DIFFERENT TYPES OF MOBILE OPERATING SYSTEMS

-sk.Reshma(intern at cdac)

WHAT IS MOBILE OPERATING SYSTEM

“Mobile operating systems (Mobile OS) manage mobile gadgets like phones and tablets. These systems run apps. They are not like desktops as mobiles have different needs. **An operating system (OS) for mobile devices manages the basic functions. It runs apps, controls memory, and connects to networks.** These systems provide an easy-to-use interface.”

“They're designed for **smartphones, tablets, and wearable tech**. Mobile OSes let you multitask, browse the web and download apps. Popular options include iOS from Apple, Android from Google, and Huawei's HarmonyOS. Mobile operating systems power modern gadgets. **With them, phones become portable computers, communication tools, and entertainment hubs.”**

KEY FEATURES OF MOBILE OPERATING SYSTEM

1.User Interface (UI): Touch inputs of Graphical User Interface (GUI) provided by mobile OS are optimized. This is where users can use touch gestures, in other words, swiping, tapping, and pinching, to interact with their gadgets.

2. Multitasking: It helps in running of many apps at the same time but what is more we can quickly switch between them without any hindrance. Such offloading is for applications that are not currently used actively.

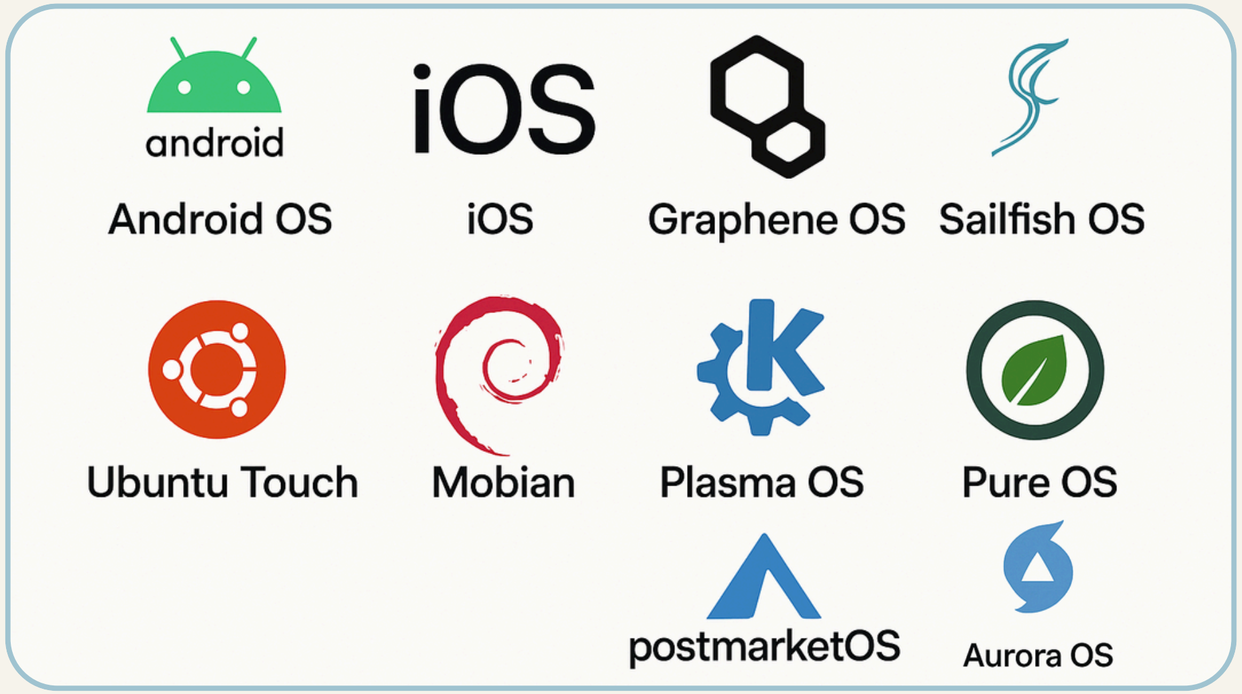
3. Connectivity: It provides a variety of connections such as cellular, Wi-Fi, Bluetooth, NFC (Near Field Communication) and others to facilitate the communication of the device with other devices and networks.

4. Application Management: Is a platform that has its own app marketplace or store which the users utilize to browse, install, run and updates the applications exclusively for that platform.

5. Resource Management: Efficiently allocates hardware resources like the CPU ,ram , and battery by achieving a balance between performance and battery life.

DIFFERENT TYPES OF MOBILE OS'ES

The current market, there exists almost more than 100 mobile operating systems to choose between. But in what follows, we'll explore the top 10 major ones that currently rule the kingdom and make a comparison among them in several different factors.



- Android OS
- iOS
- Graphene OS
- Sailfish OS
- Aurora OS
- Ubuntu Touch
- Mobian
- Plasma Mobile
- PureOS
- postmarketOS

ANDROID OS

Android mobile OS is an ***open source mobile*** operating system powered by ***Linux*** Kernel developed by ***Google***. It's majorly programmed by ***C, C++ and Java*** and is designed primarily and specifically for touchscreen mobile devices like smartphones and tablets. Since it was released, it has been Google's champion mobile OS and always shows no sign of retreat. More importantly, it's by far the most dominant mobile operating system across the globe.

Apps, UI, and User Experience on Android

Android offers users a huge variety of apps through the Google Play Store, supporting both official and third-party applications thanks to its ***open-source nature.*** The user interface is highly flexible, allowing users to customize home screens with widgets, shortcuts, and personalized layouts. Android also excels at ***multitasking, letting multiple apps run in the background and making it easy to switch between them for a smooth and efficient experience.***

Security

Due to the lack of speedy updates, Android mobile OS is much less secure compared to the other mobile operating systems especially Apple's iOS. Indeed, Google has been being criticized for not letting Android really secure all the time.
As reports indicate, Android mobile OS is most vulnerable to malware, viruses and serious hacks such as ***Stagefright and Certifi-gate.*** No wonder it has a poor reputation for security - it's behind in the update world and is still running software that's years old.

- **Stagefright:** A set of vulnerabilities in Android’s media playback engine, allowing remote code execution via specially crafted multimedia messages.

- **Certifi-gate:** A vulnerability that enabled privilege escalation through insecure remote support plugins, potentially allowing attackers to take control of affected devices.

STAGE FRIGHT

How the Vulnerability Works

- An attacker can send a crafted media file (e.g., via MMS(MULTI MEDIA SERVICES), web browser, or app) that exploits the integer underflow, corrupting memory and allowing remote code execution without user interaction.
- The exploit can be delivered through various vectors, including MMS messages, web pages with embedded video, or any app that processes media files using the vulnerable library.
- Successful exploitation gives the attacker significant control, potentially accessing the camera, microphone, and sensitive data.

