



---

## GEC9 Demo : SONOMA: Secure Opportunistic Network-Wide Offload for Mobile Applications

Participants: Aaron Gember, Aditya Akella, *University of Wisconsin - Madison*

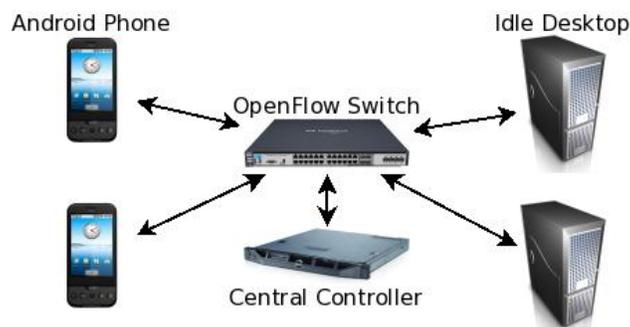
URL: <http://cs.wisc.edu/~agember/research/offloading>

### Abstract

Smartphones are being increasingly employed to run complex new applications, placing high demands on these resource-constrained devices. Offloading allows these resource-intensive mobile applications to run more frequently without performance or energy costs, by taking advantage of external computing resources. As an idea, Offloading has been around for nearly a decade and several systems have been proposed. However, we see two main reasons for a lack of wide adoption: (i) a majority of proposals lack mechanisms to ensure private data remains secure, and (ii) existing systems focus on *what* to offload and pay little attention to *where* to offload. We demonstrate SONOMA, an offloading system designed to address these limitations.

SONOMA's goal is to *opportunistically* leverage available computational resources and offer both *security guarantees* and *performance/energy improvements* to smartphone users. Our system shares the common ideas of automated application partitioning and server-based execution implemented in other systems, but expands on prior works by augmenting the decision process of *where* to execute offloaded applications. A logically central controller considers: (a) the resource needs of multiple mobile devices; (b) the availability of multiple types of secondary resources (idle desktops, local data centers, and remote clouds); and, (c) administrator specified security policies for applications and devices. Offloading destinations are carefully selected and communications are encrypted when appropriate, to maintain security and leverage available resources.

Our demonstration uses a set of emulated Android phones, idle desktops, and an OpenFlow switch to demonstrate the offloading capabilities of SONOMA. Compute intensive applications are launched on the emulated Android phones and request offloading resources when an offload point is reached. The controller is responsible for pairing available resources with offloading requests. Execution state is transferred from a phone to a desktop for offloaded execution, and results are returned back to the phone when the execution is complete.



**UW - Madison OpenFlow Network**

