The Study and Evaluation of Storing Sensitive Data Protection based on Mobile Virtualization

Manjunath R Sneha R
Head of Department of CSE, M.Tech 2nd Semester
City Engineering College Department of CSE
City Engineering College

***Abstract*:- Virtualization technology is adopted to offer several isolated execution environments and make better use of computational resources in server environment. The development of smart phones has been reported the number of security vulnerabilities.Smartphones are a prime target for sensitive personal and corporate data.The extensive utilization of mobile smart devices has led to a series of issues such as security, wasting of resources, and power consumption. We propose a system framework for secure storage of sensitive data in smartphone. The system is divided into general domain (GD) and secure domain (SD) in mobile device utilizing domain separation technique of virtualization, and SD provides a secure execution environment to protect sensitive data and secure services. Lastly, the experiments are conducted to measure the performance overhead imposed by security features in SD and by overall system with inter domain communication from GD to SD.**

INTRODUCTION

Connection of wireless devices or smart phone devices on a hardware virtualization Platform is called mobile virtualization. Enabling computing single unit of data on multiple virtual machine or operating systems simultaneously on a mobile phone or a connected wireless devices. With the increase in computing capabilities of mobile Phone which was earlier found in mainframe computers and workstations. Mobile CPU registers hundreds of MHz and 32 bit processors accesses gigabytes of memory.

Real world and cyber space objects can be fused with the help of mobile virtualization depending on the networks. Smart phones usage are skyrocketing due to increase in exclusive feature added in two departments such as software and hardware.

Recently there is a wide increase of demand in services in terms of data, video and data communication. Enterprise applications are being implemented in mobile phones hence mobile phones are no longer stand alone devices. Mobile or smart phone devices and virtualization have been the two of the trending topics to enter IT in last decade.

Analyzing virtualization in terms of server perspective has been are destroying the IT world. Making VMWare becoming one of the largest software company in terms of market value. Analysts verifying number of servers being run on virtualization was well over 50%.

Mobile devices form the integral part of human life and found everywhere. Bytes of data get generated from the devices through various apps running in phone, various environments have been exposed in areas like security threats such as denial service attack exploiting processed information and exposing the capability of low powered CPU. Mobile devices are now the focal point of cyber-crime and current security strategies are proving to be lethal. They are becoming the targets as they contain diverse and wealthy set of information. There is a new hub for professional communication, gaming, photography, social networking, data access, and personal information known as the smartphone.Hence forming a better target for hackers.

Despite these solutions, users of smartphone still face numerous threats such as sensitive data of user being exposed by unauthorized data access. Thus, users are requiring more smart and secure devices which can securely manage separating workplace resource and personal sensitive data in the device itself. In this paper, therefore, we first discuss the efforts towards solving the above problem in virtualization. Then, we propose a secure storage system based on trusted environments separated from existing platform of smart electronic devices and present secure functions to securely manage sensitive data in trusted environments.

This paper describes virtualization and propose system framework for secure storage based on trusted virtual domain and suggest secure functions to be operated in trusted virtual domain. Section 4 presents the experimental results about the performance overhead of secure storage system. Lastly, we summarize and conclude.

RELATED WORKS

In virtualization it makes use of a single piece of software, which every time operates in kernel mode. The hypervisor, which is at two orders of magnitude smaller than normal

Operating Systems and very rare to encounter failures. Two approaches used to implement the technique are by using hypervisor type 1 or type 2. Explaining hypervisor type 1, which is also known as hardware level virtualization, without any reference it can be considered as the operating system, as it is a one piece of software that always work in kernel mode. The main task here is to manage multiple copies of real hardware similarly like an OS which is managing multitasking. According to type2 hypervisor, which is also known as operating system level virtualization, with own terms hypervisor can be compared to another user application which simply completely "interprets" the guest machine. In recent days, virtualization is gaining all the cynosure as a possible security tool adding the new feature for resource sharing.

