Journal of Theoretical and Applied Information Technology
20th September 2015. Vol.79. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DEVELOPMENT OF TRUSTED OPERATING SYSTEM FOR MOBILE DEVICES

D.M. MIKHAYLOV, M.I. FROIMSON, A.V. ZUYKOV, A.S. SMIRNOV, A.V. STARIKOVSKIY, I.A. OVCHINNIKOV, R.O. ROSLAVTSEV, D.A.ANDRYAKOV

National Research Nuclear University "MEPhI" (Moscow Engineering Physics Institute),
Kashirskoe highway 31, 115409, Moscow, Russian Federation
E-mail: polynna@yandex.ru

ABSTRACT

The number of mobile devices used daily all around the world is growing every day. Mobile devices have become "smart" and, thus, suffer intrusion attempts. Although different antivirus software is being developed, it does not provide overall security. That is why trusted operating systems appear. Trusted (or secure) operating systems for computers exist and are widely spread. However, there is none for mobile devices. This paper is devoted to the development of a trusted operating system for mobile devices. Authors highlight the differences in security issues of ordinary computers and mobile devices and discuss intruder models used in PC operating systems (OS) and in mobile OS. Generation and analysis of requirements to a secure mobile OS and secure mobile OS architecture are provided. Authors also tell about general scheme of the mobile device's access to the Internet and provide suggestions to improve current security policy.

Keywords: *Android, secure operating system, trusted operating system for mobile devices.*

1. INTRODUCTION

The issue of mobile devices` (devices of personal use such as mobile phones and tablet computers run by mobile operating systems) security is of great interest today as they have become an integral part of our everyday life and the value of the processed information keeps growing. Many scientific papers have been devoted to vulnerabilities exploration [1-4] and proposals of protection means for mobile devices [5-9].

The capability of mobile devices is provided to a large extend by functionality of the running operating system (OS). The basic threats to information are relevant for mobile OS as well. Different scientific papers were devoted to security issues of OS.And many of them deals with the trusted OS.

The "trust" of the system is an ability to provide a finite set of features previously known to the user and appropriate for its working conditions within the system without compromising its data, as well as the lack of unknown for the user functionality, representing a potential threat of confidential or personal data disclosure.

The OS can be defined as trusted (secure) if we know the full list of features and these features ensure the safe operation with data. Otherwise, it is untrusted.

For example, Hanspach and Keller inter alia provide a set of output filters that can be applied to trusted OS components to enforce higher level security goals [10]. In [11] the concept of trusted real time operating system is discussed. Xiao-Wei Nie et al. investigate security operating systems based on trusted computing [12]. The design of trusted OS based on Linux is described [13].

However, the use conditions of mobile devices impose additional requirements that must be considered when developing a mobile OS protection system. One of the most important differences is the way of interaction with user. Due to small dimensions and weight of devices, mobile phones are often used "on the move" (in transport, when walking, indoors, etc.).

Another important difference is the intensity of communication with the outside world and the speed of its preparation. Today a mobile device has become the means of interaction by which a user can in a very short period of time access any information available in the world. Reading news, watching weather forecasts, ordering tickets, watching friends' photos, e-mail messaging - everything has become a routine part of a mobile device use and should be provided by manufacturers of mobile operating systems and devices.

In addition, when using PC a person often has

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

one or several personal devices and one or several corporate devices, each of which is used for personal or business purposes respectively. In case of mobile phones, most often there is no distinct boundary between business and personal device. The same device can be used for processing personal and corporate data, for access to social networks and corporate network services, for using a personal and corporate mailbox etc.

These factors point out clearly visible differences of mobile phones and tablets not only from personal computers, but also when compared to other mobile devices such as laptops or netbooks. Requirements for the protection different from those for personal computers operating systems of information should be introduced.

Thus, this paper is devoted to the issue of development of trusted operating system for mobile devices. It is notable that such system does not exist today.

2. INTRUDER MODELS USED IN PC OPERATING SYSTEMS COMPARED WITH THE INTRUDER MODEL FOR MOBILE OS

The main differences between traditional operating systems and mobile ones are the following:

— Only one user of the mobile device. Authentication procedure is by simple password entering.

— Probability of physical movement of the device (smartphone) and interaction with infrastructure (GSM, Wi-Fi)

— Inability to provide a constant complete physical protection of the device against unauthorized access.

— Session length for the mobile device.

Mobile operating system has a number of differences for the information interloper compared to the model of the intruder used in standard information systems [14]. For example, because of specific use of mobile devices, the traditional division of intruders into entitled and not entitled to access the controlled area (internal and external intruders) is relative.

Another factor is the hardware on which such a system is installed. In particular, unlike the PC, smartphones are often equipped with tools such as microphone, camera, navigation tools, etc. Meanwhile, the majority of modern mobile communication devices are equipped with such traditional hardware inherent to PC as slots for external drives, ports for connecting external devices (mini and microUSB or proprietary interfaces). Thus the possibility of unauthorized access to the information by using these means, including abnormal mode, should be considered.

