iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IV **Month of publication:** April 2023

DOI: https://doi.org/10.22214/ijraset.2023.50071

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

Comprehensive Study on Android Kernel

Shubham Das[1], Mikhal John[2], Ganesh Rathod[3], Pravin Gajghate[4], Siddhesh Dongare[5]
Department of Information Technology, JD College of Engineering, Nagpur, Maharashtra, India

Abstract: ***A kernel is used to manage the entire operating system. The purpose of this study was to build a kernel from scratch which works perfectly with the Android operating system. The methods we use here require additional compilation tools like clang and gcc, anykernel3 and a kernel manager. In our study we found that the kernel works fine with Android with all the necessary features working as intended. It has all the stock CPU and GPU frequencies by Qualcomm working with the latest security patches from Linus Torvalds. We have also merged the latest patches from Qualcomm for better stability and performance.***

Index Terms: anykernel3, Android, Kernel.

I. INTRODUCTION

Linux is a clone of the operating system Unix, written from scratch by Linus Torvalds with assistance from a loosely-knit team of hackers across the Net. It aims towards POSIX and Single UNIX Specification compliance. It has all the features you would expect in a modern fully-fledged Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management, and multistack networking including IPv4 and IPv6. It is distributed under the GNU General Public License - see the accompanying COPYING file for more details.

Linux was initially developed by Linus Torvalds in 1991 as an operating system for IBM-compatible personal computers based on the Intel 80386 microprocessor. Linus remains deeply involved with improving Linux, keeping it up to date with various hardware developments and coordinating the activity of hundreds of Linux developers around the world. Over the years, developers have worked to make Linux available on other architectures

The Android kernel is based on an upstream Linux Long Term Supported (LTS) kernel. At Google, LTS kernels are combined with Android-specific patches to form what are known as Android Common Kernels (ACKs).

Newer ACKs (version 5.4 and above) are also known as GKI kernels as they support the separation of hardware-agnostic Generic Core Kernel code and hardware-specific GKI modules. The GKI kernel interacts with hardware-specific vendor modules containing system on a chip (SoC) and board-specific code. The interaction between the GKI kernel and vendor modules is enabled by the Kernel Module Interface (KMI) consisting of symbol lists identifying the functions and global data required by vendor modules.

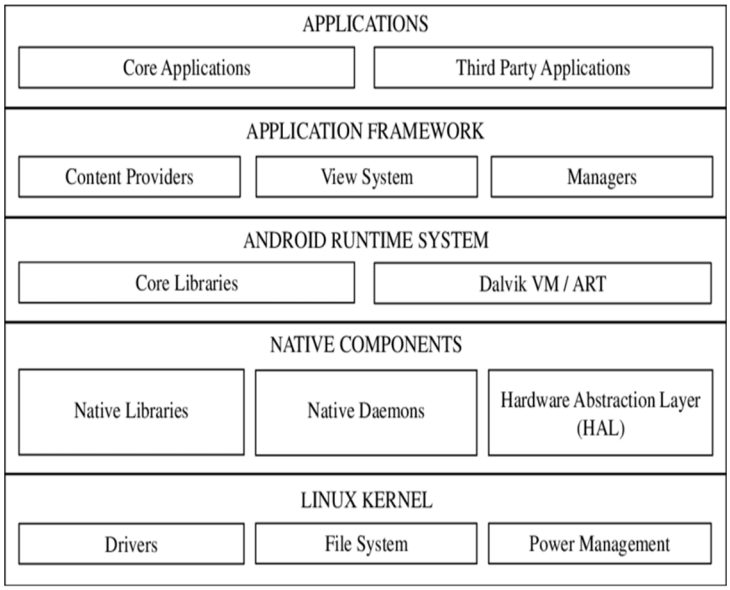


Figure 1- Architecture of Android Kernel

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

II. LITERATURE REVIEW

An OS is a set of programs that manage a computer's resources and provide services efficiently to all application processes. At the heart of an OS is the kernel. It is the first process loaded at boot time, and it remains in continuous use for the duration of the session[1].

The kernel rootkit is a common way to compromise OS kernels (i.e., malicious kernel modules). When rootkits are loaded into kernel space, they can perform arbitrary high-privileged malicious operations. In recent years, they concentrated on defeating kernel rootkits by offering several approaches. However, some restrictions apply to modern methods: 1) most approaches concentrate on rootkit-detection in user-mode; 2) some tools are only used to find obscure kernel modules[2].

The bottom layer of the Android operating system is the Linux kernel layer. This layer does not interact directly with the developer. This layer contains many important hardware device drivers and is responsible for power management, memory management, device management and resource access[3].

III. RESEARCH METHODOLOGY

A. Android Kernel File System Support

[4]While there are a large number of file systems with implementations in the Linux Kernel, many haven't been approved for production use in Android, and are not supported in Android.

Android's test infrastructure, OTA mechanisms, update processes, and privacy requirements depend on specific file system support. Not all file systems are suitable for use on Android devices.

For example, Android requires support for file-based encryption through fscrypt and file-based authentication through fsverity, hence file systems that don't support fscrypt or fsverity aren't suitable for production use.

