

CT-T: Resource-guided Implementation of Secure Embedded Software

Rajeev Alur, Pavol Černý, Andre Scedrov, Steve Zdancewic
University of Pennsylvania



Analyzing Confidentiality for Programs

For programs such as J2ME midlets for mobile devices, a central correctness requirement concerns confidentiality of data that the user wants to keep secret. We formalize this requirement as conditional confidentiality, and develop an automated verification technique for establishing it.

Confidentiality: A property F is confidential if for every execution r of the program, there exists an execution r' such that 1) r and r' appear equivalent to the observer, and 2) they disagree on the truth of F

Application area: Midlets Midlets are small Java programs designed for enhancing features of mobile devices. When downloading a third-party midlet, the user would like some guarantee that the midlet does not leak confidential data.



```
public Vector<String> phoneBook;
public String number;
public int selected;

public void sendEvent() {
    phoneBook = getPhoneBook();
    selected = chooseReceiver();

    number = phoneBook.elementAt(selected);
    if ((number == null) | (number == "")) {
        //output error
    } else {
        String message = inputMessage();
        sendMessage(number, message);
    }
}
```



The phone number from the phone book should not leak

Approach and Impact

New approach

- Framework for specifying and verifying secrecy.
- Finite state systems: model checking algorithms for logics for temporal logics for confidentiality.
- For programs, a method based on computing both over- and under-approximation. (We show that both are necessary).

Research Impact

- Algorithmic verification of secrecy and time properties for finite state systems and protocols. Example: "Agent A does not reveal x (a secret) until agent B reveal y (a password)."
- Program analyzer for a fragment of Java.
- Verifier can operate in two modes: small embedded tool running on the device, or an offline tool that certifies midlets.

•Midlets are third-party programs, often downloaded from the internet. They have legitimate reasons to **access data on the mobile device** (such as the list of contacts or a phone book), and a legitimate reason to **send messages**, be it text messages (SMS), emails or http requests. Therefore, there is a question of how to ensure that a given (possibly malicious) midlet does not leak confidential information. The problem cannot be solved by simply prohibiting input of sensitive data or general outputs without limiting the functionality of the midlet.

•Bounded programs are programs for which an upper bound on the number of iterations can be computed statically for each loop. For this type of programs, our method produces a logical **formula that characterizes the confidentiality requirement**, and we identify syntactic restrictions under which its validity can be decided by **existing decision procedures**.

•For programs that are not bounded, our method uses **invariants** (which can be user supplied or automatically discovered) for **over-approximations**, coupled with **under-approximations via loop unrolling**.

•**Logics for specifying secrecy:** CTL \approx , $\mu\approx$ -calculus - expressive logics for safety, liveness and information flow properties. Richer models that capture enough information for information flow properties: **trees with path equivalences**. Model-checking algorithms: for fragments of CTL \approx ($\mu\approx$ -calculus) that are expressive enough to capture the properties of interest, the model-checking problem is **PSPACE-complete** (EXPTIME-complete).

•**Secrecy preserving refinement:** is a strengthening of the classical trace-based refinement so that the implementation leaks a secret only if the specification also leaks it. We developed a **simulation based proof technique** for secrecy-preserving refinement. This result shows that existing refinement checkers can be used to check preservation of secrecy

[ACZ06] R. Alur, P. Černý, S. Zdancewic: Preserving Secrecy Under Refinement, ICALP 2006

[ACC07] R. Alur, P. Černý, S. Chaudhuri: Model Checking on Trees with Path Equivalences, TACAS 2007

[CA08] P. Černý, R. Alur: Analyzing Confidentiality for Software, submitted