Moreover, mobile operating systems used in smartphones often are private and cannot be adequately studied to assess the level of security. Among other reasons there are low-quality domestic mobile communication devices and foreign counterparts and the inability to carry out a full check of the design and program documentation on these devices, according to the requirements of national guidance documents. For instance, in 2011 information about covert tracking of mobile device users by Apple received wide publicity [15]. Information about the location of each subscriber was registered by the operating system in a special consolidated file [16]. This information was then transmitted to a company data center in the U.S. that was not declared in any specifications or instructions for these mobile devices. This indicates the existence of a real threat to privacy and data integrity, access to which can be gained by using operating system vulnerabilities and by introducing malicious software (including mobile device manufacturer).

Finally, peculiarities of the data transmitted by mobile devices must be considered when developing proposals for the establishment of security policy for mobile operating systems. Normally according the documents regulating the work with the restricted information do not entail storage and processing of such information on mobile devices. Despite that fact, the described technical capabilities of the smartphone may allow obtaining unauthorized access. Besides, the intruder can get all data on the location of the subscriber and the surrounding area through unauthorized access to the microphone, camera and phone navigation tools. In turn, it may lead to a breach of confidentiality of both personal, commercial, proprietary information and even state secret.

Thus, the intruder model used in traditional computer systems and networks cannot be effectively applied when working with mobile devices that operate under their own operating systems. This implies the need for the development and commissioning of specialized regulations and information security policy.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. GENERATION AND ANALYSIS OF REQUIREMENTS TO A SECURE MOBILE OPERATING SYSTEM

We will consider the most popular mobile operating systems' means of protection in order to formulate the requirements for secure mobile operating system (taking into account the use conditions mentioned above).

The most popular OS at the time of analysis, as well as the OS that are positioned as protected but have not yet entered the market were considered:

— Windows Phone;
— Android;
— iOS;
— BlackBerryOS (RIM);
— Tizen [17].

Due to the fact, Android OS is one of the most popular systems and it is open [18], [19], Android OS was chosen as the basic operating system [17].

Android OS architecture [20] is shown in Figure 1.

The requirements for a secure operating system projecting on the selected basic mobile operating system are discussed below.

Access control. Authentication and Identification

According to the use conditions of mobile devices, mobile operating system must work in single-user mode and should not provide collective use of a mobile device. It means that the user authentication should be required. There are no requirements for user's identification in operating systems.

Basic operating system assumes the possibility of identifying user in the system in the following ways:

— PIN-code of the device;
— alphanumeric password;
— graphical key;
— user's profile;
— NFC-tag.

To ensure compliance of the developed OS with the requirements formulated earlier, a restriction on the minimum length of the user's password is necessary. The alphanumeric password should comprise at least six characters. The following security policies concerning the user's password can be introduces by the device administrator:

— type the password (PIN-code/password, etc.);
— minimum length;
— mandatory symbols.

The infrastructure should provide means of identification and authentication of a mobile device for access to the services. Identification of the device can be carried out according to the identifier specified by the administrator when initializing the device. Authentication in the infrastructure is carried out according the user's password.

Number of attempts to authenticate one user should be limited.

Access control. Discretionary model.

Access to resources should be limited on discretionary principle. A clear and unambiguous enumeration of permissible access rights for a given subject (application) for this resource must be provided for each pair (subject - object).

Since the Android OS is based on the Linux core, discretionary security model of the developed mobile operating system is respectively based on the Linux security model.

On Linux a file index contains information about the file owner (UID), its primary group (GID) and file access vector for the three categories of subjects - the owner, members of his group and others.

There are three permission rights - read, write and run. For directories permission to read enables the user to view the contents of the directory; permission to write allows to create, add and delete files in the directory; run permission means permission to search for a file in the directory's name.

Moreover, individual access rights may be implemented through access control lists (ACL). This option allows introducing more flexible access rights settings of objects to subjects. For example, ACL may be used to set access rights to a directory to read for one group and to write for another group. However, engineering documentation does not generally need this requirement for automated systems with one user and one or different levels of privacy and does not need to fine-tune user's access to system files in the mobile operating system, implementation of ACLs is an unnecessary requirement.

Access control. Mandatory model.

Each object and subject in a system should be given a mandatory label - security context. Applications (each application has a unique Linux user) and the processes they run are considered as subjects; files, sockets, named pipes are considered as objects.

Implementation of mandatory access to data is based on Security-Enhanced Linux (SELinux) integrated with the basic mobile operating system.

SELinux supports the following modes:

— Permissive - violation of security policy

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

is permitted. Such violations are only recorded in the system log. In fact, SELinux does not operate, but only records security policy violations.

— Enforcing –security policy violations are blocked.

— Disable – SELinux is off.

