Read or listen to this article online

Scan the QR code to open this article on XO.PL

Article source:

https://xo.pl/it-made-fun-sys-android

OPERATING SYSTEMS

Android: Linux in your pocket, running on a thousand different phones

An alarm echoed across the Operating Systems Fleet's spaceport. Not because anything had exploded. Three thousand ships were requesting landing clearance at once. Every one had a different color scheme, bridge layout, number of cameras and set of battery-charging rules. All belonged to the **Android** fleet, but the largest proudly carried the designation **MSS Android** — flagship of this extraordinary assortment of vessels. The captain of **USS Windows** asked for a single representative. Twenty-seven appeared. Each claimed to be the right one. The captain of **NCC Linux** was already unscrewing a maintenance panel. The captain of **HMS macOS** surveyed the confusion and quietly polished his own control panel. Finally the commander of the flagship, MSS Android, came ashore wearing the smile of a man about to explain the difference between an operating system, a phone manufacturer and an app store. 🤣

Three thousand ships, one captain and a parking problem

— WHICH ONE OF YOU IS ANDROID?! — shouted the **USS Windows** captain.
— All of us — replied the MSS Android captain.
— THEN WHY DOES THIS ONE HAVE FOUR CAMERAS, THAT ONE THREE, AND THE OTHER ONE LOOK LIKE A FOLDING NOTEBOOK?!
— Different manufacturers and models.
— THAT ONE HAS DIFFERENT MENUS!
— Manufacturer customization.
— AND THAT ONE DOESN'T HAVE THE SAME APP STORE!
— A different set of services.
— THEN HOW MANY OPERATING SYSTEMS DO YOU HAVE?!
— Many Android versions and configurations, but a recognizable common foundation.
— SO ONE SHIP?
— More like an entire fleet.
— THEN WHO'S IN CHARGE OF KEEPING THINGS ORDERLY?!
— That depends which shipyard you're asking.

Android is a family of operating systems developed primarily for mobile devices. It powers phones and tablets from many manufacturers. The shared foundations are recognizable, but devices differ in hardware, interface, features, bundled software and support policies. Android isn't the name of one phone model or one hardware brand. What you see on screen results from the combination of the operating-system platform, manufacturer software and installed apps.

— Technical question — said Greg.
— Go ahead.
— Could we find two identically configured ships?
— Certainly.
— Excellent. Where?
— In the warehouse, before the first owner switches them on.
— And after that?
— One installs forty apps; the other changes everything except the wallpaper, because they couldn't find the wallpaper setting.

A **Linux** engine room, but a very different bridge crew

The captain of **NCC Linux** entered the MSS Android engine room and beamed at a sign reading “**Linux** kernel”.
— AH! FAMILY!
— Related engines — corrected the Android captain.
— So I can install Debian and run my usual desktop?
— Not automatically just because there's a **Linux** kernel.
— My scripts?
— Depends on the environment and permissions.
— Package manager?
— We have our own application-installation system.
— SO WE'RE RELATED, BUT I DON'T HAVE THE KEY TO YOUR KITCHEN?!
— Every guest here gets a separate room. And nobody gets to rearrange the reactor.

Android uses the **Linux kernel** to manage processes, memory, drivers and fundamental isolation mechanisms. But Android isn't a conventional GNU/Linux distribution with the usual desktop and server utilities. It has its own application platform, libraries, services and device-management model. Many Android apps run using the **Android Runtime (ART)**, not as ordinary desktop **Linux** programs. The kernel is related; the rest of the ship is built differently.

— Why give the programs separate rooms? — asked the **USS Windows** captain.
— So the recipe app can't simply walk into the engine documentation room.
— BUT WHAT IF THE RECIPE IS FOR A VERY IMPORTANT CAKE?!
— It's still not engine documentation.

