Virtual Smartphone over IP

Mitsutaka Itoh, Eric Y. Chen[†]*, and Tetsuya Kusumoto*

Abstract

We describe the Virtual Smartphone over IP system (IP: Internet protocol), which allows users to create virtual smartphone images in a mobile cloud and customize each image to meet different needs. Users can easily and freely tap into the power of a data center by installing desired mobile applications remotely in one of these images. Because the mobile applications are controlled remotely, they are not constrained by the processing-power, memory, and battery-life limits of a physical smartphone.

1. Introduction

The number of smartphone users and mobile application offerings is growing rapidly. Recent smartphones can offer personal-computer-like functionality and are sometimes used as substitutes for laptop computers. Moreover, with the advent of Android, the first open source and fully customizable smartphone operating system (OS), developers of smartphone applications now have unprecedented freedom to create killer applications that were unimaginable before. The downside of this trend, however, is that smartphones are expected to suffer increasing security risks from malware and data leakage in much the same way as personal computers today. The fact that smartphones still have limited processing power and battery life further complicates the problem since conventional countermeasures such as anti-virus programs can seriously hamper the usability of smartphones. We are developing a system that provides a cloud computing environment specifically designed to extend the capacity of smartphones. It is called Virtual Smartphone over IP (Internet protocol).

2. Overview of Smartphone over IP

Our system allows smartphone users to create virtual smartphone images in the mobile cloud and customize each image to meet different needs. Users of our system can selectively run their applications in these images as they would locally. Running applications remotely in the cloud has a number of advantages: mobile applications installed remotely in these images can easily tap into the power of a data center, so they are not constrained by the processing-power, memory, and battery-life limits of a physical smartphone; the system avoids untrusted applications accessing local data; more effective security solutions can be deployed; and new ways of using smartphones become possible.

The system consists of a server program, client program, and communication protocol. The server program resides in each virtual smartphone image while the client program is installed in a physical mobile device. The client programs enable a user to remotely interact with and control various mobile applications installed in the virtual smartphone. Using a VNC-based protocol (VNC: virtual network computing), this system transmits various events from the physical device to the virtual smartphone and sends graphical screen updates in the other direction.

We have successfully implemented this concept with a virtual Android image that runs on x86 platforms and can be remotely controlled from an ARM-based Android device (**Fig. 1**). The main reason behind our choice is the fact that the Android OS is available on both x86 and ARM platforms as open source software. This particular system implementation allows for tight integration between both physical and virtual smartphones. However, this conceptual model can be extended to work on hybrid platforms, such as allowing an i-mode user to access a virtual Android image, thus enabling the user to virtually execute Android applications on their i-mode phones.

† NTT Information Sharing Platform Laboratories
Musashino-shi, 180-8585 Japan

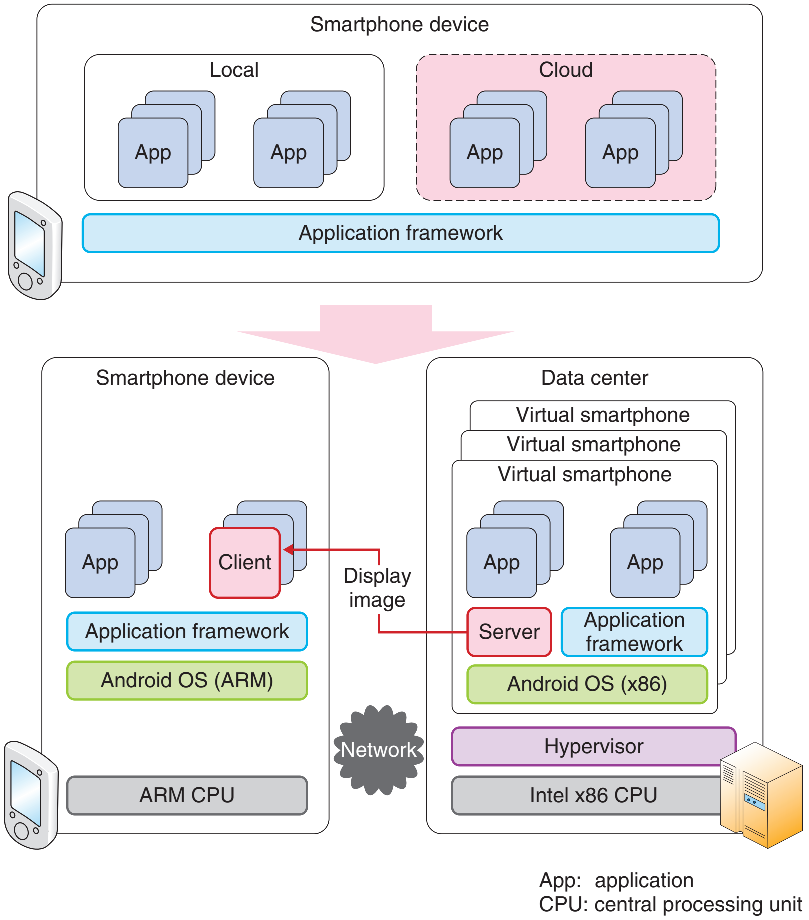


Fig. 1. Virtual Smartphone over IP.