In basic OS a default value is running in logging mode. In protected mobile OS SELinux mode should be switched to Enforcing. Some improvements have to be introduced to implement the required confidentiality tags and restrictions on access of subjects to objects.

There is also a need to impose restrictions on access to the device functions while using applications. Thus, ban access to particular Android functions while allowing continuing to run application without access to selected resources will be possible.

Event registration. A logging subsystem should register orders to the relevant function of the core and save information about the event in a separate log accessible only for a system administrator.

Cryptographic subsystem should store protected information encrypted using Russian Encryption Standard.

Integrity control subsystem. Integrity control must be implemented by the operating system loader. During OS loading images, checksumming of the loading system should be implemented and the results should be compared to the reference checksums.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

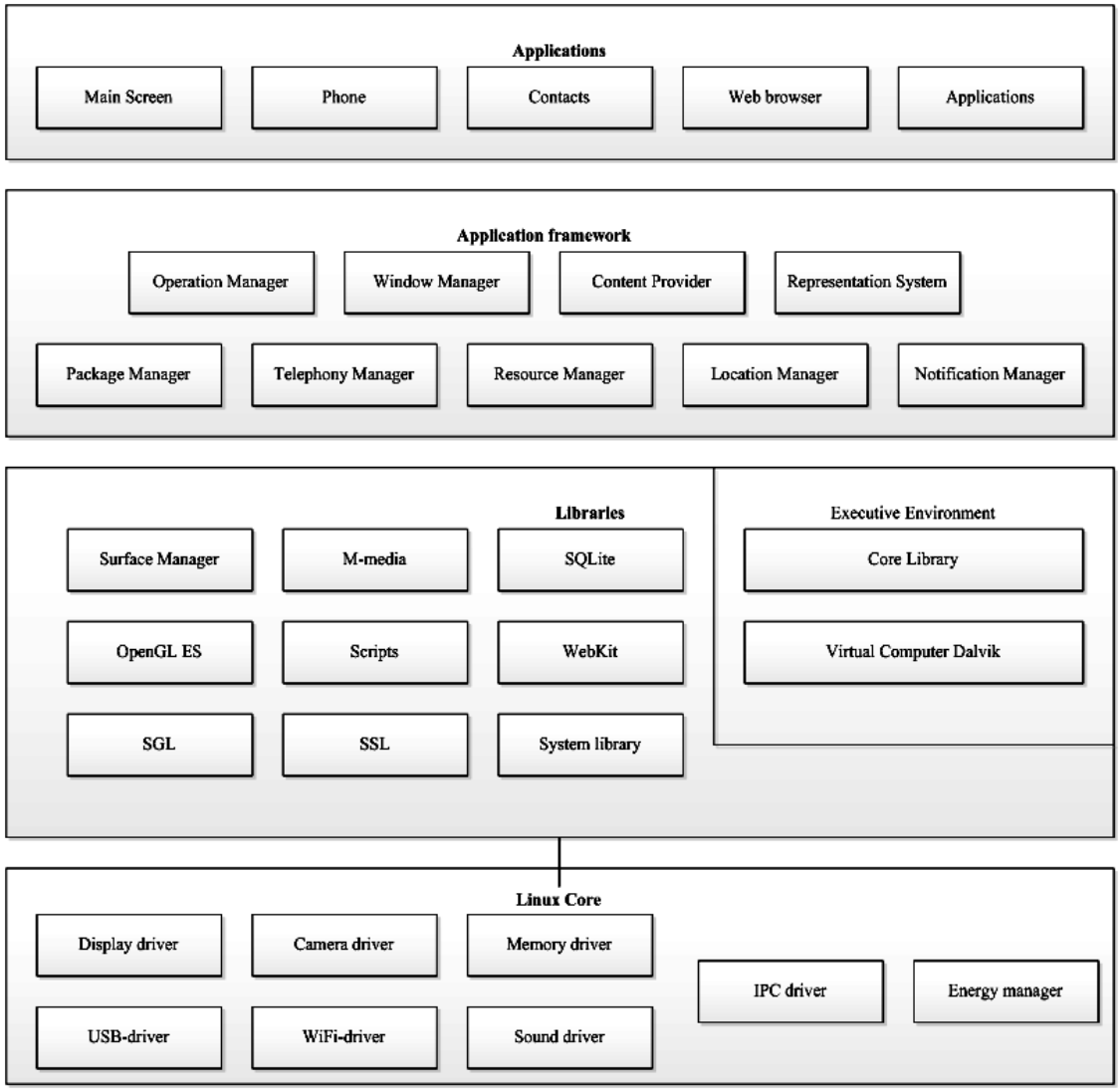

Figure 1: Android OS architecture

The dynamic integrity control should ensure the integrity check of information security system (ISS) components during system's operation. In case of errors during checksumming the the system administrator must intervene: administrator notification, blocking the device, deleting the key information and protected data or a complete system reset.

