# Towards a User-Friendly Security-Enhancing BYOD Solution

Dennis Gessner, Joao Girao, Ghassan Karame, Wenting Li

## Abstract

Bring Your Own Device (BYOD) is attracting considerable attention nowadays. In BYOD scenarios, enterprises wish to integrate their employees' mobile devices in enterprise operations (e.g., reading emails, editing documents). This clearly raises serious security concerns since the mobile device in question is not under the control of the enterprise and is vulnerable to a wide range of security threats. In this article, we address this problem and propose a solution that enhances the security in the BYOD scenario without compromising the usability, and flexibility of the system. Our proposed solution does not require modifications to the underlying operating system of the device and enables IT officials to remotely manage their desired security policies.

Keywords
Bring Your Own Device (BYOD), Mobile Device Management (MDM), mobile security, data privacy

## 1. Introduction

Bring Your Own Device (BYOD) is gaining significant attention nowadays. BYOD denotes the problem of enterprise usage of devices, such as smartphones, tablets, PCs or notebooks, personally owned by employees. Such solutions enable companies to (i) reduce costs, since they no longer have to purchase enterprise-dedicated devices for their employees, and (ii) to offer the flexibility for employees to choose their own devices. While providing clear advantages, BYOD poses serious challenges with respect to securing data that originates/belongs to the enterprise and that is stored or displayed on the personal device of the employee. More specifically, given that the employee has full control of his/her device, the solution space to secure access to sensitive or security/critical information on the personal device of employees becomes rather limited.

We contrast this to the traditional case, where the device being used by the employee is provided by the company. In this case, IT officials of the enterprise can make the necessary modifications to the device kernel and Operating System (OS), to enforce that the required enterprise policies cannot be bypassed by employees. For instance, the enter-prise could certify a certain OS configuration and guarantee by means of a certified micro-kernel and a Trusted Platform Module (TPM) chip on the device, the correct execution of binaries.

In BYOD scenarios, these solutions cannot be deployed. This stems from the fact that given the device does not belong to the enterprise, the latter does not have any justification - and rightly so - in modifying the underlying kernel of the personal device of the employee.

This gives rise to the following security challenge: How can security policies be enforced on mobile devices with minimal modifications to the state/OS of the device?

This problem is further hardened when considering possibly untrusted device owners or employees, who may (purposefully or accidentally) alter their device settings to bypass security policies.

For instance, enterprises might be interested in ensuring that sensitive information is always stored in encrypted form on the device, and is only communicated to trusted pre-approved entities, in spite of potential threads caused by malware, malicious applications, etc.

In this article, we address this problem and we explore the solution space to secure mobile devices in the context of BYOD without incurring changes on the mobile device. First, we clarify the adversary model adopted in our context and we briefly discuss the challenges in securing application execution and information access in BYOD settings. We then present a solution that strengthens and isolates the execution of enterprise applications. Finally, we discuss a complementary policy management and policy enforcement layer that lev-erages application isolation and inter-faces with a user-friendly Mobile De-vice Management (MDM) layer.

## 2. Related Work

Nowadays, there are a number of products in the market that are designed to support BYOD. Some of these products are based on virtualization of the device [1) 2)] , other offer specialized applications for specific business processes such as emails or VPNs [3) 4)] . Furthermore, some products offer remote connections to the enterprise networks.

Most of these solutions, however, ex-hibit the disadvantage that they either require a modification of the underlying operating system, or they only consider passive adversaries (i.e., weak attacks). As mentioned earlier, modifying the OS of an employee device is not an appealing solution since it (i) prevents employees from installing updates on their devices and (ii) it requires the consent of the employee.

On the other hand, existing solutions that do not incur a modification are vulnerable to Man-In-The-Middle (MITM) attacks in which the adversary intercepts (e.g., by means of a malicious application) communication between the enterprise and the mobile device, acquires sensitive material (e.g., keys, sensitive emails), and can imper-sonate the mobile device, forge authentication credentials, etc.

In systems that support a Trusted Execution Environment (TEE) [8)] , a chain-of-trust is achieved using static measurement of system components at system-boot-time. The chain of trust starts from the Core root of Trust and includes measurements for the OS running on the device. Given these measurements, it is easy to decide whether a modification of the OS (or any other component) has occurred. These measurements could also be used for remote attestation purposes, where the device proves to a remote entity the correctness of it software.

Trusted Network Connect (TNC) equally adopts a similar approach [9)] that leverages TPM chips; the network switch in question can then grant or deny access to the network, based on the correctness of the measurements.

## 3. Adversarial Model

We start by clarifying the underlying adversarial model particular to our BYOD settings. Here, we assume that the adversary is interested in (i) diverting the execution flow of applications (e.g., to inject, modify code or execute malicious code), (ii) acquiring secrets (e.g., cryptographic keys), or sensitive information (e.g., sensitive emails, messages, etc.).