3. Application scenarios

To create a new smartphone image in the cloud, the user can simply select from a number of preconfigured image templates to get up and running immediately. The following are examples of how users might utilize our system and what types of image templates we can provide to match different user needs.

3.1 Remote sandbox

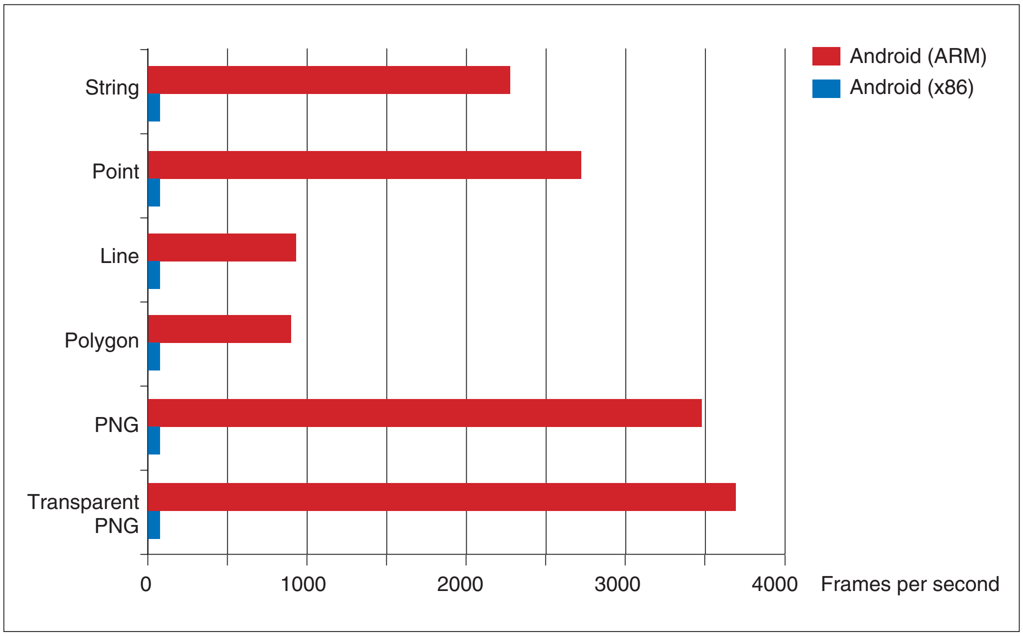
A virtual smartphone image can be used to execute unknown mobile applications from unverified third parties. This environment is conventionally called a *sandbox* because applications do not run natively on the physical device and can access only a tightly controlled set of remote resources visible from the virtual smartphone image. Network and data access is heavily restricted to minimize possible negative impacts of potentially harmful applications. A sandbox is particularly useful for Android users who would like to install the less-trusted applications obtained outside the official Android Market.

3.2 Data leakage prevention

Our system can also be used as a viable solution against data leakage if the data is stored in the data center and accessible only through one of the virtual smartphone images. Since only the graphic pixels of display images are transferred to the physical smartphone, corporate data is securely contained within the data center and never stored in the physical smartphone. This allows employees to work with the data remotely and securely without retaining it on their devices.

3.3 Performance boost

The fact that Android uses the same Java application framework on both x86 and ARM processors provides seamless application portability on these platforms. While most existing applications are



PNG: portable network graphics

Fig. 2. Comparative benchmark results.

designed with the ARM processor in mind, the performance of these applications can be boosted on an x86 by using the vast cloud resources. The comparative benchmark results in **Fig. 2** demonstrate the potential performance leverage that can be gained by executing the same mobile applications in the cloud. By executing computation-intensive applications remotely on virtual smartphones and transmitting only the graphical results to the physical smartphone, we free users from the processing-power, memory, and even battery-life limits of their smartphones.

3.4 Other possibilities

There are many other ways to use our system. It can be used to archive the less frequently used applications and free up storage space on physical smartphones. It can help users prevent their local device from accumulating unwanted residual files from trial applications. Android applications, for example, sometimes leave residual files even after they have been uninstalled by the user.

Developers may also take the advantage of the fact that virtual smartphones are consistently online and devise server-style mobile applications that would be difficult to deploy on physical smartphones.