Impact and Exploitation

- The vulnerability is considered highly severe (CVSS base score: 10), with complete compromise of confidentiality, integrity, and availability possible.
- Proof-of-concept exploits have been demonstrated, and the bug was actively researched and exploited in the wild.
- Even after patches were released, a large proportion of devices remained unpatched for months or years—Zimperium reported that as of 2025, over 42% of Android devices tested were still vulnerable to CVE-2015-3864.

Example Exploit Scenario

A user receives an MMS with a malicious video file.
The Android system's media server automatically processes the file, triggering the vulnerability and allowing the attacker to execute code remotely, potentially taking control of the device without any user action

CVE-2015-3864(STAGE FRIGHT)
Integer underflow in the MPEG4Extractor::parseChunk function in MPEG4Extractor.cpp in libstagefright in mediaserver in Android before 5.1.1 LMY48M allows remote attackers to execute arbitrary code via crafted MPEG-4 data, aka internal bug 23034759. NOTE: this vulnerability exists because of an incomplete fix for CVE-2015-3824.

CERTIFI-GATE

The Certifi-gate vulnerability refers to a set of security flaws discovered in 2015 that affected remote support tools pre-installed on many Android devices. These vulnerabilities allowed privilege escalation, enabling malicious apps to misuse insecurely implemented plugins to gain system-level access without user consent.

The main CVE identifiers associated with Certifi-gate are:

Vulnerability Name	CVE Number(s)	Description
Certifi-gate	CVE-2015-2885	Lens Peek-a-View has a password of 2601hx for the backdoor admin account, a password of user for the backdoor user account, and a password of guest for the backdoor guest account.
	CVE-2015-2886	iBaby M6 allows remote attackers to obtain sensitive information, related to the ibabycloud.com service.
	CVE-2015-2887	iBaby M3S has a password of admin for the backdoor admin account.

CVE-2015-2885

A real-life example of a ***hardcoded password vulnerability in Android*** occurred with several popular apps and even some device firmware. In one documented case, researchers reverse-engineered an Android app and discovered that the ***app contained hardcoded credentials***–such as **admin or API passwords–directly in its code.**Attackers who downloaded the APK could ***easily extract these secrets using tools like apktool*** or by decompiling the app.

For instance, in 2017, a popular **Android banking app** was found to have a hardcoded API key and admin password within its codebase. Attackers who obtained the APK could extract these credentials, log in as an administrator, and access or manipulate sensitive user data. In another case, IoT device management apps for Android shipped with hardcoded default passwords for remote access, allowing attackers to take control of devices if the password was not changed.

CVE-2015-2886 & CVE-2015-2887

Here's a real story to help you understand the security risks of baby monitors like the iBaby M6:
A few years ago, security researchers discovered that the **iBaby M6S baby monitor** had serious security flaws. These flaws meant that hackers could remotely access the ***camera's video feed, download saved pictures and videos,*** and even control the camera's movement–all without the owner's permission. In one investigation, experts found that anyone with ***some technical knowledge could find the camera's device ID,*** use it to pull up personal information, and view or download private recordings stored in the cloud. This meant strangers could potentially watch families in their most private moments, like feeding or changing their babies, without anyone in the home knowing it was happening

Here is an explanation of CVE-2025-27363, the top actively exploited Android vulnerability of 2025, with a real-life example.

What it is:

CVE-2025-27363 is a high-severity vulnerability in the ***FreeType font rendering library,*** widely used in Android 13 and 14 (as well as other platforms). The flaw occurs when ***FreeType parses*** certain TrueType GX and variable font files. Due to improper handling of data types, an attacker ***can trigger an out-of-bounds write, leading to arbitrary code execution.***

How it works(example)

Imagine you receive a document, app, or website that uses a specially crafted font file. As soon as your Android device tries to display the text, the ***malicious font exploits the FreeType bug, corrupting memory and allowing the attacker to execute code on your phone***. This could happen without you clicking anything– simply opening a document or viewing a web page could be enough

impact

- Attackers could take control of your device, steal data, install malware, or spy on you.
- The vulnerability has been exploited in the wild, meaning real attacks have occurred.
- Because FreeType is used in many apps and system components, the risk is widespread.

In the news:

In the news

In March 2025, ***Facebook security researchers flagged this vulnerability as being actively exploited***. Google responded by issuing urgent patches in the May 2025 Android security update, warning users to update immediately

✅ Pros of Android

🌐 **Open Source & Customizable**
Android is based on the Linux kernel and is open source, allowing manufacturers and developers to modify the OS extensively.

🔄 **Wide Hardware Compatibility**
Runs on a variety of devices from different manufacturers (Samsung, Google Pixel, OnePlus, etc.), giving users more options.

📲 **App Ecosystem & Flexibility**
Supports millions of apps via Google Play Store and third-party sources like APKs and alternative app stores.

🧩 **Widget & Home Screen Control**
Users can place widgets, shortcuts, and customize their home screens far more flexibly than on iOS.

🔓 **Greater User Control**
Offers options like file browsing, installing apps from unknown sources, and setting default apps.

💰 **Broad Price Range**
Available on both high-end and affordable devices, making it accessible to a wider audience.

❌ Cons of Android

📱 **Fragmentation**
Many devices run different Android versions, causing inconsistencies in app behavior and security updates.

🔐 **Slower Updates & Patches**
Non-Google devices often receive updates late or not at all, affecting security and performance.

🦠 **Greater Malware Risk**
Open APK installation and lax app store policies on some platforms can expose users to malicious software.

🧯 **Varying Quality Control**
Device performance, build quality, and software stability can vary greatly across manufacturers.

📦 **Bloatware**
Some OEMs preload unnecessary apps that take up space and can't be uninstalled easily.

🔋 **Power Efficiency**
While improving, Android devices sometimes lag behind iPhones in battery optimization and resource management.

iOS

iOS is a special mobile operating system developed by ***Apple***. It is so far the most advanced mobile OS in the world and currently the strongest rival against Android in the section of the mobile OS. iOS uses a hybrid kernel called ***XNU(X is not unix)***.Coded in ***Objective-C***, this mobile OS is completely ***closed-source*** and is exclusively designed for Apple devices such as the ***iPhone, iPad and iPod Touch, etc.***

Apps, UI, and User Experience on ios

OS offers a smooth, easy-to-use interface and a rich selection of apps through the Apple App Store, making it popular among individuals and enterprises. Unlike the open-source Android, iOS is closed-source and designed exclusively for Apple devices, providing a consistent and unified user experience. Its multitasking approach differs by allowing only certain app features to run in the background, optimizing performance and resource use.