Basic OS comprises means of digital signature verification software when the latter is installed. However, anonymously signed application can be installed. This option should be removed from the developed OS. Besides the digital signature verification must be implemented in accordance with national standards.

Interworking subsystem

A mobile device should ensure access to the Internet, which means threats of attacks through this channel. The traffic of the mobile device should be subject to additional tests and analysis. Given the limited resources of the mobile device this functionality is most appropriate within the mobile device framework. Thus, access requests to the Internet resources and the results are analyzed by means of a private network without any limits for the user.

Thus, a virtual network interface performing secure communication with the corporate network is implemented within the OS. Requests from mobile devices are forwarded to the Internet, and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the response is further received and analyzed. If a framework with implemented traffic analysis service considers traffic safe, response is forwarded to the mobile device.

Ensuring compatibility with existing software

As the digital signature verification module of the installed software requires modification, additional installation package signature should be implemented to install third-party software. This restriction does not impose additional difficulties on the process of software development and requires no changes to the code of already existing software.

In addition, in view of allowing the user to set limits on the access of applications to device functions, software developer must correctly handle exceptions resulting from the access refusal. A standard list of exceptions can be used for the purpose and it does not require additional training of software developers. This requirement is not new, but may influence the performance of existing software.

Clearing liberated areas of random access memory (RAM) and external drives.

For RAM allotment, the Linux OS memory manager uses the "copy-on-write" principle. As a result, the process gets memory pages refined from the prior process.

However, the blocks of disk space are not cleared after a file was deleted. Nevertheless, since storage of protected data uses cryptographic disk, this requirement is a requirement to cryptographic disk functions.

The general scheme of the requirements and their implementation in a secure mobile operating system is shown in Figure 2.

Thus, requirements to finalize the basic operating system are formulated in order to bring it in line with the previously defined general requirements for secure mobile OS.

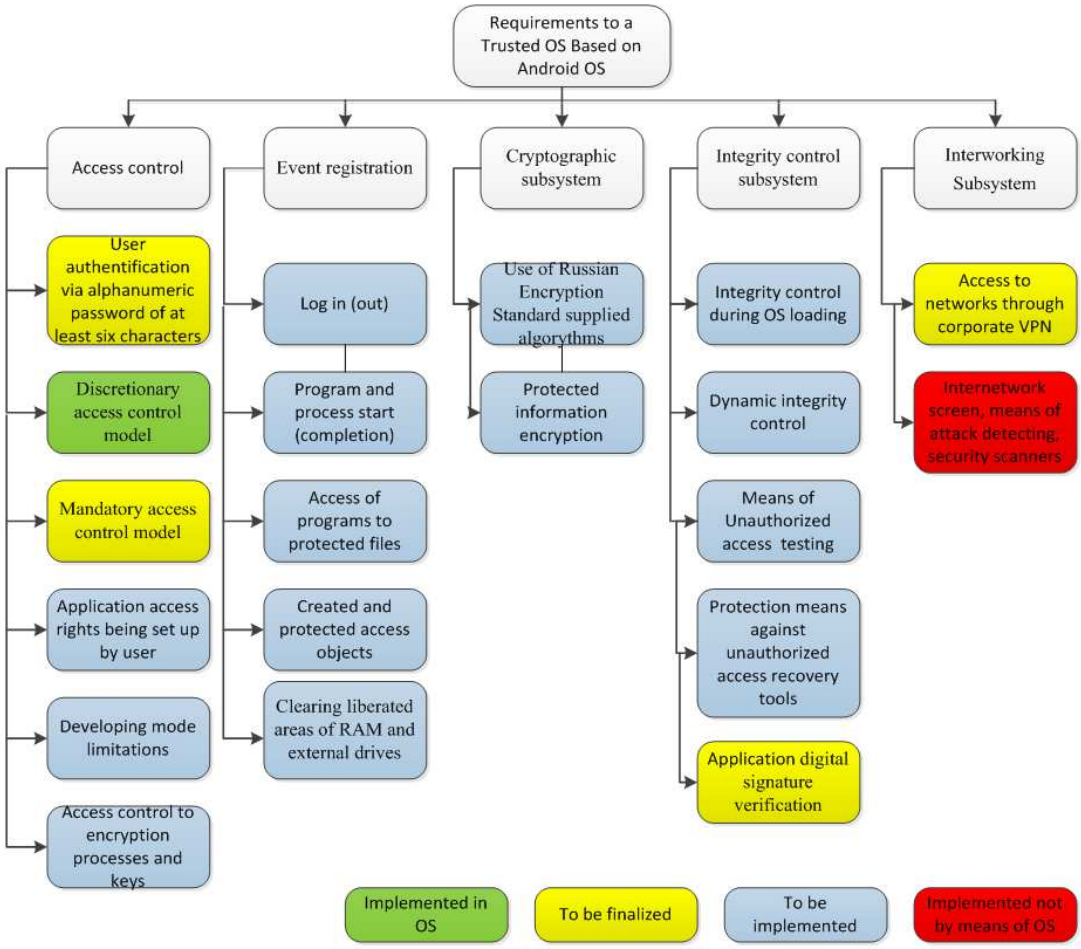

Figure 2: Requirements to a secure mobile operating system based on Android OS

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. SECURE MOBILE OS ARCHITECTURE

Since the Android OS is based on the Linux core, the implementation of security mechanisms is supposed to be done by means of core. The mechanisms to access the ISS at a higher level are provided.

