Sensibility Testbed: an Internet-wide Cloud Platform for Programmable Exploration of Mobile Devices

Yanyan Zhuang[†‡], Leonard Law[†], Albert Rafetseder[*], Lai Wang[†], Ivan Beschastnikh[‡], Justin Cappos[†]
[†]NYU Poly, [‡]University of British Columbia, [*]University of Vienna

***Abstract*—Smartphones contain sensors that provide information about the user of the device. Studying smartphones, therefore, provides an invaluable window into the behavioral patterns of users. In this work, we describe an open infrastructure that provides research communities with principled access to mobile devices and their sensors. Our platform, the Sensibility Testbed, is a free, community-driven platform for mobile devices. It provides secure data access to user-owned mobile devices while preserving user privacy. The unified programmable interface to sensors on heterogeneous devices in the Sensibility Testbed enables research scientists to design and deploy experiments that run across large numbers of devices, for example to measure the coverage and performance of cellular and WiFi networks. The Sensibility Testbed makes the sensing capabilities of smartphones more accessible to a broad range of researchers.**

I. Introduction

Thanks to embedded sensors and physical proximity to users, mobile devices produce valuable data for researchers and engineers across various scientific and engineering disciplines. For example, accelerometers on these devices can detect vibrations within the frequency and intensity range of seismic waves, and can assist distributed earthquake detection [6]. WiFi and cellular information, such as radio coverage and received signal strength, can be used by service providers to improve the design of their wireless infrastructure [11]. Sensors on mobile devices can objectively record information gathered from the end-user's perspective. Research scientists and engineers can harness this data to test hypotheses, improve network protocols, and design new systems. However, to conduct large-scale research, a deployment of many mobile testbed nodes is necessary to test research hypotheses, protocols and system designs, etc. Furthermore, security and privacy considerations have historically limited public access to personal sensor data. Existing research testbeds on mobile platforms provide partial solutions to these challenges [8], [9].

In this paper we introduce the Sensibility Testbed [2], a distributed platform that allows researchers to directly experiment with end-user sensor data. This testbed provides ubiquitous, secure data access while preserving user privacy. Through a programmable interface to sensors on devices in the wild, the testbed enables researchers to deploy services tailored to their needs. Once installed, the Sensibility Testbed provides a virtual programming environment to researchers wanting to program an experiment, without requiring them to recruit device donors to provide dedicated resources for the experiment. By allowing researchers to securely experiment with data from a variety of mobile sensors the Sensibility Testbed has the potential to stimulate new research opportunities in a wide range of disciplines.

II. Challenges and Contributions

A. Challenges

A testbed that allows researchers to execute code across donated mobile devices faces three major challenges. First, the *mobile platform* composed of the sensing hardware and the mobile device poses a portability challenge — applications on these devices should be executable on as many OSes as possible. The second challenge is *secure data access* — sensors on smartphones pose a risk to device donors whose devices are exposed to potentially erroneous or malicious code written by testbed users. The third challenge is *sensor data privacy* — testbed users should not gain access to information that donors want to be kept private. The Sensibility Testbed has solved the first two challenges by using a secure and performance-isolated sandbox. The last challenge is our ongoing effort.

B. Contributions

1) Mobile Platform: The Sensibility Testbed is based on Seattle [3], a community-driven, open-source cloud computing system. Our testbed enables new research capabilities by improving on the virtualization techniques of Seattle [4], and leveraging mobile devices. To handle mobile platforms, Sensibility Testbed uses a subset of the POSIX API constructed with the Restricted Python (Repy) sandbox [4], the core of Seattle. Repy is highly portable, and has been running on a variety of different operating systems over the past five years.

2) Secure Data Access: The long-term nature of many research experiments requires the use of minimal resources in a non-intrusive manner. This is also achieved by Repy. To securely interact with sensors on remote user devices, Repy uses a small amount of the storage, network, memory, and CPU resources of a device in a performance-isolated manner. Repy provides hard limits on resource consumption. As a result, programs are sandboxed and securely isolated from other programs on the same device. Currently, a Sensibility Testbed prototype has been released in the Android app store, and has been downloaded by over 500 users.

3) Sensor Data Privacy: While the GPS locations of a mobile device produce a rich history, e.g., for traffic prediction, such information invades user privacy. Currently, access control to smartphone resources is static and coarse-grained. For example, on most Android devices, access permissions are either granted or denied completely during app installation. As a result, apps may ask for more permissions than is necessary for their operation. Although the more recent Android versions have added the ability to revoke granted permissions, this capability is available to about 5% of all devices [1]. A number of other solutions have also been proposed to address

978-1-4799-3088-3/14/$31.00 ©2014 IEEE

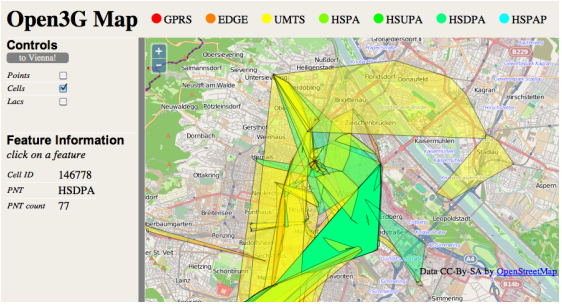

Figure 1: Live Cellular Map for Network Deployment.

this issue; however, they require modifications to the mobile platform.