Security

Among the top mobile operating systems, ***Apple's iOS is the one which has gained most reputation for security.*** On the one hand, the tight guard that the Apple mobile OS has on its exclusive apps and the capability to roll out updates to all its devices timely bring up its ***strength in security.***
On the other hand, Apple's iOS prioritizes user privacy by encrypting personal data in its apps such as ***iMessage*** and has been offering innovative security features to its users, most notably ***Touch ID and Face ID*** in the iPhone X and later.

1. CVE-2025-31200: Memory Corruption in CoreAudio

- **Type:** Remote Code Execution
- **How it works:** Attackers craft a ***malicious audio file or stream***. When a vulnerable iOS device processes this file (for example, by playing a song or receiving a media message), the flaw in CoreAudio memory management allows the attacker to execute arbitrary code on the device.
- **Impact:** The attacker can gain control over the device, steal data, or install malware—often without any user interaction.
- **Real-world status:** Actively exploited in the wild in targeted attacks; patched in iOS 18.4.1

✅ Pros of iOS

🛡️ **Strong Security**
iOS is known for tight security, with strict app store review processes, regular updates, and effective sandboxing of apps.

📦 **Seamless Ecosystem**
Deep integration with Apple's ecosystem (Mac, iPad, Apple Watch, AirPods, etc.) enables features like Handoff, iCloud sync, and AirDrop.

📱 **Uniform User Experience Consistent**
UI/UX across all iOS devices, providing a smooth and intuitive experience for users.

⚡ **Fast & Regular Updates**
Apple directly controls hardware and software, enabling timely updates across all supported devices.

🔋 **Excellent Optimization**
Hardware and software integration leads to high performance, efficient battery usage, and long device longevity.

📷 **High-Quality Native Apps**
Default apps (Photos, Mail, Safari, etc.) are polished, stable, and well integrated with the system.

❌ Cons of iOS

🚫 **Limited Customization**
Users have minimal control over UI changes, default apps, and widgets compared to Android.

🔒 **Closed Ecosystem**
Less freedom for developers and users to modify the OS or sideload apps. Apple restricts access to the file system and advanced settings.

💰 **Expensive Hardware**
iOS runs only on Apple devices, which tend to be pricier than equivalent Android hardware.

📥 **App Store Restrictions**
Developers must comply with strict App Store guidelines and pay a 15–30% commission on app revenue.

🔧 **Limited File Management**
File sharing and management are more restricted, especially with non-Apple devices.

🧱 **Slower Innovation Adoption**
iOS tends to adopt new features (like widgets, split-screen, etc.) later than Android.

GrapheneOS

GrapheneOS is an ***open-source mobile operating system*** built on the ***Android Open Source Project (AOSP)***, designed for high security and privacy.it uses ***Monolithic Linux kernel*** (LTS versions, e.g., 6.1, 6.6, 6.12) It focuses on providing a hardened, secure environment with advanced security features, making it a top choice for users who prioritize privacy and data protection. GrapheneOS is known for its minimalistic approach and lack of Google services, which enhances its security and privacy capabilities.

Apps:

- GrapheneOS supports most Android apps, but it does not include Google apps or services by default. You can install apps from alternative app stores (like F-Droid or Aurora Store) or even install Google services in a sandbox if needed.

User Interface:

- The user interface is very similar to standard Android, making it familiar and easy to use for most people. The design is clean and simple, focusing on usability and privacy rather than flashy features or branding.

User Experience:

- GrapheneOS is fast, stable, and smooth in daily use. It emphasizes privacy and security, so some features (like app permissions and notifications about sensor access) are more transparent and customizable. The OS avoids unnecessary pre-installed apps, giving users more control and less bloat

Security

- GrapheneOS builds security into its operating system through a ***comprehensive, layered approach*** that addresses both known and unknown threats. First, it reduces the attack surface by disabling unnecessary features and code by default, so only ***essential components are active,*** minimizing opportunities for exploitation. Next, it implements advanced exploit mitigations—such as ***hardened memory allocators, stricter SELinux and seccomp-bpf policies,*** and the use of memory-safe languages and libraries—to make it much harder for attackers to exploit vulnerabilities, even if they exist.

- GrapheneOS further strengthens security through ***robust sandboxing,*** isolating apps and system components so that a compromise in one area cannot easily spread to others. The OS also enhances verified boot and anti-persistence protections, ensuring only trusted software runs and preventing attackers from downgrading or persisting malicious changes. ***Hardware-based security features, like attestation and encryption, are leveraged for device integrity and data protection.***

vulnerabilites

- The most significant vulnerabilities in 2025 for GrapheneOS (and similar systems) are kernel-level flaws related to USB handling: CVE-2024-53104, CVE-2024-53197, CVE-2024-53150, and CVE-2024-50302.

(CVE-2024-53104)	In the Linux kernel, the following vulnerability has been resolved: media: uvcvideo: Skip parsing frames of type UVC_VS_UNDEFINED in uvc_parse_format This can lead to out of bounds writes since frames of this type were not taken into account when calculating the size of the frames buffer in uvc_parse_streaming.
(CVE-2024-53197)	In the Linux kernel, the following vulnerability has been resolved: ALSA: usb-audio: Fix potential out-of-bound accesses for Extigy and Mbox devices A bogus device can provide a bNumConfigurations value that exceeds the initial value used in usb_get_configuration for allocating dev->config. This can lead to out-of-bounds accesses later, e.g. in usb_destroy_configuration
(CVE-2024-53150)	In the Linux kernel, the following vulnerability has been resolved: ALSA: usb-audio: Fix out of bounds reads when finding clock sources The current USB-audio driver code doesn't check bLength of each descriptor at traversing for clock descriptors. That is, when a device provides a bogus descriptor with a shorter bLength, the driver might hit out-of-bounds reads. For addressing it, this patch adds sanity checks to the validator functions for the clock descriptor traversal. When the descriptor length is shorter than expected, it's skipped in the loop. For the clock source and clock multiplier descriptors, we can just check bLength against the sizeof() of each descriptor type. OTOH, the clock selector descriptor of UAC2 and UAC3 has an array of bNrInPins elements and two more fields at its tail, hence those have to be checked in addition to the sizeof() check.
(CVE-2024-53302)	In the Linux kernel, the following vulnerability has been resolved: HID: core: zero-initialize the report buffer Since the report buffer is used by all kinds of drivers in various ways, let's zero-initialize it during allocation to make sure that it can't be ever used to leak kernel memory via specially-crafted report.

✅ Pros of Grapheneos

🛡️ **Strong Privacy & Security Focus**
GrapheneOS is built with enhanced security hardening, including improved memory safety, stronger sandboxing, and minimized attack surfaces.

🔐 **No Google Services by Default**
It runs without Google Play Services or proprietary apps by default, helping users avoid tracking and data collection.