For that purpose, we assume a remote and/or local adversary that can install a malicious application on the mobile device. According to the permissions given to the application, the adversary can then get access to certain resources (e.g., Internet, contacts, etc.). Here, the adversary can install two or more applications to that can collude to acquire sensitive corporate information. Examples include privilege escalation attacks, collusion using overt and covert channels [7)] .

The adversary could also attach exter-nal peripherals to the device. Here, we assume that the adversary is only interested in acquiring the secret keys used to log in to corporate services and can have access to the entire disk space on the mobile device.

However, we do not consider the case where the adversary installs malware in the device firmware, has access to low/high speed buses or compromises the OS. As far as we are aware, the only workable mechanism that could be used to circumvent such attacks would be the reliance on trusted computing in conjunction with fully homomorphic functions [6)] .

## 4. System Design and Architecture

In this section, we describe a solution that achieves the isolation of applica-tion execution as a means to harden possible tampering with the mobile device.

### 4.1 Overview

In contrast to existing BYOD solutions in the market, we present in what follows a solution that enables the reliance on BYOD while achieving “best-effort” guarantees on the security of the information stored on the mobile device. Our proposal does not require any modifications to the device and can be easily integrated within most existing mobile devices.

Furthermore, our proposal does not penalize the owners of the mobile devices since it only affects the applications/information that belong to the enterprise and that are deemed to be sensitive.

### 4.2 Architecture

Our solution consists of installing a container within the user application space in the mobile device ( **Fig. 1** ). The application container consists of a set of interceptors for both native and java function calls from the application to the system. The container loads the application in such a way that critical function calls outside the application, to libraries or system calls,

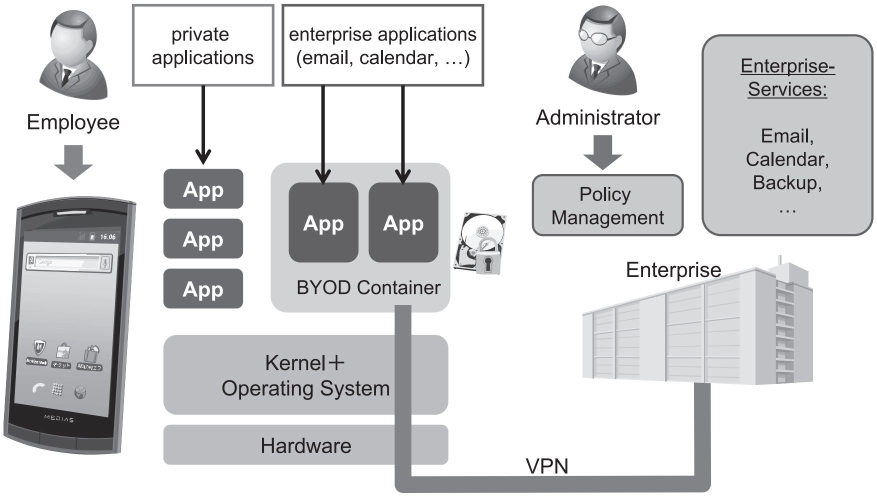
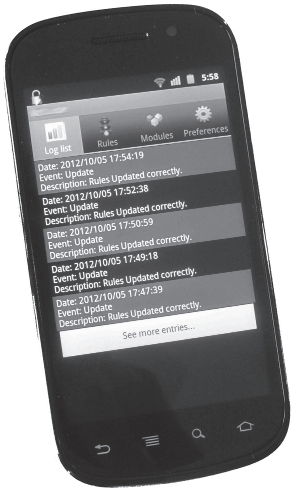


Fig. 1 BYOD architecture and deployment views.

are replaced by our own stubs. In turn, our stubs provide not only extra functionality, such as for example performing a write to the flash memory using encryption, but also enable us to allow or deny certain calls to be performed based on device context and policies. While the same applications will also run outside the container, they will not have access to the enterprise network or the encrypted data in the device.

Our application container is used in conjunction with carefully tailored security protocols that ensure that no long term secret keys can be leaked to the adversary. We assume that the long term secret keys are perpetually stored on tamper-resistant storage such as SIM cards or SD cards. These keys never leave their storage facility and are not exposed to any other entity. Our solution incorporates a security algorithm that negotiates temporary short-lived session keys with the enterprise servers upon the start of every session. This ensures that (i) the adversary cannot acquire long term keys even if she compromises the OS of the mobile device, and (ii) the advantage of the adversary in the system is time-bounded, since new session keys are established upon every session.

As such, our proposed architecture offers a tradeoff between the security and the usability of our system. Note that, within the application container, numerous services such as remote wipe, secure data storage, or fine-granular policy management) can be provided. This can be achieved without affecting the user experience in executing any other kind of application - and without limiting the private functionality of the mobile device.



Date: 2012/10/05 17:54:19
Event: Update
Description: Rules Updated correctly.
Date: 2012/10/05 17:52:38
Event: Update
Description: Rules Updated correctly.
Date: 2012/10/05 17:50:59
Event: Update
Description: Rules Updated correctly.
Date: 2012/10/05 17:49:18
Event: Update
Description: Rules Updated correctly.
Date: 2012/10/05 17:47:39
Event: Update
Description: Rules Updated correctly.
See more entries...