On providing a high degree of gap between individual Environments, restoring the complete OS and all functions and processes under it is easier and possible compared to the traditional single operating system environment. Usage of virtualization in terms of confinement, the terra virtual machine and Qubes operating systems were brought into picture.
The Terra being a virtual machine monitor (VMM), which is a s/w that helps in managing several virtual machines, in order to provide isolation and privacy between them. There are differential points between Terra and normal virtual machine monitor, one being defined types of virtual machines that actually can run on Terra. The other point is open boxes – real and traditional VM's have no distinction, closed boxes – which ensures privacy and integrity of the contents through isolation.

For managing processes and providing each with own view under virtualization in terms of operating system Qubes operating system serves as a hybrid virtual machine monitor.
All processes act as if they are operating in the same environment and can be viewed on a single screen which is the main difference of terra with Vms. Even though there is a strong isolation there is a partial improvement in the usability drawbacks for the mechanism of file transferring between security domains on the machine for a Qubes Operating system. For ensuring proper isolation of the closed VMs from other VMs they make use of a mechanism called trusted platform module hardware.

Verification of startup process I done through trusted platform module, but still has many leakages to unauthorized software, as it runs on the guest machine, In order to increase the software adaptability to virtualization storage capsules has been introduced in Terra for convenient usage for the users.

In order to switch between secure and in secure modes in attempt to loose or gain sensitive data, which will be stored in a secure VM having a common channel to guest VM can be managed through a single untrusted guest operating system. Capsules are designed in order to prevent the leakage of data during the event of comprising with the operating systems. They also simplify usability when user's access files available to them in insecure mode during strong isolation and they are unable to access the data which is outside a secure mode.
Usage of virtualization in mobile devices is due to the result of technology advancements as virtualization solutions have been around for years in the pc/ laptops markets. Enterprises will have a workspace management tool available to them for suitable management of devices in BOYD environment as the technology gains acceptance in the upcoming years.

In order to create a virtual work and personal profiles on an employee's mobile device through mobile virtualization we make use of type 1 or type 2 hypervisor which provides greater control and security while the personal area is kept private from the business workspace. While providing mobile virtualization there are few issues to be taken care even though it is similar to desktop/enterprise virtualization which are memory capacities, current mobile processors which is lacking virtualization support,hardwarerequiring paravirtualization support, Limited battery life imposing usability restrictions. Indulged researchers researched into lighter weight virtualization systems.

By the separation and isolating applications and data a policy based framework is presented by Moses system for android. The notion of security profiles in Moses is the crucial one. A security profile can access the data of the same security profile and each one represents a unit of isolation. A combination set used by Moses consists of taint Propagation, reference monitors, and file system namespaces in order to achieve a strong isolated manner of information. Reference monitors and taint trackers behavior can be controlled by policies, it can be configured to switch automatically based on the devices physical context. A lightweight virtual smartphone architecture for android has been proposed by Cells. To conclude, in network-based high