To ensure compatibility with existing software certain components of the operating system should be finalized.

The proposed architecture of a secure operating system (ROMOS) based on Android operating system is shown in Figure 3.

The necessity for implementation of these functionalities on different levels of operating system is discussed below.

System integrity control should be undertaken at the outset of the OS running. Start of the loader is the earliest stage of the system operation. Therefore, the most appropriate option for system integrity control is checksumming system images within running of the loader.

After starting the system dynamic integrity control should be performed at the lowest level at which ISS are located, i.e. at the level of the OS core.

Mandatory and discretionary access differentiation is implemented by means of the Linux core.

Protected information encryption can be realized through creation of the core module that implements a file system with data encryption. Another option is through encrypting particular files, record field and other information within the application software directly, i.e. creating appropriate libraries and providing software access to them. Nevertheless, core module creation allows using the same interface for all cryptographic operations without modifying the very method of information handling. This approach allows the user to set up the applications installed after the operating system's installation. The storage of application data is in an encrypted form without modifications to the application back code. Thus, the possibility to work with existing third-party software will be preserved. Thus, creation of a core module that implements a file system encryption is subsequently a more flexible approach.

At the level of OS core a virtual private network (VPN) module is also implemented in order to create a virtual network driver to transfer the traffic. User authentication is also performed by VPN module in the corporate network.

Traffic encryption is implemented by VPN module using cryptographic information protection facility (CIPF) libraries at libraries level.

Journal of Theoretical and Applied Information Technology
20th September 2015. Vol.79. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

