A Self-Protecting Agents Based Model for High-Performance Mobile-Cloud Computing

Pelin Angin, Bharat Bhargava, Rohit Ranchal

PII: S0167-4048(18)30394-8
DOI: 10.1016/j.cose.2018.04.011
Reference: COSE 1333

To appear in: *Computers & Security*

Received date: 1 October 2017
Revised date: 14 April 2018
Accepted date: 16 April 2018

A Self-Protecting Agents Based Model for High-Performance Mobile-Cloud Computing

Pelin Angin, Bharat Bhargava and Rohit Ranchal

Abstract—Mobile-cloud computing (MCC) allows devices with resource and battery limitations to achieve computation-intensive tasks in real-time. While this new paradigm of computing seems beneficial for real-time mobile computing, existing MCC models mainly rely on keeping full clones of program code at remote sites and unstandardized/uninteroperable environments, hampering wider adoption. Moreover, the security risks arising from offloading data and code to an untrusted platform and the computational overhead introduced by complex security mechanisms stand as deterrents for adoption of MCC at large. In this paper, we present a context-dependent computation-offloading model for MCC, which is based on application segments packed into autonomous agents. This approach only requires isolated execution containers in the cloud to provide a runtime environment for the agents, and minimal involvement of the mobile platform during the computation process. The agents in the proposed model are able to protect themselves from tampering using integrity-checkpointing and an authenticated encryption-based communication mechanism. Experiments with two mobile applications demonstrate the effectiveness of the approach for high-performance, secure MCC.

1 Introduction

Mobile computing devices have replaced desktops and mainframes for daily computing needs for the past decade. Despite the everyday advances in mobile computing technology, size restrictions impose limitations on the processing power and battery life of these devices, which limits their capabilities for real-time, computing-intensive applications such as image processing. Cloud computing offers the ability to fill the gap between the resource needs of mobile devices and availability of those resources, through the concept of *mobile-cloud computing* (MCC), which partitions mobile applications between mobile and cloud platforms for execution, by dynamically offloading parts of mobile computation to cloud hosts.

Achieving high performance with mobile-cloud computing requires optimal partitioning of the mobile application components between the mobile and cloud platforms based on dynamic runtime conditions. Recent work on this problem has resulted in frameworks with various partitioning and optimization techniques. However, most of these frameworks impose strict requirements on the cloud side, such as a full clone of the application code or special application management software, hindering wide applicability in public clouds.

The other major obstacle for wider adoption of MCC is the security risks associated with sending sensitive data and code to an untrusted platform, including but not limited to the following:

- Lack of control on resources and multi-tenancy of different users' applications on the same physical machine make cloud platforms vulnerable to attacks [1].
- In addition to privacy issues, programs running in the cloud are prone to tampering with code, data, execution flow and communication, as well as masquerading attacks [2].
- Mobile code can navigate through multiple platforms before returning to the origin, giving rise to the end-to-end security problem, which involves decreasing control with every further hop in the chain of platforms [3].

In order to provide complete security, the application should ensure all communication with/execution on the cloud platforms are trusted. Achieving ultimate flexibility and performance in mobile-cloud computing is contingent upon the availability of a secure computing framework capable of dynamic decision making with regards to the execution location of different program partitions. The performance requirements of real-time MCC call for a generalized computation offloading model that requires minimal involvement of the mobile platform for monitoring of offloaded computation, where all dynamic decision-making processes and integrity checks are achieved with lightweight components. The main challenges for achieving dynamic tamper resistance in MCC are the following:

- The tamper detection mechanism should involve minimal modification to the software structure, and be transparent to the programmer.
- Appropriate measures should be taken in case of tamper detection in a way that is transparent to the software.
- The runtime performance overhead of tamper detection and response should be minimal.
- Communication costs for detection and reporting of tamper should be minimal.
- The size of the program code/data should not be increased significantly due to the protection mechanism.

In this paper we present a mobile-cloud computation model based on autonomous agent-based application mod-

- *Pelin Angin is with the Department of Computer Engineering, Middle East Technical University, Ankara, Turkey. E-mail: pangin@ceng.metu.edu.tr.*
- *Bharat Bhargava is with the Department of Computer Science, Purdue University, West Lafayette, IN, USA. E-mail: bb@cs.purdue.edu.*
- *Rohit Ranchal is with IBM Watson Health Cloud, Cambridge, MA, USA. E-mail: ranchal@us.ibm.com.*