🔄 **Regular Security Updates**
Maintained by a focused team with frequent patches and updates specifically targeting Android's security flaws.

🧩 **Supports Sandboxed Google Play**
Users can optionally install Google Play in a sandboxed user space, retaining compatibility with most apps while limiting Google's control.

🧬 **Open Source & Transparent**
Entirely open source, allowing independent audits and community trust in its integrity.

🚫 **Minimal Background Tracking**
No built-in telemetry or data collection, which protects user privacy out of the box.

🔐 **Advanced Security Features**
Includes features like hardened malloc, app spawning restrictions, and enhanced exec/spawn policies beyond standard Android.

❌ Cons of Grapheneos

📱 **Limited Device Support**
Officially supports only Pixel devices (currently Pixel 5 and newer), due to the need for secure boot and firmware validation.

⚙️ **Requires Manual Setup**
Installation is more technical than standard Android – suitable for advanced users.

📥 **Limited App Compatibility**
Some apps (especially those relying heavily on Google APIs like Google Maps, banking apps, or push notifications) may not work without sandboxed Google Play.

🧰 **Missing Features for Casual Users**
Lacks conveniences like Google Assistant, built-in Google backup, or auto-sync – features many everyday users rely on.

🛠️ **Smaller Community & Support Base**
Compared to Android/iOS, fewer tutorials, forums, and community help resources exist.

🧪 **Some Features May Break**
Because of its security restrictions, a few apps or features may behave unexpectedly (e.g., push notifications, location services).

SAILFISH OS

Sailfish OS is an ***open-source*** and independent mobile operating system based on ***Linux***. It's primarily targeted for smartphones. It has been an excellent alternative to the current dominant mobile operating systems.

Apps, UI and Customizability

Just like the other mobile operating systems, Sailfish mobile OS is very easy to operate and control with ***simple user interface.*** But in terms of apps, it installs very few apps and for better access to different apps, you need the support of the ***Android mobile OS***.
Just like Android, Sailfish OS is an open-source mobile OS, making it customizable to other third-party manufacturers. Hence, it has ***flexible user interfaces*** and personally-tailored home screen.

Aurora OS

Aurora OS is a ***Russian Linux-based*** mobile operating system, originally branched from ***Sailfish OS*** and developed primarily for ***business and government use.***

Apps, UI, and Customizability

Just like other mobile operating systems, Aurora OS is designed to be ***easy to operate with a straightforward*** and intuitive user interface. However, in terms of apps, Aurora OS comes with a ***limited selection of pre-installed applications,*** mainly focused on essential tools and productivity. For broader app access, users rely on Aurora's own app store, but the app ecosystem is ***smaller compared to Android or iOS.***
Aurora OS is a Linux-based mobile OS, and while it is not fully open-source like Android, it is designed for flexibility and can be ***adapted by third-party manufacturers,*** especially for enterprise and government needs. The user interface is clean and can be customized to some extent, allowing organizations or users to tailor the home screen and system settings for ***specific requirements.***

Security

Sailfish OS builds security with full device encryption, app sandboxing, secure boot, a built-in firewall, VPN support, and regular security updates. It also offers privacy by design, giving users control over their data and strong protection for both personal and enterprise use.

In 2025, the main vulnerabilities affecting Sailfish OS are largely related to underlying Linux kernel issues and some app-specific flaws include:

- **Linux Kernel Vulnerabilities:**
  Several CVEs impacted Sailfish OS due to its Linux base, such as:
  - CVE-2024-45018: Denial of service via missing initialization in flow offload.
  - CVE-2024-40961: Denial of service from NULL pointer dereference in fib6_nh_init().
  - CVE-2024-35839, CVE-2024-38608, CVE-2024-35939, CVE-2024-41066: Various denial of service vulnerabilities caused by memory management issues in the kernel.
- **OpenSSH Vulnerability:**
  - CVE-2024-6387: Remote code execution vulnerability in the OpenSSH server, relevant for devices running SSH services.
- **App-Specific Vulnerabilities:**
  - CVE-2025-24903, CVE-2025-24904: These CVEs were associated with the unofficial Whisperfish Signal client, potentially bypassing end-to-end encryption and compromising message privacy

Aurora OS builds security with full device encryption, strict app sandboxing, secure boot, a built-in firewall, GOST-standard VPN, and robust remote management tools. Regular security updates and compliance with Russian certifications ensure strong protection for enterprise and government use.

In 2025, the main vulnerabilities affecting Aurora OS are primarily due to its Linux foundation and enterprise-focused features include:

- **Linux Kernel Vulnerabilities:**
  - CVE-2025-23153: A vulnerability in the Linux kernel that could impact systems running Aurora OS.
  - CVE-2023-52926: A flaw in the Linux kernel's IORING_OP_READ, which could allow improper buffer handling.
- **X.Org and Xwayland Vulnerabilities:**
  - CVE-2025-26601, CVE-2025-26600, CVE-2025-26599, CVE-2025-26598, CVE-2025-26597: Multiple use-after-free, buffer overflow, and out-of-bounds write vulnerabilities in X.Org and Xwayland, potentially leading to privilege escalation or system crashes.
- **GRUB2 Bootloader Vulnerabilities:**
  - CVE-2025-0678, CVE-2025-0689, CVE-2025-0686, CVE-2025-0685, CVE-2025-0684, CVE-2024-45778, CVE-2024-45779, CVE-2024-45780: Integer overflows and buffer overflows in GRUB2 could allow arbitrary code execution or denial of service during boot.
- **libarchive Vulnerability:**
  - CVE-2025-1632: Null pointer dereference in libarchive, possibly leading to local denial of service.

✅ Pros of Sailfish os

🛡️ **Privacy-Focused**
Designed with privacy and user control in mind; minimal data collection and no mandatory cloud services.

🧭 **Independent from Google & Apple**
Not reliant on Google Play Services or Apple infrastructure, giving users more freedom and independence.

🤖 **Android App Compatibility**
Includes a built-in Android runtime that allows many Android apps to run smoothly on Sailfish devices.

🖐️ **Gesture-Based Interface**
Intuitive swipe and gesture navigation system that's different from traditional Android/iOS UI.

💻 **Linux-Based & Open Source Core**
Built on a Linux kernel with open-source components, appealing to developers and open-source enthusiasts.

🌐 **Multilingual & Government-Backed**
Officially supported in regions like Russia and parts of the EU, and supports numerous languages.

❌ Cons of Sailfish os

📱 **Limited Hardware Support**
Officially supports only a few devices (e.g., Sony Xperia phones); porting to others requires technical expertise.

📉 **Small App Ecosystem**
Native Sailfish apps are limited; users must rely on Android compatibility for many popular apps.

⚙️ **Complex Setup for Android Apps**
Android app compatibility works, but may require setup tweaks and lacks Play Store access unless manually installed.