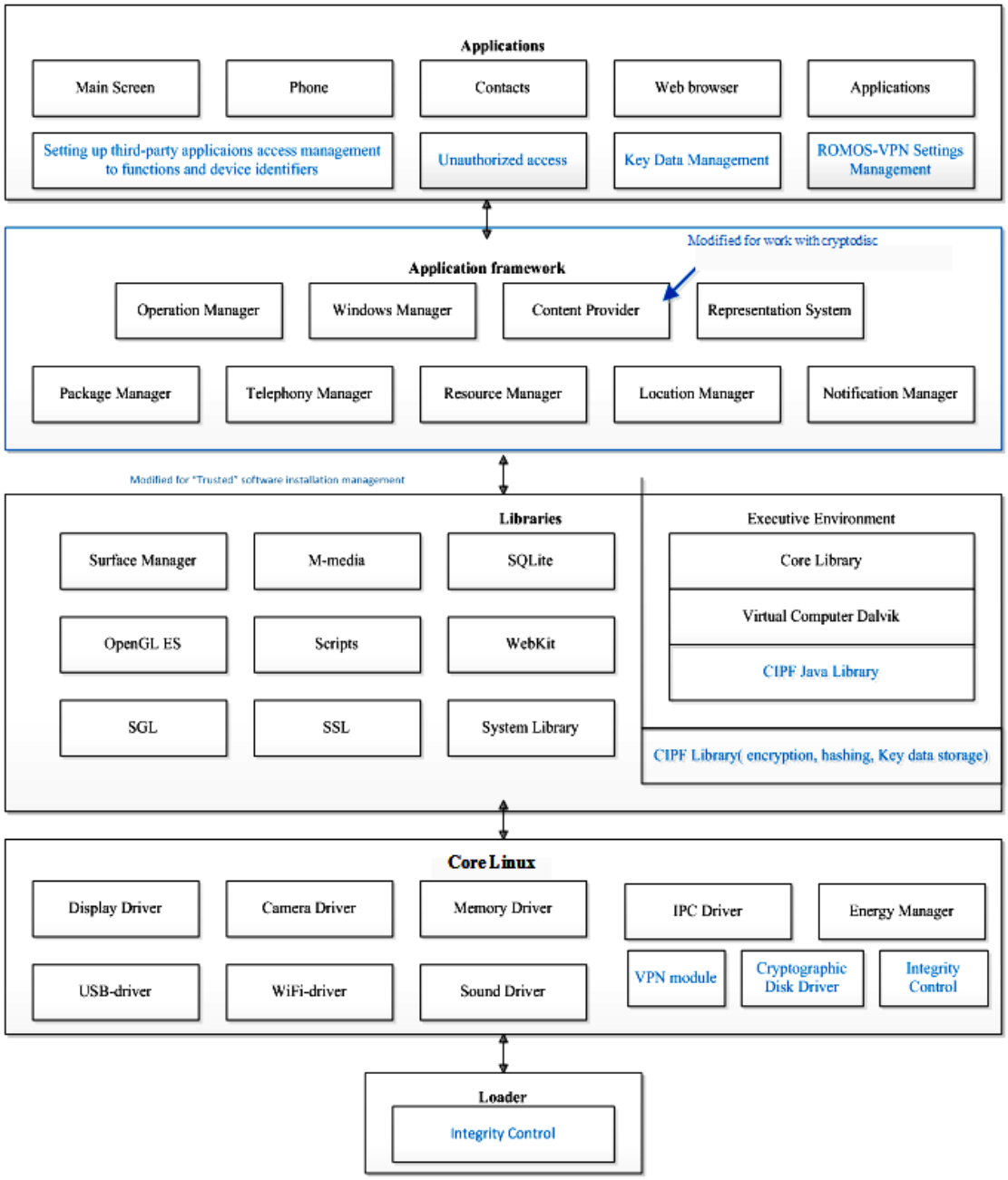

Figure 3: Architecture of a secure mobile operating system based on Android operating system

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

At the level of application framework the package manager is modified in order to control software installation. In Android OS this manager implements electronic signature verification of installed software. Additional restrictions on the digital signature verification must be introduced to implement the requirements of the mandatory digital signature of all the applications installed in the system, as well as requirements to limit potential sources of applications to the trusted only (the signature cannot be anonymous and must belong to a known source). Besides, signature verification is implemented according to national standards.

To work with cryptographic disk the standard content providers of the applications included in operating system, such as "Contacts", "Messages", etc. are also modified. Default storage way undergoes modifications as well.

At the user-application level event-based ISS control is implemented. These tools allow to track user's location by standard means of Android and, if necessary, to block access to, or destroy information. Implementation at the application level is caused by lower labour inputs of this approach without visible deterioration in terms of safety compared to the implementation at the OS core level.

Application software providing application access to the device functions rights control, carried out by user and/or the security administrator, is also introduces at the application level.

The default settings for the application responsible for the identification of the user in the system also can be modified. By default it is mandatory to use alphanumeric password comprising at least 6 characters.

Event registration in the system is carried out at different levels of the operating system's architecture. In most cases, system safety-related event registration is implemented at the same level as the system module responsible for the registered event. Thus, the following events are registered at the application level:

— user's log in (out);
— change of applications access rights to the device functions;
— entering (leaving) target area.

The following events are registered at the library level:

— access to protected files;
— protected objects creation.

Programs and processes start (completion) registration is carried out at the level of executive environment.

5. GENERAL SCHEME OF A MOBILE DEVICE ACCESSING THE INTERNET

General scheme of a mobile device accessing the Internet is shown in the Figure 4.

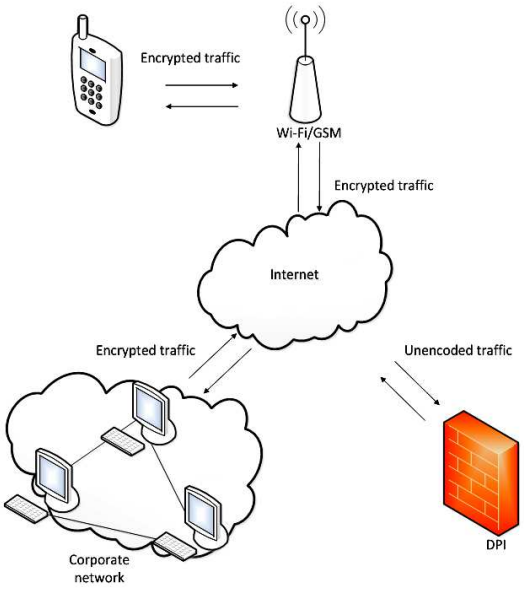

Figure 4: Scheme of a mobile device accessing the Internet.

Traffic exchange occurs in the following scenario:

— The encrypted data is transmitted from the mobile device to the corporate network via the Internet;
— when received by the corporate network the data is decrypted and analyzed using DPI;
— further, if a query is to be transmitted to the Internet, transmission and receiving response is carried out;
— the answer received is analyzed by means of DPI;
— if the query is legitimate, the traffic is encrypted and transmitted to the mobile device via the Internet .

VPN support must be implemented to ensure correct operation of the mobile device according to the scheme. VPN implementation is carried out by

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

creating a virtual network driver in the Linux core. This driver will exchange encrypted traffic within the corporate network.

6. TESTING

The proposed trusted operating system for mobile devices – ROMOS was tested and the results of its performance were compared with general and encrypted Android OS. The testing evaluated the performance of functions of the OSs` to analyze the possible deterioration in the performance of the device during operation. The results are shown in Table 1.

The test results show slight variations in performance compared to the basic OS.