We are working on a dynamic, fine-grained, and flexible access control mechanism, which incorporates privacy filters in user space [5]. We are developing a framework of reference monitors to enforce mandatory access control to sensor data in realtime. A reference monitor is a method or function that implements an access control policy for a set of resources and is specified in terms of what capabilities are allowed. Specifically, when a procedure call is made to the mobile OS to request sensor data, the call will be handled by a reference monitor. Based on a sensitivity score of the application, the data returned by the mobile OS can be passed through, filtered, or dropped by the reference monitor.

III. Deployment and Example Applications

A. Deployment

Mobile devices can benefit from each other's sensor data by making their sensor measurements accessible to a remote computing service. For instance, cellular information such as the radio coverage of a particular service provider and the cellular technology used, the received signal strength (RSS), can be used for service providers to improve the design of cellular infrastructure [11]. Figure 1 showcases such an application: it is composed of a web service that displays cellular access technology data points at their GPS locations collected by devices on the Sensibility Testbed. With such data openly accessible, providers can control base stations to adjust transmission power for effective radio coverage and to prepare for user handover in a timely manner and with higher accuracy.

Furthermore, the publicly observable cell tower and WiFi access point information, also available in the Sensibility Testbed, can provide an open geolocation service. As a result, mobile devices with weak GPS signals and laptops without GPS hardware can use the public cell tower information to quickly identify their approximate location.

B. Other Example Applications of Sensibility Testbed

1) Resource consumption and user perceived properties: Mobile platforms present greater challenges to application developers because of the constraints on energy, CPU and memory. Consumption of these resources also has a profound impact on application responsiveness, thermal heat, data cost, etc. By constantly gathering and monitoring resource usage with low-level interfaces, application developers can interpose between resources and applications, influencing the scheduling and control mechanisms of the mobile system [10]. Applications can automatically adapt to their programming environment by limiting resource consumption and managing processes dynamically.

2) Emergency Response: With location awareness mobile devices can achieve high transfer rates of sensor data with minimal communication latency in emergency situations. Examples include real-time traffic updates via remote process communication, emergency message dissemination and disaster response, broadcasting audio/video clips of life-threatening hazard [12], etc.

3) Social network: Users' behavioral patterns can also be observed through mobile devices. As people follow their daily routines, mobile sensing applications can observe a diversity of user locations and other sensor data that reflects the users' offline affiliations and personal preferences. Groups of users with distinct behaviors can be identified through advanced algorithms, even when the collected data set is anonymized [7]. By leveraging a subset of sensor data such as WiFi access point connections and Bluetooth encounters history, offline communities can be identified, i.e., by identifying re-appearance of a set of users at the same location.

IV. Conclusion

The Sensibility Testbed is an open infrastructure that provides research communities with principled access to mobile devices and their sensors. It provides secure data access to user-owned mobile devices while preserving user privacy. By allowing researchers to securely experiment with data from a variety of mobile sensors the Sensibility Testbed has the potential to stimulate new research opportunities in a wide range of disciplines.

References

[1] Cyanogenmod updating to privacy guard 2.0 with new features, coming to cm 10.2. http://www.androidcentral.com/cyanogenmod-updating-privacy-guard-20-new-features-coming-cm102.

[2] The Sensibility Testbed. https://sensibilitytestbed.com/.

[3] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson. Seattle: a platform for educational cloud computing. In *SIGCSE*. ACM, 2009.

[4] J. Cappos, A. Dadgar, J. Rasley, J. Samuel, I. Beschastnikh, C. Barsan, A. Krishnamurthy, and T. Anderson. Retaining sandbox containment despite bugs in privileged memory-safe code. In *CCS*. ACM, 2010.

[5] J. Cappos, L. Wang, R. Weiss, Y. Yang, and Y. Zhuang. Blursense: Dynamic fine-grained access control for smartphone privacy. In *Sensors Applications Symposium (SAS), accepted*. IEEE, 2014.

[6] M. Faulkner, M. Olson, R. Chandy, J. Krause, K. M. Chandy, and A. Krause. The next big one: Detecting earthquakes and other rare events from community-based sensors. In *IPSN*. IEEE, 2011.

[7] W.-j. Hsu, D. Dutta, and A. Helmy. Mining behavioral groups in large wireless lans. In *MobiCom*. ACM, 2007.

[8] Mobile Territorial Lab (MTL). http://www.mobileterritoriallab.eu/.

[9] PhoneLab: Programmable Participatory Smartphone Testbed. http://www.phone-lab.org/.

[10] L. Ravindranath, J. Padhye, S. Agarwal, R. Mahajan, I. Obermiller, and S. Shayandeh. Appinsight: mobile app performance monitoring in the wild. In *OSDI*. USENIX Association, 2012.

[11] M. Siebert, M. Lott, M. Schinnenburg, and S. Göbbels. Hybrid information system. In *VTC 2004-Spring*. IEEE, 2004.

[12] Y. Zhuang, J. Pan, V. Viswanathan, and L. Cai. On the uplink mac performance of a drive-thru internet. *Vehicular Technology, IEEE Transactions on*, 61(4):1925–1935, 2012.