🌐 **Less Mainstream Support**
Niche OS with minimal attention from major app developers and limited global user base.

🔄 **Slower Updates & Development Pace**
Smaller development team compared to iOS or Android; updates may take longer and have limited new features.

📥 **Limited Documentation & Resources**
Fewer tutorials, guides, and community forums compared to more mainstream platforms.

✅ Pros of Aurora os

🛡️ **Privacy-Oriented**
Built with strong emphasis on data sovereignty and privacy, especially for enterprise and government use.

🌐 **Russian Government Endorsed**
Officially backed and developed for state and enterprise use in Russia, often replacing foreign platforms in sensitive environments.

📴 **Independence from Google/Apple**
No dependence on Google services or Apple infrastructure, providing greater control over data and software behavior.

🖥️ **Linux-Based Architecture**
Built on open-source Linux principles, offering strong customization potential and transparency.

📲 **Android App Compatibility (partial)**
Inherited from Sailfish OS, some versions include support for Android apps via a compatibility layer.

🧭 **Gesture-Based UI**
Uses a gesture-driven interface similar to Sailfish OS, designed for one-handed navigation.

❌ Cons of Aurora os

📉 **Very Limited Device Support**
Officially runs on only a small number of sanctioned or state-approved devices; no broad commercial hardware support.

📦 **Small App Ecosystem**
Very few native apps available; limited third-party support and minimal developer interest outside official circles.

⚙️ **Limited Android App Support**
Android compatibility, if available, may be outdated or restricted, reducing access to essential apps.

🧭 **Geopolitical Constraints**
Mostly used within Russia and sometimes associated with state surveillance concerns or restricted access to foreign content.

🐢 **Slow Development Pace**
Slower to adopt new features compared to mainstream mobile OSes due to smaller development teams and limited funding.

🚫 **Not Widely Available**
Lacks global distribution; typically unavailable to average consumers outside government/enterprise channels.

ubuntu

Ubuntu Touch is a mobile operating system developed by ***UBports,*** based on the ***Linux distribution*** Ubuntu. Designed for smartphones and tablets, it aims to offer a convergence experience, meaning the same system is used across both ***mobile and desktop environments***. Ubuntu Touch is known for its ***open-source nature***, providing users with the ability to control and modify the operating system to their needs, while also focusing on security and privacy.
Key features of Ubuntu Touch include its ***clean and minimalistic user interface, extensive customization options,*** and its support for both native and web-based apps. The system is built to work with a range of devices. It provides a more desktop-like experience on mobile, including the ***ability to run desktop applications*** when connected to a larger display.

Apps, UI, and User Experience on Ubuntu Touch

Ubuntu Touch offers a ***clean, gesture-based interface*** that is intuitive and easy to use, aiming for a seamless experience across ***smartphones and tablets.*** The OS provides access to a curated selection of native and web apps through the ***OpenStore,*** though its app ecosystem is smaller compared to Android or iOS. Unlike iOS, Ubuntu Touch is open-source and designed to run on a variety of supported devices, allowing for greater flexibility and community-driven development. Its multitasking approach enables ***real-time switching between apps using edge gestures,*** and its "Scopes" feature provides unified views for content like ***news, music, and contacts, enhancing productivity and personalization.***

Certainly! Here are the security features of Ubuntu Touch :

AppArmor:
Restricts apps to their own files and resources, minimizing the attack surface.

VPN Integration:
Built-in support for VPN protocols to encrypt internet connections and protect user data.

Periodic Security Updates:
Regular security and performance updates from the Ubuntu community to keep devices safe.

Read-only File System:
The OS runs on a read-only file system, enhancing both stability and security.

Full Disk Encryption (FDE):
Encrypts the entire storage device, protecting data at rest.

Default Secure Configuration:
Secure out-of-the-box with most network ports closed and a firewall enabled.

Vulnerabilities and mitigation steps for Ubuntu Touch in 2025:

Examples of 2025 Vulnerabilities Affecting Ubuntu Touch

- CVE-2025-3887:
  Stack-based buffer overflow in GStreamer's H265 codec parsing (remote code execution risk).
- CVE-2025-4664:
  Insufficient policy enforcement in Google Chrome's Loader (potential cross-origin data leaks).
- CVE-2025-32913:
  Null pointer dereference in libsoup (can cause application/system crashes).
- CVE-2025-31115:
  Bug in XZ Utils multithreaded decoder (risk of crashes and memory corruption).
- CVE-2025-2784:
  Heap buffer over-read in libsoup's skip_insight_whitespace() (potential data leakage/crashes).
- CVE-2025-30691:
  Vulnerability in Oracle Java SE (allows unauthorized data access).

Mitigation Steps

1. Keep Ubuntu Touch and all installed software up-to-date.
2. Use Ubuntu Pro for early access to critical security fixes.
3. Avoid installing untrusted software or using unverified sources.
4. Follow secure coding practices when developing custom apps for Ubuntu touch.

✅ Pros of Ubuntu Touch os

🛡️ **Privacy & Open Source**
100% open-source, with no built-in trackers or data harvesting – ideal for privacy-conscious users.

🖥️ **Convergence (Desktop-Mobile Hybrid)**
Devices can transform into full desktops when connected to a monitor, keyboard, and mouse – similar to a lightweight Linux desktop.

📱 **Google-Free Ecosystem**
No Google Play Services, offering users full control over their data and app choices.

🌐 **Active Community Development**
Maintained by the UBports Foundation with strong community involvement and transparency.

🧰 **Customization & Root Access**
Based on Ubuntu, allowing advanced users to customize the system deeply or use terminal tools.

📦 **Open App Store (OpenStore)**
Users can publish and install apps freely without heavy restrictions or gatekeeping.

❌ Cons of Ubuntu Touch os

📉 **Limited App Ecosystem**
Fewer native apps and limited support for popular Android/iOS apps – no official Android compatibility.

📱 **Limited Hardware Support**
Officially supports only a few devices (e.g., Fairphone, PinePhone, select Nexus models); community ports exist but may lack full functionality.

⚙️ **Incomplete Feature Set**
Some features (e.g., camera, sensors, fingerprint reader, VoLTE) may not work reliably across all supported devices.

🐢 **Slower Development Cycle**
Maintained by volunteers, so updates and new features arrive more slowly than on major commercial OSes.

🧪 **App Compatibility Challenges**
Apps built for Android or iOS may not be portable without full rewrites due to different APIs and architecture.