AOSP: the shipyard plans are available to inspect

Technical drawings were being handed out in the port control center. The **NCC Linux** captain picked up a binder.
— I can read the plans?
— A lot of them, yes. That's AOSP.
— And build my own device?
— You can technically develop a system from this base, but you still have to support the hardware, drivers and everything else.
— Will I get all the Google apps automatically?
— No.
— WHY NOT? THE PLANS ARE OPEN!
— Shipyard plans aren't a subscription to the company canteen.

The **Android Open Source Project (AOSP)** is Android's open-source code base, developed under Google's leadership with contributions from others. It enables manufacturers and communities to build their own variations. But not everything found on a popular Android phone is open source. **Google Mobile Services (GMS)**, including Google services and apps offered under the relevant arrangements, is a separate layer. An Android device doesn't have to include GMS. A variant without those services may behave differently from a phone its owner is used to.

— So you can have Android without Google Play? — asked the **Windows** captain.
— Yes.
— CAN YOU HAVE GOOGLE PLAY WITHOUT A PHONE AT ALL?!
— Captain, perhaps first decide what you want to run.
— EVERYTHING!
Greg noted on the requisition form: “Requirements: everything. Deadline: yesterday. Budget: none”.

Every shipyard has its own paint, menus and favorite buttons

The delegation toured four neighboring ships.
— Where are Settings? — asked the **Windows** captain on the first.
— Here.
— On the second?
— A little further down.
— The third?
— Expand the additional menu.
— AND THE FOURTH?!
— You could use the settings search function.
— DO THE MANUFACTURERS EVER MEET EACH OTHER?!
— They do — replied the Android captain. — Then everyone goes back to their own shipyard with new ideas.

Device manufacturers can add their own **user interfaces**, applications, camera features, battery-management policies and hardware functions. Users can often replace the **launcher**, which organizes the home screen, icons and access to apps. Doing so doesn't automatically replace the kernel or the entire operating system. That's why two phones with similar Android versions can look different and offer different features. The same setting may even be tucked away in a different menu.

— I want one “Make Them All The Same” button — demanded the **USS Windows** captain.
— And which appearance should we copy? — asked the Android captain.
— MINE!
The **HMS macOS** captain raised an eyebrow.
— I cannot endorse the method, but I admire the confidence.

The app store, the bazaar and the guard at the loading ramp

There was a shopping arcade aboard MSS Android.
— Google Play — read the **Windows** captain. — SO THAT'S WHERE YOU BUY THE ENTIRE OPERATING SYSTEM?!
— Mainly apps and other content, not the whole OS.
— AND THAT BAZAAR NEXT DOOR?
— Other app stores and software distribution.
— AND THAT MAN IN A HIGH-VIS VEST?

— Checking that nobody brings in suspicious packages.
— WHAT IF THE PACKAGE SAYS “DEFINITELY SAFE”?!
— Then he looks even more closely.

Android applications are commonly distributed as **APK** packages or through store mechanisms that deliver appropriate packages to a device. **Google Play** is a popular source, but it's not the only one. Android also supports installing apps from other sources, subject to device settings and local restrictions. Such installations require care: an unfamiliar file or fake update may be dangerous. Features such as **Google Play Protect** can help identify suspicious apps on supported devices, but they don't provide absolute protection. Our story about **IT security basics** explains how to think about such risks.

— I'd like an app that always tells the truth — said the captain.
— You can read the permissions and reviews, but nobody can guarantee that.
— IS THERE AN APP TO CHECK WHETHER OTHER APPS ARE LYING?!
— We can examine some behavior, yes.
— THEN ADD ONE TO CHECK THE APP THAT CHECKS THE APPS!
The security guard closed his booth for lunch.

The phone requests camera access. The flashlight wants the microphone. The captain wants an explanation

A notification rang out on the bridge.
— App requests location access — read the captain.
— What's the app for? — asked Greg.
— Drawing mustaches on photographs.
— Why does it need your location?
— MAYBE THE MUSTACHES ARE GEOGRAPHICAL?!
— It also wants the microphone.
— DO THE MUSTACHES TALK?!
— I have no idea, Captain. That's why we check permissions.