The Android kernel team continues taking fixes for all file systems by snapping to upstream Long Term Support (LTS) kernels. However, the following file systems receive more frequent patches through the

Android security patch release schedule:

1) exfat (supported in kernel 5.10 and later)
2) ext4
3) f2fs
4) Vfat
5) EROFS

B. Boot Time Optimization

Use LZ4 compression for kernel and ramdisk-

Gzip generates a smaller compressed output compared to LZ4, but LZ4 decompresses faster than Gzip. For the kernel and modules, the absolute storage size reduction from using Gzip isn't that significant compared to the decompression time benefit of LZ4.

Switching to LZ4 should give 500ms to 1000ms faster boot time.[5]

C. The Generic Kernel Image (GKI)

Android Common Kernels (ACKs) are the basis for all Android product kernels. Vendor and device kernels are downstream of ACKs. Vendors add support for SoCs and peripheral devices by modifying the kernel source code and adding device drivers. These modifications can be extensive to the point that as much as 50% of the code running on a device is out-of-tree code and isn't from upstream Linux or from AOSP common kernels.

Thus, a device kernel is comprised of-

1) *Upstream:* The Linux kernel from kernel.org
2) *AOSP:* Additional Android-specific patches from AOSP common kernels
3) *Vendor:* SoC and peripheral enablement and optimization patches from vendors
4) *OEM/Device:* Additional device drivers and customizations.[6]

D. Android Common Kernels

AOSP common kernels (also known as the Android common kernels or ACKs) are downstream of kernel.org kernels and include patches of interest to the Android community that haven't been merged into mainline or Long Term Supported (LTS) kernels. These patches can include:

©IJRASET: All Rights are Reserved

1) Backports and cherry-picks of upstream functionality needed for Android features.
2) Features ready for Android devices but still under development upstream (for example, Energy Aware Scheduler task placement optimizations).
3) Vendor/OEM features that are useful for other ecosystem partners (for example, sdcardfs).[7]

E. Stable Kernel Releases & Updates

The Linux kernel stable release model started in 2005, when it was determined that the existing kernel development model (a new release every 2-3 months) was not meeting the needs of most users. Users wanted bug fixes made during those 2-3 months, and Linux distributions found it difficult to keep kernels up to date without feedback from the kernel community. In general, attempts to keep individual kernels secure and with the latest bug fixes was a large and confusing effort by lots of different individuals.[8]

F. Kernel Hardening

Android 8.0 added kernel hardening features to help mitigate kernel vulnerabilities and find bugs in kernel drivers. The features are in kernel/common in branches android-3.18, android-4.4, and android-4.9.[9]

G. Linux-stable Merges

Every day a significant number of changes are committed to the upstream Linux kernel. These changes are generally not evaluated for security impact, but many of them have the potential to impact the security of the kernel. Evaluating each of these changes for security impact is an expensive and likely unfeasible operation. Instead a more sustainable and maintainable approach is to regularly sync changes with the upstream Linux kernel.

It is recommended to regularly update devices with newer Long Term Supported (LTS) kernels. Regular LTS updates can help address potential unrecognized security vulnerabilities, such as this Project Zero report from earlier in 2019, prior to public disclosure or discovery by malicious actors.[10]

H. EROFS

EROFS is a read-only filesystem introduced in Linux 4.19. It supports compression and deduplication, and is optimized for read performance.

The primary difference between EROFS and other compressed file systems is that it supports in-place decompression. Compressed data is stored at the end of blocks, so that it can be decomposed into the same page. In an EROFS image, more than 99% of blocks are able to use this scheme, thus eliminating the need to allocate extra pages during read operations.

EROFS images do not have to be compressed. When using compression, however, images are around 25% smaller on average. At the highest levels of compression, images can be up to 45% smaller.[11]