💼 **Not Ideal for Average Users**
Ubuntu Touch requires more technical knowledge for setup, maintenance, and daily use – not as beginner-friendly as Android or iOS.

mobian

Mobian is a Debian-based mobile operating system designed for smartphones. It focuses on ***privacy, simplicity, and open-source principles.*** It uses the ***GNOME desktop environment adapted for*** mobile devices, providing a consistent and familiar interface for ***Linux users***. Mobian is designed to run on a variety of smartphones, allowing users to repurpose older hardware while maintaining a lightweight and secure OS.
Key features of Mobian include its full integration with Debian's vast repositories, providing access to a wide range of software. It also offers a ***straightforward, clean user interface and strong privacy features,*** making it an appealing choice for Linux enthusiasts. Mobian is still in active development, and its user experience continues to ***improve as the project matures.***

Apps,UI &User Experience:

Mobian is a Debian-based Linux distribution for mobile devices, using the Phosh graphical shell to deliver a ***GNOME-like, touch-friendly interface.*** It supports a wide range of Debian and GNOME apps, though not all are optimized for small screens. Key features include the ***Squeekboard on-screen keyboard,*** a dedicated Mobile Settings app, and support for web app launchers. The user experience is familiar for ***GNOME users,*** focusing on touch interaction and easy access to the Debian ecosystem, with software management via GNOME Software or apt. Customization is minimal, ***aiming for a straightforward, integrated mobile Debian experience.***

Security

Mobian's security model builds on Debian's trusted foundations, inheriting its robust package management, regular security updates, and active community vulnerability patching. By targeting the mainline Linux kernel, Mobian benefits from frequent upstream security fixes. Its open-source nature encourages community security audits, and its package management system helps maintain software integrity and safety.

vulnerabilities

In 2025, Mobian's security is primarily challenged by vulnerabilities ***inherited from its Debian base, the Linux kernel,*** and widely used open-source components. Notable threats and CVEs include:

- **Linux Kernel and Driver Vulnerabilities:**
  Mobian devices using Qualcomm, Arm (Mali), or Imagination Technologies (PowerVR) hardware may be impacted by kernel-level and device driver flaws.
  - Examples:
    - CVE-2024-45580: Kernel vulnerability affecting Qualcomm-based devices1.
    - CVE-2025-0072, CVE-2025-0427: High-severity vulnerabilities in Mali GPU drivers1.
    - CVE-2024-12577, CVE-2024-46974, CVE-2024-46975, CVE-2024-47891, CVE-2024-47896, CVE-2024-47900, CVE-2024-52939: Multiple PowerVR GPU vulnerabilities1.

- **Third-Party Component Vulnerabilities:**
  Vulnerabilities in widely used libraries or services can affect Mobian if those packages are installed.
  - Examples:
    - CVE-2025-27363: Critical remote code execution in Android, also relevant to Linux-based systems using affected components.
    - CVE-2025-4427, CVE-2025-4428: Authentication bypass and code injection in Ivanti EPMM (if such enterprise tools are deployed on Mobian devices).
- **Application and Library Vulnerabilities:**
  As Mobian uses Debian's repositories, any vulnerabilities in Debian-packaged apps or libraries can affect Mobian users if not promptly patched.

Summary

Mobian's main threats in 2025 are kernel and driver vulnerabilities (especially on Qualcomm, Mali, and PowerVR hardware), flaws in open-source libraries, and inherited Debian package vulnerabilities. Timely security updates and patch management remain critical for mitigating these risks

✅ Pros of Mobian os

🐧 **Based on Debian Linux**
Built directly on Debian, one of the most stable and well-supported Linux distributions. Benefits from Debian's vast package ecosystem.

🛡️ **Open Source & Privacy-Friendly**
Fully open-source and privacy-respecting, with no proprietary services or trackers by default.

💻 **True Linux Desktop on Mobile**
Offers a full Linux environment with access to a standard terminal, Linux tools, and desktop apps optimized for mobile use.

📦 **Rich Package Availability**
Can install thousands of applications from Debian's repositories with APT.

🖥️ **Convergence-Ready**
Potential to use the phone as a desktop when connected to peripherals – ideal for minimalist computing setups.

🌐 **Active Community**
Backed by a growing group of developers and enthusiasts focused on making Linux mobile viable.

❌ Cons of Mobian os

📱 **Limited Device Support**
Primarily targets Linux phones like the PinePhone and PinePhone Pro; other devices may require custom porting.

📉 **Few Mobile-Optimized Apps**
Most Debian apps are not optimized for touchscreens, leading to a clunky experience on small displays.

⚙️ **Not Feature-Complete for Phones**
Lacks full support for some phone features (e.g., cameras, mobile data, GPS, VoLTE) depending on the device.

🧪 **Experimental/Non-Polished Experience**
Mobian is still in development; users may face bugs, instability, and missing features compared to Android or iOS.

🐢 **Slower Development**
Progress can be gradual due to the complexity of adapting desktop Linux to mobile hardware.

🧰 **Requires Technical Know-How**
Installation, configuration, and troubleshooting may be challenging for non-technical users.

Plasma Mobile

Plasma Mobile is an ***open-source*** mobile operating system built on the ***KDE Plasma desktop environment***. It is designed for users who ***value privacy, security, and customization***. Plasma Mobile features a highly flexible and user-friendly interface, deep integration with the KDE ecosystem, and support for a wide range of ***Linux applications.*** The OS allows users to personalize their experience extensively, from themes to app layouts, and benefits from active community development. Plasma Mobile aims to provide a ***modern, open alternative to mainstream mobile*** operating systems, focusing on user control, transparency, and compatibility with both ***touch and traditional Linux*** desktop workflows.

PureOS

PureOS is a ***privacy-focused, open-source operating system developed by Purism***. Based on Debian, PureOS is designed to put user privacy and security at the forefront, making it the default OS for Purism's Librem laptops and the ***Librem 5 smartphone.*** It features only free/libre and open-source software, ensuring transparency and user control over the system.

PureOS integrates privacy-enhancing technologies such as ***DuckDuckGo as the default search engine, HTTPS*** Everywhere, and privacy-respecting applications. It uses ***GNOME as its primary desktop*** environment (with support for other environments) and offers a clean, user-friendly interface. On mobile devices, PureOS is optimized ***for touch interaction and convergence,*** allowing seamless switching between mobile and desktop modes.

Security

Privacy & Security are the key prorities for plasma mobile os, & Operating System includes a range of features to protect user data. Plasma Mobile ensures security through ***end-to-end encryption, secure boot, regular security updates, and user-controlled privacy settings***. It leverages app sandboxing (where supported), strong cryptographic protocols, and the transparency of open-source development to ***protect user data*** and system integrity.