Fig. 2 BYOD application, view of current security issues.

### 4.3 BYOD policy management

At the heat of our solution is a policy based management for all available mobile devices. Given this, our solution offers a modular approach, inside the mobile device, that can be fine-tuned according to the required security level of the company. Our solution offers the IT administrators of the enterprises to remotely activate the security features that are mandatory to achieve the required level of security. While doing so, our solution is efficient and scales with the number of devices in the system; as such, it can be easily integrated within medium and large-sized companies.

**Fig. 2** illustrates an example of a running BYOD application in our solution. One can see that multiple times updated policies (rules) have been received from the MDM server one can safely assume that there were no violations of enterprise IT security policies.

## 5. Conclusion

Even though there are currently a number of BYOD security solutions in the market, these solutions are either vulnerable to a large number of security threats or require modifications of the underlying mobile device OS. In this article, we portrayed a solution that is flexible and user-friendly, while hardening the overall security for enterprise use. Our proposed solution combines the use of an application container, secure storage facilities on the mobile device, along with specially-tailored cryptographic protocols that ensure the secrecy of long-term keys and sensitive materials in the mobile device. We show how this can be achieved simply by installing a single application within the user-level application space of the mobile device.

Clearly, our solution offers a tradeoff between the security and usability of the mobile device. We believe that our proposal can be used to enforce most existing security policies adopted by current enterprises. It also promises a convenient and effortless means for the IT officers of enterprises to remotely manage security policies for a large number of devices. Finally, our solution makes it also possible to enforce to user to follow the IT security policies on all installed application and data from boot-time on.

*Java is a registered trademark of Oracle Corporation and/or its affiliates in the U.S. and other countries.

*SD is a trademark of SD-3C, LLC.

### References

1) Citrix:Best practices to make BYOD simple and secure, http://docs.media.bitpipe.com/io_10x/io_104481/item_530202/BYOD%20Best%20Practices%20Guide.pdf, 2012.3

2) VMware:The BYOD Opportunity, http://www.vmware.com/files/pdf/view/VMware-BYOD-Opportunity-Whitepaper.pdf, 2012.7

3) Good Technology:Good for Enterprise Android Version:http://media.www1.good.com/documents/gfe_android_ds.pdf, 2012

4) AirWatch LLC.:Mobile Security:http://www.air-watch.com/solutions/mobile-security, 2012

5) Array Networks Inc.:Bring Your Own Device (BYOD), http://www.array-networks.com/solutions-byod-bring-your-own-device.html,2012

6) Craig Gentry:Fully homomorphic en-cryption using ideal lattices,Proceedings of STOC 2009

7) Claudio Marforio, Hubert Ritzdorf, Aur&eacute;lien Francillon, and Srdjan Capkun:Analysis of the Communication between Colluding Applications on Modern Smartphones, In Proceedings of ACSAC, 2012

8) Global Platform:The Trusted Execution Environment: Delivering Enhanced Security at a Lower Cost to the Mobile Market, http://www.globalplatform.org/documents/GlobalPlatform_TEE_White_Paper_Feb2011.pdf, 2011.2

9) Trusted Computing Group:TCG Trusted Network Connect TNC Architecture for Interoperability, http://www.trustedcomputinggroup.org/files/resource_files/2884F884-1A4B-B294-D001FAE2E17EA3EB/TNC_Architecture_v1_5_r3-1.pdf, Version 1.5, Revision 3, 2012.5

### Authors' Profiles

**Dennis Gessner**
Research Scientist
NEC Laboratories Europe
NEC Europe Ltd.

**Joao Girao**
Manager
NEC Laboratories Europe
NEC Europe Ltd.

**Ghassan Karame**
Research Scientist
NEC Laboratories Europe
NEC Europe Ltd.

**Wenting Li**
Research Associate
NEC Laboratories Europe
NEC Europe Ltd.

# Information about the NEC Technical Journal

Thank you for reading the paper.
If you are interested in the NEC Technical Journal, you can also read other papers on our website.

**Link to NEC Technical Journal website**

Japanese

English

## Vol.7 No.3 Smart Device Solutions

### ◇ Papers for Special Issue

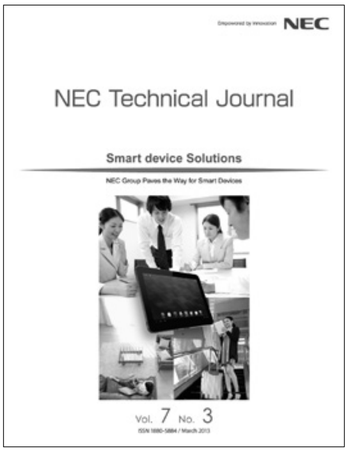
**Service platforms**

**Smart devices**

**Solutions**

**Technical researches**

### ◇ General Papers



**Vol.7 No.3**

**March, 2013**

Special Issue TOP