Android isolates apps through **sandboxing**. An app shouldn't have unrestricted access to other apps' data or the entire system. Sensitive features — such as the microphone, camera and location — often require separate permissions, with increasingly detailed controls in newer platform versions. Tapping “Allow” is a user decision: it's worth asking what the app gains and whether it really needs it. Not every permission category works the same way, and the available controls depend on the Android version.

— Take away its microphone! — declared the captain.
— Very well.
— And its camera!
— Then it won't take photographs.
— AND ITS LOCATION ACCESS!
— Sensible if it doesn't need that.
— AND ACCESS TO THE SCREEN!
— Then using it would be rather difficult.
— YOU MEAN I HAVE TO MAKE ANOTHER DECISION?!
— Yes, Captain. The unfortunate thing about security is that it requires thinking.

The update will arrive. Nobody knows which delivery service

A message flashed in the port: “UPDATE AVAILABLE”.
— EXCELLENT! — cheered the **Windows** captain. — ON EVERY SHIP AT THE SAME TIME?!
— Not always — replied the Android captain.
— WHY?!
— Different models, manufacturers, regions, sometimes carriers, different schedules.
— WILL THAT OLDER SHIP GET IT TOO?!
— Depends on its support period and compatibility.
— THEN SEND IT BY POST!
— Delivering the parcel isn't enough. It has to fit the engine and equipment.

Android development has a central platform, but actual device updates depend on the manufacturer, model, drivers and support policy, and sometimes others involved in distribution. One result is **fragmentation**: several operating-system versions and patch levels coexist. That doesn't mean all Android phones have equally poor update support. Some models have long, explicit support periods. Parts of the platform and certain services can also update separately from a full OS release, but that doesn't replace every fix that requires device firmware or an OS update.

— How long is my phone supported? — asked the **NCC Linux** captain.
— Check the manufacturer's commitment for that particular model.
— And my second phone?
— Same procedure.
— AND THE THIRD?! — interrupted the **Windows** captain.
— Same procedure.
— THAT'S THE MOST CONSISTENT PROCEDURE IN YOUR ENTIRE FLEET!
— True. And one of the most overlooked details before purchase.

Four percent battery and a declaration of martial law

Half the decorative lights aboard MSS Android went dark.
— MALFUNCTION?! — yelled the **Windows** captain.
— Power saving — explained the first officer.
— But my app was supposed to send a notification!
— The system restricted its background activity.
— MY WATCH?!
— Working.
— MY ALARM?!
— It should work according to the system and app's scheduling rules.
— MY AUTOMATED REPORT AT FIVE O'CLOCK?!
— Check the settings, restrictions and how the job was scheduled.
— THEN WHY DO WE SAVE POWER?!
— So we can reach the charger, Captain.

A phone must share resources among applications, connectivity, the screen and the battery. Android includes mechanisms that limit background activity, such as **Doze** and app-state management. Manufacturers may add their own power-saving policies. As a result, a badly scheduled or heavily restricted job may behave differently while the screen is on than after a long period of inactivity. Applications that need timely execution should use the appropriate platform APIs, while users sometimes need to check battery and notification settings.

— Increase the battery size — ordered the captain.
— It's not user-replaceable on this model.
— THEN CONNECT THE USS **WINDOWS** REACTOR!
— It's a telephone, not an orbital station.
— I NEED ANOTHER FIVE PERCENT!
— We could stop playing that 4K cat video.
— LET'S NOT TAKE ENERGY SAVING TOO FAR!

The phone must know the route along the packet highway

Fleet Command ordered a communications test.
— Internet ready? — asked the Android captain.
— Wi-Fi, cellular and **VPN** available — reported the communications officer.
— Open the star chart.
— Unable to connect to the service.
— BUT WE HAVE SIGNAL!
— Signal doesn't mean every connection is working.
— TRY THE OTHER NETWORK!
— The app still can't resolve the server name.
— SO THE WHOLE GALAXY IS BROKEN?!
Greg looked at the display.
— No. So far it's just **name resolution**.