PureOS provides ***strong security by default through full disk encryption, app sandboxing,*** and a strict open-source policy that excludes proprietary code and firmware. It includes a preconfigured firewall, regular security updates, and privacy-focused applications like ***PureBrowser and Tor integration.*** PureOS does not track user activity by default and supports secure communications with end-to-end encrypted messaging and PGP email. Security is further strengthened by community auditing, the use of Debian's trusted package management, and ***hardware kill switches on supported devices***

vulnerabilities

In 2025, there have been no publicly disclosed vulnerabilities specifically affecting KDE Plasma Workspace (the core of Plasma Mobile) itself. The most recent CVEs for KDE Plasma Workspace were reported in 2024, such as:

CVE-2024-36041:

Allowed local users to gain access to the session manager and potentially execute arbitrary code via session-restore features.

CVE-2024-1433:

Path traversal vulnerability in the Theme File Handler, requiring either write access to the user's home or installation of third-party global themes.

For 2025, no new CVEs have been published for Plasma Mobile or KDE Plasma Workspace1. However, general threats to Plasma Mobile may still arise from vulnerabilities in third-party apps, the Linux kernel, or device-specific drivers, but no CVEs specific to Plasma Mobile have been reported as of now. Regular updates and community monitoring remain essential for maintaining security.

In 2025, PureOS, like other Linux-based systems, is primarily exposed to vulnerabilities inherited from its open-source components and widely used applications. Notable threats and CVEs relevant to PureOS in 2025 include:

CVE-2025-24813:

- A critical vulnerability in Apache Tomcat (CVSS 9.8) that could allow remote code execution if Tomcat is installed or used on PureOS systems.

CVE-2025-1308:

- An information disclosure vulnerability in PX Backup, where sensitive information could be logged under certain conditions (CVSS 8.4). This is more relevant to enterprise deployments or if PX Backup is used on PureOS.

Google Chrome Vulnerabilities:

- If Chrome or Chromium-based browsers are installed, several high-severity vulnerabilities were reported in 2025, such as:
  - CVE-2025-1914: Out of bounds read in V8, allowing remote attackers to access memory via crafted HTML (CVSS 8.8).
  - CVE-2025-1915: Path traversal in DevTools, potentially bypassing file access restrictions (CVSS 8.1).
  - CVE-2025-1916: Use-after-free in Profiles, possibly leading to heap corruption (CVSS 8.8).
  - CVE-2025-1918: Out of bounds read in PDFium, exploitable via crafted PDFs (CVSS 8.8).

✅ Pros of Plasma Mobile os

🧩 **Built by KDE (Trusted Open Source Org)**
Developed by the KDE community, known for KDE Plasma Desktop – well-supported and transparent.

📱 **Designed for Touch from the Ground Up**
Plasma Mobile is tailored specifically for mobile devices, with a touch-friendly UI and mobile-first design.

🛡️ **100% Free and Open Source**
No proprietary components, telemetry, or data collection – ideal for privacy-conscious users.

🧠 **Linux at the Core**
Full Linux system under the hood, allowing access to desktop-class tools, terminal, and package managers like APT or pacman.

🔄 **Convergence Support**
Can transform into a desktop-like interface when docked to a monitor – supports mouse, keyboard, and windowed apps.

💻 **App Versatility**
Supports both native Linux apps (via Kirigami framework) and web apps; also compatible with Flatpak and some Android apps through third-party tools.

❌ Cons of Plasma Mobile os

📉 **Limited Hardware Support**
Officially supports devices like the PinePhone, PinePhone Pro, and select Android ports – limited performance on mainstream phones.

🧪 **Still in Development**
Not production-ready for most users; many features are experimental or incomplete.

🧱 **App Ecosystem is Sparse**
Fewer mobile-optimized native apps; most Linux desktop apps don't scale well to small screens.

🛠️ **Inconsistent Performance**
Performance and UI responsiveness can be laggy or buggy, especially on low-powered devices.

⚙️ **Basic Phone Features May Be Missing**
Phone calls, SMS, camera, GPS, and mobile data might not work reliably on all devices.

🧰 **Not User-Friendly for Beginners**
Requires technical knowledge to install, maintain, and troubleshoot. Not suitable for typical consumers without Linux experience.

✅ Pros of Pure os

🛡️ **Privacy & Security First**
Designed with privacy in mind – ships without proprietary software, tracking, or telemetry.

🧩 **100% Free & Open Source**
Endorsed by the Free Software Foundation (FSF); includes only libre (free/libre) software.

📱 **Used on Librem 5 (Linux Phone)**
Tailored for the Librem 5 smartphone, offering a consistent mobile Linux experience with convergence in mind.

💻 **Convergence Support**
Switches between mobile and desktop UI when docked; ideal for those who want one OS across devices.

📦 **Based on Debian**
Offers stability and access to a large repository of packages using APT.

🧰 **GNOME-Based Mobile Interface**
Uses GNOME with the Phosh shell (also developed by Purism), optimized for touchscreen usability.

❌ Cons of Pure os

📱 **Limited Device Support**
Designed primarily for Purism's own hardware (Librem 5, Librem laptops); not easily installable on most smartphones.

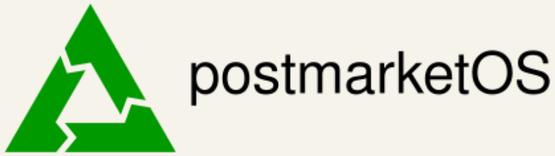
📉 **Sparse Mobile App Ecosystem**
Few native mobile apps; lacks mainstream app support like Android or iOS.

⚙️ **Incomplete Features on Phones**
Features like camera, power management, GPS, or mobile data may be partially implemented or under active development.

🐌 **Can Be Slow on Librem 5**
Due to hardware limitations, performance on the Librem 5 can lag behind Android/iOS phones.

🧪 **Still Maturing**
As a mobile OS, PureOS is a work in progress and may not be ready for daily use by average users.

🧰 **Not Beginner-Friendly**
Geared toward users comfortable with Linux, the terminal, and manual software configuration.



postmarketOS is a mobile phone operating system for phones (and other mobile devices), based on ***Alpine Linux***. Just like desktop Linux distributions, we have a package manager and a carefully crafted repository of trustworthy and ***privacy focused free software*** that will actually serve the users and not exploit them for their data. By sharing as much code as possible between various ***phone models,*** postmarketOS scales well and it becomes feasible to maintain devices even after ***OEMs(original equipment manufacturer)*** have abandoned them.

User Interface

postmarketOS offers multiple mobile-friendly UIs like Phosh, Plasma Mobile, and Sxmo, letting users choose based on preference and device.

Apps

It supports Linux apps, mobile-optimized software, web apps, and Android apps via Waydroid, with installation through GNOME Software or Plasma Discover.

User Experience

Highly customizable and privacy-focused, postmarketOS provides a familiar smartphone feel on modern UIs but varies in app optimization and hardware support.