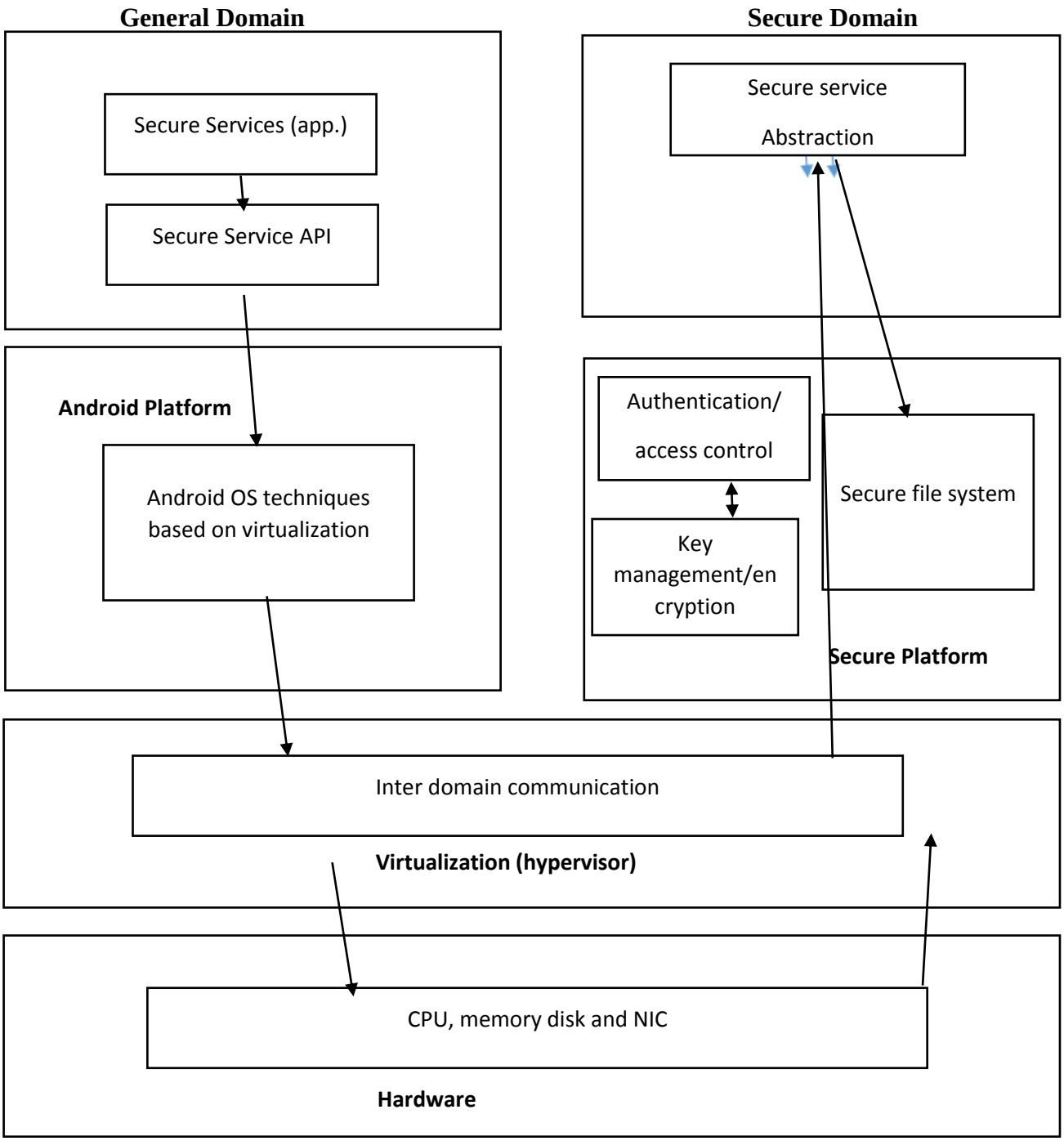

Fig: The system framework for secure storage based on domain separation.

Cloud computing or server system virtualization technologies can be provided. There has been a focused limitation on running on an identical OS, policy-based framework and switching secure and in secure modes for mobile virtualization techniques. But still there is much to work
for the development of novel mobile security system providing usability and secure schemes optimized by characteristics of devices based on mobile virtualization.

PROPOSED SCHEME

A very effective security technique of isolating the data and execution is through virtualization. Hence a system framework is designed through this paper based on domain separation on virtualization to protect sensitive data of smartphone which operates on virtual secure domain and to present secure functions.

A secure storage system framework

Using hypervisor of paravirtualization virtual secure domain (SD) is separated from general domain (GD).

A strict Prohibition has been made to make a direct call with GD from SD by a virtual machine (SD) which is created through virtualization. A distinct standalone execution environment is made use for the secure functions in SD which is separated from existing mobile platform**.** A secure storage system framework for smartphone is shown in figure 1, consisting of four layers: virtualization application, hardware and platform. Android part of GD and secure part of SD are separated in the application and platform layer. An embedded and real time operating systems composed of secure functions - μC/OS helps running of OS, middleware, secure platforms in platform layer, while the existing android OS and functions run the android platform. Container manages the personal android apps which is distinguished from secure service apps and all user apps can be downloaded in GD. To manage data of apps in SD, SS make use of API's and only those requests can attempt inter domain communication to transmit the data to SD.

