A Derivation Framework for Dependent Security Label Inference

Peixuan Li and Danfeng Zhang
Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA, 16801, United States

Core Constraint Language

1. \( a, c, d : P \); \( b : S \);
2. \( x := a + z \);
3. \( y := k ; / k : (d > 0)? S : P \);
4. \( d > 0 \rightarrow S \subseteq \alpha_y \);
5. \( d < 0 \rightarrow a_y \subseteq \alpha_x \);
6. \( c := x \);

(a) Program

(b) Core Constraints

Derivation Framework

Constraints (satisfiable)

- True \( \rightarrow P \subseteq \alpha_x \land \alpha_x \subseteq P \)
- \( d > 0 \rightarrow S \subseteq \alpha_y \)
- \( \lnot (d > 0) \rightarrow P \subseteq \alpha_y \)
- \( d < 0 \rightarrow a_y \subseteq \alpha_x \)

\[ \alpha_y : (d > 0)? S : P \]

\[ \alpha_x : P \]

Ideal Form:

\[ d > 0 \rightarrow P \subseteq \alpha_x \land \alpha_x \subseteq P \land S \subseteq \alpha_y \]

\[ d < 0 \rightarrow P \subseteq \alpha_x \land \alpha_x \subseteq P \land P \subseteq \alpha_y \land a_y \subseteq \alpha_x \]

\[ \alpha_y : (d > 0)? S : P \]

- No overlapping predicates
- Construct a global solution by merging local solutions

Sound Derivation

- Sound derivation must cover all the predicates.
- All overlapping constraints are projected.

Equivalent Derivation

- Sound Derivation = Complete Derivation.

Refinement

Strongest Complete Derivation:

\[ C_{\text{complete}} = \Lambda(C_{\text{original}}) \land \forall (P_{\text{complete}} \Rightarrow P_{\text{original}}) \]


due to \( \alpha_x \subseteq P \land \alpha_x \subseteq P \)


Inference Algorithms

Arbitrary Constraints

- \( P_1 \rightarrow C_1 \)
- \( P_2 \rightarrow C_2 \)
- \( P_3 \rightarrow C_3 \)

Hybrid

- Early Accept
- Early Reject

One-Shot

- \( P_1 \rightarrow C_{\text{sound}} \)
- \( P_2 \rightarrow C_{\text{complete}} \)

Evaluations

- Accepted with Solution Found
- Rejected with Counterexample

This work was partially supported by NSF grants CCF-1566411, CNS-1702760 and CNS-1816282.