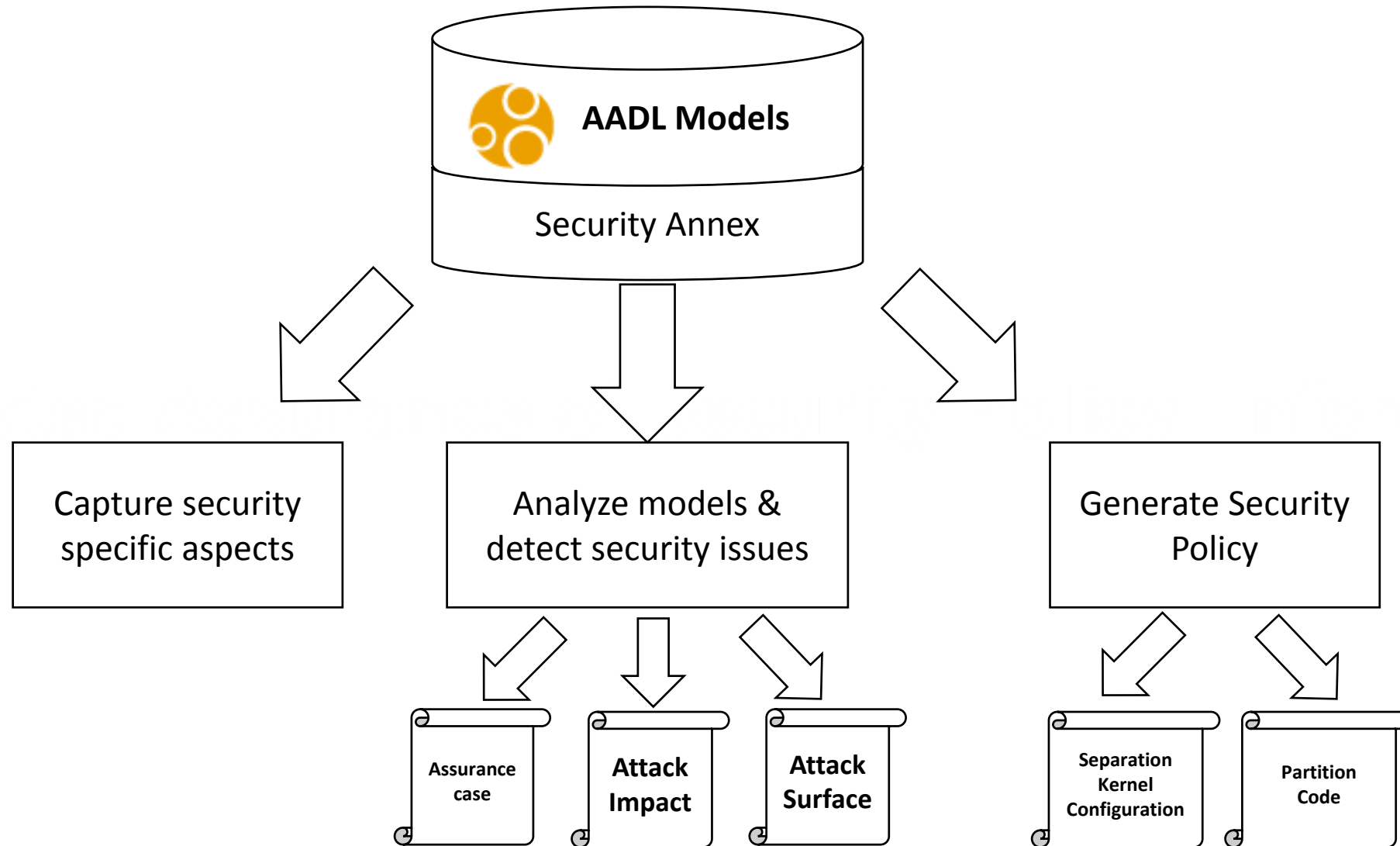
Security is not only a matter of code

- Vulnerabilities are related to software architecture
- E.g.: data protection, isolation of resources

Extend AADL with security design rules and analysis methods

- Detect common vulnerabilities in existing AADL models
- Automatically generate reports: attack surface, attack impact, etc.
- Analyze their **impact and propagation** through the architecture

Objective of an AADL Modeling Annex





The AADL Modeling Annex Document

AADL standardization document

- Modeling rules to specify architectural security aspects
- New AADL property set, compatible with existing models
- Integration with the Error-Model Annex (AADL EMV2)

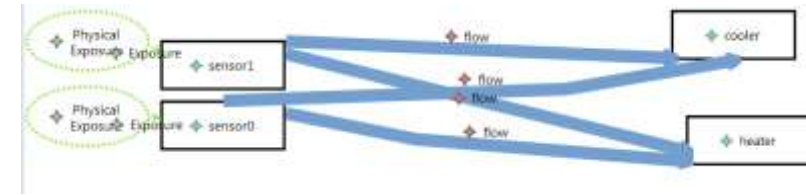
Capture security information in AADL

- Physical exposure
- Logical and physical isolation
- Encryption mechanism (algorithms, keys)
- Authentication (IP, user/pass)
- Security verification and trust