Table 1: Specifications and Performance Comparison of General and Encrypted Android OS and ROMOS

Parameter	Value			Dimension
	Generic Android	Encrypted Android	ROMOS	
Graphics				
Total graphics score	329 ± 10	325 ± 15	350 ± 40	
Draw opacity bitmap	160 ± 6	157 ± 8	170 ± 20	MPixels per sec
Draw transparent bitmap	41 ± 1	42 ± 2	41 ± 5	MPixels per sec
CPU (central processing unit)				
Total CPU score	3000 ± 100	2984 ± 117	3050 ± 120	
MWIPS DP	199 ± 5	196 ± 8	200 ± 6	MWIPS(DP)
MWIPS SP	247 ± 8	249 ± 7	252 ± 8	MWIPS(DP)
MFLOPS DP	27 ± 5	26 ± 6	28 ± 6	MFLOPS(DP)
MFLOPS SP	49 ± 13	43 ± 13	47 ± 16	MFLOPS(SP)
VAX MIPS DP	151 ± 4	151 ± 3	154 ± 4	VAX MIPS(DP)
VAX MIPS SP	160 ± 7	162 ± 7	166 ± 6	VAX MIPS(SP)
Memory				
Total memory score	450 ± 80	420 ± 90	420 ± 70	
Copy memory	410 ± 70	380 ± 80	380 ± 70	Mb/sec
File system				
Total memory score	260 ± 15	216 ± 12	210 ± 8	
Creating 1000 empty files	0.49 ± 0.05	0.491 ± 0.03	0.51 ± 0.06	sec
Deleting 1000 empty files	0.256 ± 0.005	0.257 ± 0.004	0.256 ± 0.013	sec
Write 1M into file	93.5 ± 1.3	92.8 ± 1.3	96 ± 2	M/sec
Read 1M from file	330 ± 40	340 ± 20	327 ± 16	sM/sec
SD card performance				
Creating 250 empty files	1.69 ± 0.08	1.74 ± 0.09	1.79 ± 0.07	sec
Deleting 250 empty files	0.57 ± 0.06	0.52 ± 0.05	0.49 ± 0.02	sec
Write 1M into file	26.2 ± 0.6	25.55 ± 1.13	26.6 ± 0.8	M/sec
Read 1M from file	93 ± 5	92 ± 5	57 ± 2	sM/sec
Battery				
Battery Capacity	3223		3210	mAh
Battery Rating	7194		7227	

7. PROPOSALS FOR THE SECURITY POLICY

Based on the previously described situation involving a large number of peculiarities of the mobile operating systems use, a separate security policy that regulates the user's work with mobile communication devices has to be developed.

Proposed security policy should contain requirements for mobile operating systems and rules for using smartphones, which would ensure full protection of the information stored, processed and transmitted by such devices.

The following basic requirements for mobile operating systems should be allocated:

— User authentication when accessing the mobile device should be carried out by

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

alphanumeric password of at least 6 characters.

— Identification and authentication of mobile devices when accessing to the framework.

— Introduction of discretionary and mandatory data access model:

- Possibility to set up applications access rights to the device functions;
- Prohibition of running design mode with elevated access to device features.

— Introduction of a logging subsystem which would register actions and events initiated by both user and software.

— Prohibition of software installation that has not been subjected to the UDF (undocumented features) revealing test.

— Cryptographic subsystem introduction to ensure safe data storage in device memory.

— Reorganization of the integrity check subsystem by introducing requirement of dynamic integrity checks during operation of the device along with the control of integrity when starting OS.

It is also necessary to introduce a number of measures intended to solve the problem:

— Mobile device assembling must be carried out in a controlled area using certified means hardware control to ensure absence of UDF.

— Mobile operating system should be installed on the device that passed the test for the absence of UDF.

— An integrity check according to the specifications must be carried out after the operating system installation.

— Mobile operating system initialization should be carried out according to the specifications.

If the requirements for mobile operating systems and activities to work with them are successfully introduced, mobile users will need only to observe the following rules for data security:

— Do not transmit personal mobile devices to third parties.

— Do not use the device for negotiation or other events including exchange of sensitive information.

— Use the device in accordance with the user manual.

8. CONCLUSION

The proposed trusted operating system ROMOS was developed particularly for mobile devices taking into account their peculiarities. The OS allows secure mobile communication, data processing, Internet connection, etc. ROMOS has shown good performance during initial testing.

The testing of the operating system showed its stable performance and high efficiency in comparison to existing analogues.

The provided in the paper trusted operating system for mobile devices can be used to improve existing mobile security, eliminating the vulnerabilities and ensuring more thorough protection from attackers.

The system development, additional testing and simulations are now underway to improve the functionality and performance of the trusted OS, significantly enhancing mobile device security.

In the future the trusted OS on the basis of other popular mobile platforms will be developed.