Security / Vulnerabilities

postmarketOS builds security by using the mainline ***Linux kernel,*** enabling ***full disk encryption, and activating a default firewall.*** It delivers regular security updates and benefits from open-source transparency and community audits to quickly address vulnerabilities. Privacy is managed by ***avoiding proprietary services, minimizing tracking,*** and giving users full control over software and settings. The system also encourages the use of end-to-end encrypted communication tools to ***protect user data.***

In 2025, postmarketOS is primarily affected by vulnerabilities in widely used open-source components rather than OS-specific flaws. Notable CVEs that could impact postmarketOS if the relevant software is installed or used include:

- **CVE-2025-24813:** Critical remote code execution vulnerability in Apache Tomcat, allowing unauthenticated attackers to upload and execute malicious files on servers running vulnerable Tomcat versions.
- **CVE-2025-21893:** Linux kernel vulnerability (use-after-free in key management), which could potentially allow privilege escalation or system compromise if not patched.
- **CVE-2025-30406:** Remote code execution in CentreStack and Triofox due to hardcoded cryptographic keys, though this is more relevant to enterprise file-sharing deployments.
- **CVE-2025-22457:** Critical stack-based buffer overflow in Ivanti Connect Secure and related products, leading to possible remote code execution if such software is present.

There are no reports of vulnerabilities unique to postmarketOS itself in 2025; the main threats come from unpatched third-party applications, the Linux kernel, and server software. Regular updates and prompt patching are essential to mitigate these risks.

✅ Pros of Post Markets os

🧱 **Designed for Longevity**
Aims to make phones sustainable by supporting them long after official vendors stop providing updates.
🐧 Lightweight Alpine Linux Base
Built on Alpine Linux, making it minimal, fast, and highly customizable.
🛡️ **Privacy & Open Source**
100% open-source with no proprietary software by default; ideal for privacy-conscious users.
♻️ **Runs on Older Devices**
Designed to work on hundreds of old Android phones (some with mainline Linux kernel support).
🧰 **Multiple Desktop Environments**
Offers a choice of interfaces (Phosh, Plasma Mobile, SXMO, etc.) tailored for different use cases and devices.
💻 **Convergence-Friendly**
Some environments support a desktop-like experience when docked – turning phones into portable PCs.
🌐 **Active Community**
Developer-driven and transparent, with a strong community pushing Linux on mobile forward.

❌ Cons of Post Markets os

📱 **Limited Hardware Functionality**
Many supported devices lack full support (e.g., modem, camera, sensors); intended more for experimentation than daily use.
📉 **Not Daily-Driver Ready (for Most)**
Still in early stages for most users; lacks stability, performance, and app support compared to Android/iOS.
⚙️ **Complex Setup**
Requires Linux knowledge to install and configure, especially for unsupported or community-ported devices.
🧩 **Sparse Mobile App Ecosystem**
Very few native mobile apps; no direct compatibility with Android apps without third-party layers.
🐌 **Slower UI Performance on Older Devices**
While lightweight, some desktop environments may still lag on older phones with limited RAM or CPU power.
🔧 **Minimal Vendor Support**
No official support from major hardware manufacturers – users must rely on the community for device ports.

Feature / OS	Android OS	iOS	GrapheneOS	Sailfish OS	Ubuntu Touch	Mobian OS	Plasma Mobile	PureOS (Mobile)	postmarketOS
🟦 OS Definition	Google's open-source mobile OS with broad support	Apple's proprietary OS with high security	Hardened Android fork with privacy as core	Linux-based with Android app support and proprietary	Ubuntu-based mobile OS focused on convergence	Debian-based OS for mobile devices	KDE Plasma UI adapted for phones	FSF-endorsed Debian fork focused on freedom	Alpine Linux-based mobile OS built for longevity
🧬 Kernel Type	Modified Linux Kernel	XNU (Hybrid: Mach + BSD)	Hardened Linux (Pixel kernel)	Linux Kernel	Linux Kernel	Linux Kernel	Linux Kernel	Linux Kernel	Alpine Linux (musl libc + Hardened Linux)
💥 Common Vulnerabilities	CVE-2022-20465 (Intent Hijack), CVE-2023-20954 (Privilege escalation)	CVE-2021-30883 (Memory corruption), CVE-2023-28205 (WebKit exploit)	Fewer CVEs due to limited attack surface; based on AOSP — inherits Android CVEs but mitigated	CVE-2018-9862 (Lipstick), CVE-2020-12321 (SELinux bypass risk)	CVE-2020-11932 (QML bug), susceptible to Ubuntu base flaws	Inherits from Debian: CVE-2022-0492 (Kernel), CVE-2023-0386	Same as Mobian (Debian base), plus KDE/Qt-related	Same as Debian, CVE-2022-29500 (APT), CVE-2023-0215	CVE-2023-0386, Alpine-based CVEs (low footprint but limited scrutiny)
💻 Language Used	Java, Kotlin, C++, C, some Rust	Objective-C, Swift, C, C++	Java, Kotlin, Rust, C	C++, Qt/QML, C	C++, QML, Python	C, Shell, GTK, GNOME	C++, Qt, QML, C	C, Shell, GTK, GNOME	C, Shell, musl-libc
🛡️ Privacy	Moderate🟡	Low🟡	Very High 🟢	Moderate 🟡	High 🟢	High 🟢	High 🟢	Very High 🟢	Very High 🟢
🔐 Security	High 🟢	Very High 🟢	Very High 🟢	Medium 🟡	Medium 🟡	Medium 🟡	Medium 🟡	High 🟢	Medium 🟡

📦 App Ecosystem	Massive 📱	Massive 📱	Moderate (no GMS)	Limited + Android support	Limited 📉	Limited 📉	Limited 📉	Very Limited ⚠️	Very Limited ⚠️
🤖 Android App Support	Native	No	Native (no Google Play)	Yes (Alien Dalvik)	No	No	No	No	No
📱 Touch UI Ready	Yes	Yes	Yes	Yes	Yes	Partial	Yes	Yes	Yes
🔁 Convergence Support	Partial (Samsung DeX etc.)	Limited	No	No	Yes	Yes	Yes	Yes	Yes
🧠 Target Users	General consumers	General consumers	Privacy/security-conscious users	Enthusiasts, alternative UI seekers	Linux users & open-source advocates	Developers, advanced Linux users	KDE fans, tinkerers	Freedom-first privacy advocates	Developers, long-term device users
🔧 Hardware Support	Very Broad	Apple Devices Only	Pixel Devices Only	Xperia, limited others	Fairphone, PinePhone	PinePhone, some others	PinePhone, limited ports	Librem 5 Only	Community ports, PinePhone
🔋 Stability	High	Very High	Very High	Medium	Medium	Low–Medium	Low–Medium	Medium	Experimental
🧰 Technical Skill	Low	Low	Moderate	Moderate	Moderate	High	High	High	High

THANK YOU