AADL-Based Security Analysis Tools

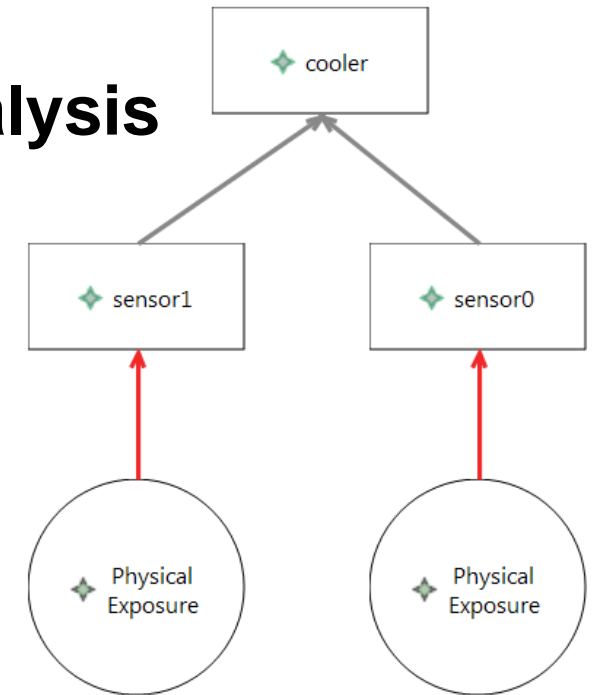
Detect vulnerabilities and their impact through the architecture

- Combine AADL semantics (e.g. connections, bindings) with security extensions
- Provide ability to analyze how architecture layers (e.g. functions, platform) impact system security



Attack Impact – similar to Failure Mode and Effects Analysis

- Visualize vulnerabilities and their propagations
- Implemented as a directed graph



Attack Tree – similar to Fault-Tree for security

- Visualize contributors (components, vulnerabilities) to a successful attack
- Implemented as a tree, vulnerabilities being leaves



Making an Impact

Integration of security annex into AADL standard

- Standardize modeling rules into the SAE AS2-C AADL standard

Demonstrate design and analysis capabilities

- Reproduce Jeep hack and analyze other automotive systems
- Application for the avionics domains (SAVI project)
- Collaboration with the Open-Source MILS working group

Publishing Security Analysis Tools

- Open Source license
- Available on SEI GitHub account

Ongoing Work

Code Generation of secure systems from AADL models

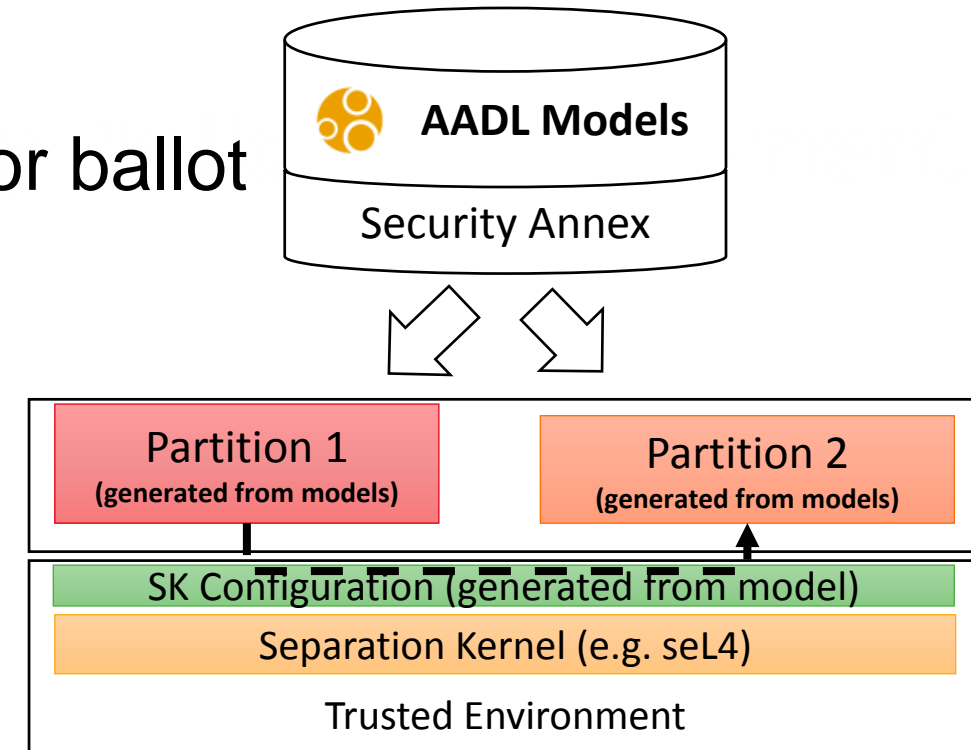
- Automatically generate security policies from AADL models
- Demonstrate using an autonomous system: create & deploy an autonomous drone on top of the formally verified seL4 kernel

Standardization activities

- Finalize the AADL annex and submit it for ballot

Dissemination activities

- SEI webinar on secure architecture design and analysis
- Academics papers and tutorials



Conclusion

Last year focus: Security Design Rules and Analysis Methods

- Integration into the AADL standard
- Security analysis (Attack Impact and Attack Tree) tools released
- Application to safety-critical systems (automotive and avionics)

Next year focus: Code Generation and Dissemination

- Auto-production of secure code from validated models
- Use the formally verified seL4 kernel (HACMS project)
- Application to an autonomous vehicle

Thank You !



AADL Tutorial at EsWeek (October 2016)

- <https://github.com/osate/examples/tree/master/esweek2016-tutorial>

AADL Security Analysis Tools on GitHub

- Open-source (BSD), integrated with OSATE (<http://osate.org>)
- <https://github.com/cmu-sei/AASPE>

SEI Blog: Improve System and Software Security with AADL

- https://insights.sei.cmu.edu/sei_blog/2016/02/improving-system-and-software-security-with-aadl.html