REFERENCES

[1] Kataria, Ankur; Anjali, Tricha; Venkat, Raghu. Quantifying smartphone vulnerabilities. International Conference on Signal Processing and Integrated Networks (SPIN), 2014. Pages: 645 – 649.

[2] Grimes, G.A. Are Apple's security measures sufficient to protect its mobile devices? Wireless Telecommunications Symposium (WTS), 2012. Pages: 1 – 7.

[3] You Joung Ham; Hyung-Woo Lee; Jae Deok Lim; Jeong Nyeo Kim. DroidVulMon – Android Based Mobile Device Vulnerability Analysis and Monitoring System. Seventh International Conference on Next Generation Mobile Apps, Services and Technologies (NGMAST), 2013. Pages: 26 – 31.

[4] La Polla, M.; Martinelli, F.; Sgandurra, D. A Survey on Security for Mobile Devices. Communications Surveys & Tutorials, IEEE (Volume: 15, Issue: 1), 2013. Pages: 446 – 471.

[5] Abolfazli, S.; Sanaei, Z.; Ahmed, E.; Gani, A.; Buyya, R. Cloud-Based Augmentation for Mobile Devices: Motivation, Taxonomies, and Open Challenges. Communications Surveys & Tutorials, IEEE (Volume: 16, Issue: 1). Pages: 337 – 368.

[6] Anne, V.P.K.; Rao, J.V.; Kurra, R.R. Enforcing the security within mobile devices using clouds and its infrastructure. CSI Sixth International Conference on Software Engineering (CONSEG), 2012. Pages: 1 – 4.

[7] Tae Oh; Stackpole, B.; Cummins, E.; Gonzalez, C.; Ramachandran, R.; Shinyoung Lim. Best security practices for android, blackberry, and iOS. First IEEE Workshop on Enabling

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Technologies for Smartphone and Internet of Things (ETSIoT), 2012. Pages: 42 – 47.

[8] Ghallali, M.; Ouahidi, B.E. Security of mobile phones: Prevention methods for the spread of malware. 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012. Pages: 648 – 651.

[9] Tao Li; Aiqun Hu. A private-data protection mechanism for trusted mobile platform. 7th International ICST Conference on Communications and Networking in China (CHINACOM), 2012. Pages: 222 – 226.

[10] Hanspach, M.; Keller, J. In Guards We Trust: Security and Privacy in Operating Systems Revisited. International Conference on Social Computing (SocialCom), 2013. Pages: 578 – 585.

[11] Isa, M.A.M.; Manan, J.A.; Hashim, H.; Mahmod, R.; Hamzah, M.M.A.M. Trusted Real Time Operating System: Identifying its characteristics. IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE), 2012. Pages: 83 – 88.

[12] Xiao-Wei Nie; Deng-Guo Feng; Jian-Jun Che; Xin-Pu Wang. Design and Implementation of Security Operating System Based on Trusted Computing. International Conference on Machine Learning and Cybernetics, 2006. Pages: 2776 – 2781.

[13] Li Hongjuan; Lan Yuqing. A Design of Trusted Operating System Based on Linux. International Conference on Electrical and Control Engineering (ICECE), 2010. Pages: 4598 – 4601.

[14] Mikhaylov D., Zhukov I., Starikovskiy A. Trends in the development of methods of fraud using mobile data services GPRS. // Defense equipment issues. Scientific and technical collection. Series 3. Economics, organization and management in the defense industry. Systems analysis and information technology in management and decision-making. 2012 - Vol. 3 (370) Restricted publication. – pp.47-50. For internal use only. Copie №11.

[15] Pikhtulov A., Mikhailov D., Bluetooth-attack using vulnerable channels. // Proceedings of the IV International scientific and practical conference "Prospects of information technologies development", Novosibirsk, 2011, p.272, p.253.

[16] Mikhaylov D., Zuykov A., Zhukov I., Beltov A., Starikovskiy A., Froimson M., Tolstaya A. Review of vulnerabilities of mobile devices Apple and Google. Scientific and Technical Journal "Special equipment and communication" №6, 2011. Moscow, 2011. Pages 38-40.

[17] Mikhaylov D.M. The concept of creating a trusted operating system for mobile devices. Defense equipment issues. Scientific and technical collection. Series 3. – Vol. 6(379). Moscow: FSUE "CNII EISU", 2013. Pages: 54 – 63.

[18] Desktop Top Browser Share Trend. [Electronic resource] - November, 2012 to September, 2013 - Mode of access: http://www.netmarketshare.com.

[19] Nicolás Montés. The most popular operating systems for 2014. Universidad Ceu Cardenal Herrera. Universidad Católica y Privada Valencia, Alicante (Elche) y Castellón. 12 Febrero 2014. URL: http://blog.uchceu.es/informatica/top-of-most-used-operating-systems-for-2014.

[20] Android operating system architecture. Android-shark, 2011. URL: http://android-shark.ru/arhitektura-operatsionnoy-sistemyi-android.