Through two authentication steps IDC is permitted. First step being at the stage of app use by the user. a user is registered at the early stage and pin is encrypted and stored into SD. Second one being app authentication by using AppID. In order to block the access or use of data generated by app A from app B, A unique AppID assigned to each app is used. Through policy setting prescribed for access control in SD we provide the facility for data sharing between apps in order to provide flexibility of data access. A session between domains for message communication is opened up by IDC if the authentications are successful. With the help of SS abstraction IDC are mapped to security functions for the app requests to reach SD through IDC. To represent secure API's in SD to call secure functions SS abstraction is used. A detailed secure features to help data protection comprises in secure functions, adding secure file system, key generation, access control and encryption are also included. Hence now only authorized SS API'S through IDC can access security functions in SD. Permissions for user applications to directly access SS abstractions or security functions in SD is not provided in our system because user applications are not allowed. SD in security functions have mutual relations with each other.

To encrypt the file data received from GD SFS calls encryption functions and key management Functions which generates the file encryption key is called by encryption functions. In combination access control and authentication are applied along these functions on platform of SD. Goals of the attacker is clearly been stated: compromise, privilege escalation, container compromise and denial of service. These attack groups are further classified into two classes depending on the type of attack mechanisms. Attack mechanisms can be via storage and communication mechanisms. A set of security requirements are derived from this classification. System is analyzed through these security requirements. Existing android apps are denoted by Di (i=1, 2… N), SS apps by Si (j=1, 2…., M). *Di* And *Sj* do not allow overlapping and maximum number of apps are denoted by N and M.

(a) Separation of processes – Processes running in distinct containers can be isolated through this.

Through container di can be separated from si of the proposed system, allowing only sj to call the SS API in a secure container. Through special user pin the container can be managed. To prevent the illegitimate access through IDC from GD to SD and even if Di can directly access SS API's of Si by the help of PIN and AppID. Importantly distinct embedded OS operates SD meaning all processes in SD are totally distinctive form GD.

IDC isolation is needed to prevent *Di* from accessing or modifying data of *Sj* being transmitted over IDC channels. To prevent illegal access of data in IDC, IDC can perform encrypted communication and message queue for IDC can keep the encrypted messages. Besides, IDC does not allow channel open if user and App authentication fail in modifying message for
Communication between domains.

(iii) *File system isolation* is required to prevent illegitimate access to file system objects. SFS of our system is located in SD, and it can be only accessed by *Sj* having legitimate user authority and app authentication information.

(iv) *Device isolation* should protect device drivers shared between different containers or domains. Our system is based on the paravirtualization, on which hypervisor resides on top of the hardware and operates through a set of low-level routines with the hardware called hypervisor calls. We maintain independence and safety of device drivers in SD on the assumption that the embedded OS of SD can interface with the hypervisor through these hypervisor calls.

(v) *Network isolation* aims to prevent attacks by *Sj* via available network interfaces. Since SD does not permit any direct network communication such as app download, system, and security policy, our system is guaranteed safety from attacks through network. Instead, settings for secure platform can be securely supplied from GD.

(vi) *Resource management* is needed to limit the amount of resources available to each domain depending on the system load. The resources for SD in our system are allocated appropriately considering embedded OS through hypervisor. The closed SD can protect the physical resources available from exhausting works by intentional attacks.

3.2. Secure Functions in Secure Platform.
In this subsection, we introduce secure functions run on the secure platformin SD and mention concretely the SFS closely associated with secure storage of data. Secure functions of the proposedsystemcanbe classifiedintothreemodules: authentication/ access control, encryption/key management, and SFS. The role andmethod of functions are as follows.

(vi) Authentication and Access Control Module.
(i) The proposed system performs access control function by requiring user and app authentications for communication channel open in IDC. First, the unique AppID given to each SS app is utilized to verify whether app has the legal authority to access and to use other secure functions in SD. If AppID is successfully verified, our system identifies user PIN to check whether SS app is being used by legitimate user. This module is linked with SFS module for managing AppID and PIN information. In addition, new secure policies can be set in SD to provide the strict access control between domains, apps, and functions, and policy data should be also managed through SFS.