4. Features

It is important that we allow remote applications on a virtual smartphone to be executed and controlled in the same way as local applications on a physical smartphone. To achieve this goal, we have implemented the following features in our system.

4.1 Sensor readings

Most modern smartphones are equipped with various sensors such as a GPS (global positioning system) device, accelerometer, orientation sensor, magnetic field sensor, and thermometer. While most conventional thin-client solutions support only the keyboard and mouse as primarily input devices, our system allows applications installed in a virtual smartphone to access a physical smartphone's sensor readings. The sensors appear as if they were mounted directly on the remote virtual smartphone. As more and more mobile applications take advantage of these devices, this will become an important feature for seamlessly running applications in a virtual remote environment.

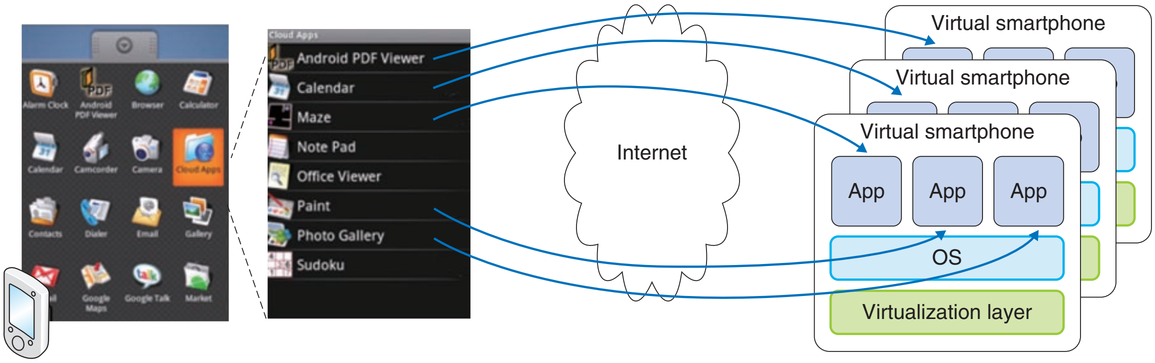


Fig. 3. Features.

4.2 GUI integration

Our system allows applications running in the cloud to appear like native applications on the physical device with graphical-user-interface (GUI) functions such as copy & paste between local and remote applications. It also features shortcuts to remote applications in the virtual smartphone that minimize the steps required for users to launch remote applications (**Fig. 3**). Furthermore, each shortcut can be bound to a different virtual smartphone, which allows users instant access to remote applications residing in multiple virtual smartphones in a single menu.

4.3 Relocation of applications

Another innovative feature is the ability to relocate applications and their associated data between a remote virtual smartphone and a local physical smartphone. This gives smartphone users a new degree of freedom in managing their mobile applications.

5. Future work

We have experimentally implemented a workable prototype of this system using Android devices (Dev-Phone) designed for developers, which give us more freedom than commercial Android devices. Some features described in this article require customization of Android. In the future, we intend to work closely with service providers and manufacturers to integrate these features.

Mitsutaka Itoh

Senior Research Engineer, Supervisor, Secure Communication Project, NTT Information Sharing Platform Laboratories.

He received the B.S. and M.S. degrees in mathematics from Waseda University, Tokyo, in 1984. He joined NTT Laboratories in 1984. Since then, he has been engaged in network security R&D. He created NTT-CERT and has contributed to improvements in information-system security. His current research interests are VoIP security, mobile cloud computing, web security, and countermeasures against botnets. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan and the Information Processing Society of Japan (IPSJ).

Eric Y. Chen

Chief Research Scientist, Secure Communication Project, NTT Information Sharing Platform Laboratories.

He received the B.Sc. and M.B.A. degrees from McGill University, Canada, in 1997 and 2000, respectively, and the Ph.D. degree from the University of Tokyo in 2005. He won the 2004 Nikkei BP Technology Award (in Japan). He has served on the Technical Advisory Board of the VoIP Security Alliance since 2006 and the Editorial Board of the International Journal of Multimedia Intelligence and Security since 2008. He also served as a Technical Program Co-Chair of IPTCOMM 2008 and SECURWARE 2008 and as a Technical Program Committee member of various conferences including Globecom, SAINT, and SIGCOMM LSAD Workshop. His current research interests are VoIP security and mobile cloud computing.

Tetsuya Kusumoto

Researcher, Secure Communication Project, NTT Information Sharing Platform Laboratories.

He received the B.E. and M.E. degrees from the Department of Electronic Information and Communication Engineering, Waseda University, Tokyo, in 2005 and 2007, respectively. Since joining NTT in 2007, he has been engaged in network security research. His current research interests are VoIP security, cloud computing, and mobile networks. He is a member of IEICE and IPSJ.