IV. CONCLUSION

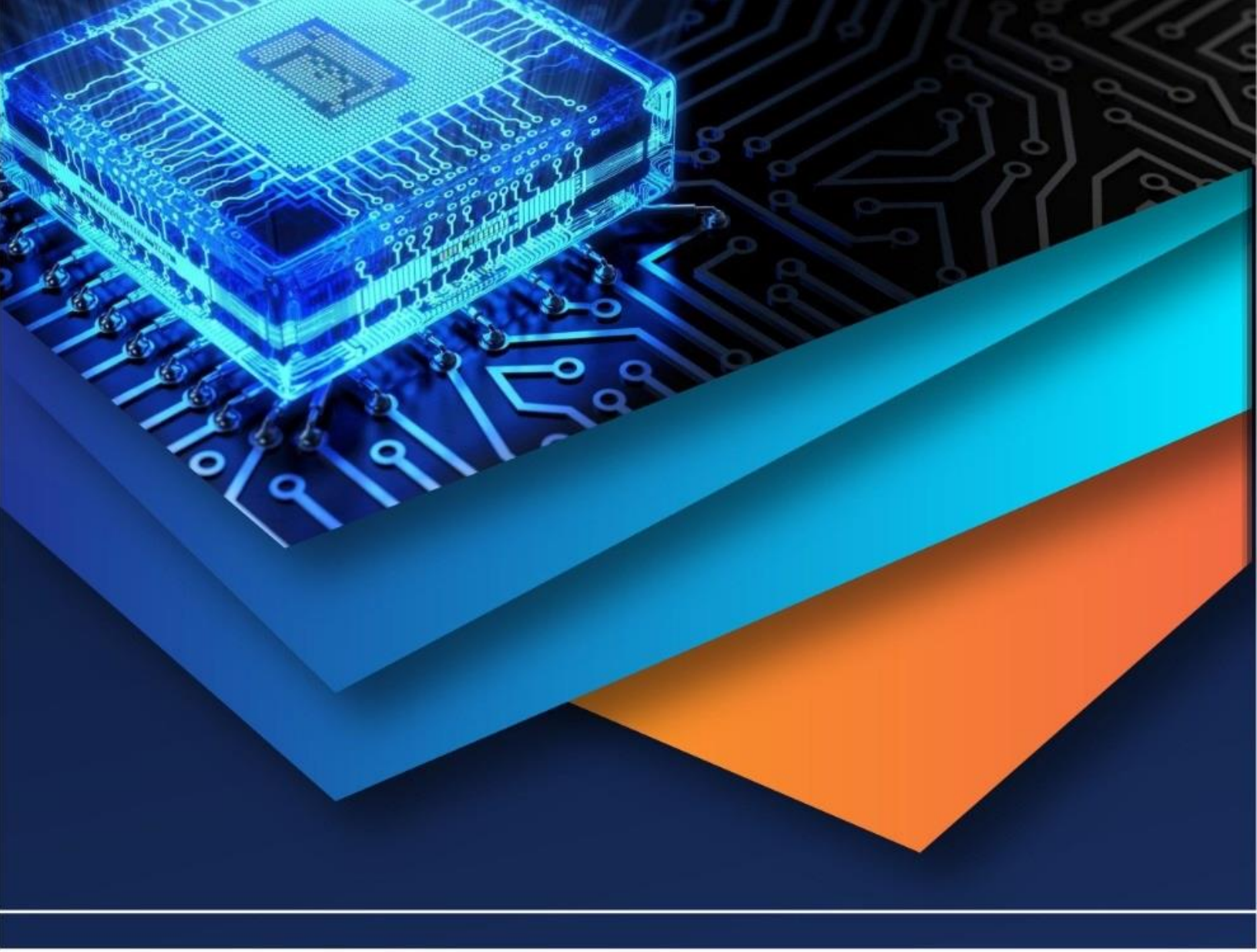
By implementing all the necessary features to a Linux Kernel we can get a faster, more optimized and more secure Android phone with better battery life. We can also see a lot of upstreams in the backend of the kernel regarding critical security and bug fixes.The majority of studies focus on improving the performance of the OS kernel in a variety of areas, including attack, malware, and intrusion prevention and detection. This also improved their performance and security features, while the company operating system lacks design versatility. This will not stress the fact that every OS should be desired or accepted, but users' preference depends on the services needed.

REFERENCES

[1] https://en.wikipedia.org/wiki/Kernel_(operating_system)
[2] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, AlexHo, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and theart of virtualization. In SOSP ’03: Proceedings of the nineteenth ACMsymposium on Operating systems principles, pages 164–177, New York,NY, 2003. ACM.
[3] Michael Brengel, Michael Backes, and Christian Rossow. Detectinghardware -assisted virtualization. In Proceedings of the 13th InternationalConference on Detection of Intrusions and Malware, and VulnerabilityAssessment, pages 207–227, 2016.
[4] Martim Carbone, Weidong Cui, Long Lu, Wenke Lee, Marcus Peinado,and Xuxian Jiang. Mapping kernel objects to enable systematic integritychecking. In Proceedings of the 16th ACM Conference on Computer andCommunications Security, CCS ’09, pages 555–565, 2009.
[5] https://www.geeksforgeeks.org/what-is-dvmdalvik-virtual-machine/
[6] Narmatha, M., and S. Venkata KrishnaKumar. "Study the Android operating system and its versions." International Journal of Scientific Engineering and Applied Science 2.2 (2016): 439-444.

[7] Shukla, Harshverdhan. "A Survey Paper on Android Operating System." Journal of the Gujarat Research Society 21.5 (2019): 299-305.
[8] Wukkadada, Bharati, Ramith Nambiar, and Amala Nair. "Mobile operating system: Analysis and comparison of Android and iOS." International Journal of Computing and Technology 2.7 (2015): 273-276.
[9] S. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions", Neural Netw., vol. 12, no. 6, pp. 783-789, 1999.
[10] A. Ruiz and P. E. Lopez-de Teruel, "Nonlinear kernel-based statistical pattern analysis", IEEE Trans. Neural Netw., vol. 12, no. 1, pp. 16-32, Jan. 2001.
[11] C. Blake, E. Keogh and C. J. Merz, UCI Repository of Machine Learning Databases, 1998

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)