(ii) Encryption and Key Management Module.
All data transmitted from GD is encrypted before storage through SFS. Also, the encrypted data is decrypted when it is called for use for other secure functions. The encryption module can provide various algorithms and operation modes depending on symmetric or asymmetric key. We assume that keys for encryption/decryption are provided on root of trust, which are inherently trusted using hardware/software components, and they are generated andmanaged only in SD.

(iii) Secure Filesystem (SFS) Module. Sensitive data transmitted from GD is managed by SFS. This module is connected with encryption module to request data encryption/ decryption and is linked with authentication and access controlmodule to provide information related to authentication or secure policy. Here, we describe SFS in detail. As shown in Figure 2, SFS consists of four parts divided into volume, bitmap, objects, and files. Volume supports the system information like boot record, and bitmap is used to manage total blocks composed by file system. Here, the used block is expressed as "1" in a bit. Objects,which are called inodes in general file system, include information to find and access the file. Lastly, files mean the real file data to be stored in the storage. For simplicity of expression in this paper, the scope of volume, bitmap, and objects is referred to as a filesystem metadata, and each object is called a file's metadata. The secure features for SFS have two aspects as follows.

(i) First is file data protection.
(a) Each file is encrypted by various algorithms and operation modes. The encryption key can be generated by root key of hardware chip and theseed value for generating the encryption key is stored in the corresponding file metadata.
Whether the file encryption is applied or not, it is marked to "Crypto flag" field in file metadata.
(b) When a file is recorded, a hash value of each file is generated by hash functions like MAC (Message Authentication Code) or MD5, and it is stored in the corresponding file metadata with length 20 bytes. Also, file integrity is verified whenever the file is read.

(ii) Second is filesystem information protection.
(a) Exposure of file system metadata can still threaten the stored file data in attack. Thus, bitmap and objects in file system metadata are encrypted to securely protect data in hardware. At this time, volume is excluded for encryption since its values should be used to mount SFS to memory. If the filesystem metadata size imposes loads, nevertheless, some parts of objects can be selectively used for filesystem metadata encryption.

(b) Integrity of SFS is also checked when mounting or remounting. Verification value is generated using bitmap and objects by the hash function at the unmounting point, and it is restored in volume.

CONCLUSION

Since sensitive information stored in an insecure manner is vulnerable to theft, the ways to safely store and manage data have been the focus.Thus, the proposed system framework is to protect sensitive data of smartphone. It provides general domain (GD) and secure domain (SD) in mobile device.
Utilizing domain separation technique of virtualization. Namely, GD means general android execution area, and SD can provide a secure execution environment that runs secure functions to securelymanage all data input by secure service apps in GD. These sensitive data by secure service apps can be called through only secure service API and can be transmitted through inter domain communication (IDC). As IDC requires user and app authentications, the secure communication between domains satisfies isolation of file system, network, and IDC.

In addition, we suggested the secure functions such asAuthentication/access control, encryption/key

management, and secure file system, and especially secure file system is discussed as a key function for secure storage. Thus, we evaluated the performance of secure file system imposed by security features in SD and by overall system based on IDC. While the
SFS imposed a lot of time overhead in SD, the performance based on IDC is almost not affected since IDC consumes a pretty long time for basic communication. It will provide many possibilities about security functions based on virtualization domain.

The target data of our system were small size data set as sensitive private information. As amounts of memory of latest smartphone are occupied by pictures or videos, processing methods for big size data should be considered together in our system. Therefore, we first will enhance scalability and efficiency of our system, and then we will suggest various secure functions which can be combined with our system and describe each security function in detail.

ACKNOWLEDGEMENT

REFERENCES

1. A Secure Storage System for Sensitive Data Protection Based on Mobile Virtualization by Su-Wan Park, JaeDeok Lim and Jeong Nyeo Kim
2. When Virtualization meets mobile article by Tom Kemp
3. Smartphone virtualization: Status and challenges conference paper by Xiaoyi Chen.
4. What is Mobile Virtualization by General Dynamic's broadband.
5.'Mobile Security' Through Mobile Virtualization and Cloud Mobile Virtualization by LG CNS paper.