Like other operating systems, Android uses standard networking mechanisms. A device obtains an **IP address**, may use automatic configuration through **DHCP**, resolves service names using **DNS**, and connects to servers. Communication can be protected with technologies including **HTTPS** and properly configured **VPNs**. No green signal-strength indicator guarantees that the whole path is working. When something breaks, actual diagnosis beats ritual airplane-mode toggling.

— What now? — asked the **Windows** captain.
— Find where the **packets stop** — said Greg.
— CAN'T WE JUST TURN IT OFF AND ON?!
— We could, but then we'd lose evidence about the cause.
— I DON'T WANT TO KNOW THE CAUSE! I WANT IT TO WORK!
— Which is why you ordered diagnostics.

A safe in your pocket and the captain who forgot his code

The security officer stopped the delegation at the data vault.
— Please unlock your device.
— Fingerprint? — asked the captain.
— On devices that support it.
— Face?
— On some models.
— PIN?
— Yes.
— WHAT IF I FORGET THE PIN?!

— Recovery options depend on the device setup, and data may become inaccessible.
— THEN STICK THE PIN TO THE CASE!
— Captain, that rather defeats the purpose of the lock.

Modern Android devices use protections such as **data encryption**, access controls, hardware-backed storage of certain secrets and verification of software integrity during startup, known as **Verified Boot**. Details depend on the platform version and device. Unlocking the bootloader or modifying system software — often discussed in terms of rooting or custom ROMs — can change the security model, availability of services and the way updates arrive. It's not a magical “everything is now allowed without consequences” switch. You can read more about **protecting data through encryption** in our separate article.

— So even a captain shouldn't have permanent access to everything? — asked the **HMS macOS** captain.
— Correct — replied the officer.
— Finally, a sensible policy.
— BUT I'M THE CAPTAIN! — protested the **USS Windows** commander.
— Exactly why we've added confirmation steps.

Photographs, messages and a backup that isn't a magic spell

During the final briefing, the MSS Android captain handed the visitors a new phone.
— Transfer everything from your old device.
— Everything? — asked the **Windows** captain excitedly.
— Depends on the apps, accounts, permissions and available migration methods.
— THEN WHY DOES THE BUTTON SAY “TRANSFER DATA”?!
— “Attempt to transfer supported data, then manually check the rest” didn't fit.
— MY PHOTOGRAPHS?!
— You might use a backup or synchronization.
— MY SIGN-IN CODES?!
— Those need special attention.
— MY TWO THOUSAND SEVEN HUNDRED MEMES?!
— Captain, decide which ones are mission-critical.

Android data can live in **local device storage**, individual app spaces or online services. Available backup and migration options depend on the device, OS, accounts and whether apps support particular transfer methods. **Synchronization is not always an independent backup**: an accidental deletion can spread to other devices. Important photographs, documents and recovery keys call for a deliberate plan. Even a fleet of a thousand phones needs the rules for **creating and testing backups**.

— What if the phone falls into a black hole? — asked the **Windows** captain.
— I wouldn't count on the local Recycle Bin.
— WHAT IF THERE'S A RECYCLE BIN INSIDE THE BLACK HOLE?!
Greg set down his tablet.
— Captain, that's for the astrophysics team, not the backup department.

Root access, custom systems and freedom with a warning label

The **NCC Linux** captain examined the engine-room lock.
— What if I want to rebuild this ship?
— Some models can have their software modified, but manufacturers vary.

— Can I get root access?
— Not as an ordinary user of a standard device.
— And after modifications?
— Sometimes, but there are risks.
— A CUSTOM OPERATING SYSTEM?!
— Where the hardware and project support it.
— SO I CAN TURN MY PHONE INTO A LABORATORY!
— Yes. Just don't call customer support if halfway through the experiment your phone becomes a paperweight.

Android's open foundation encourages community projects and custom operating systems. But driver availability, bootloader unlocking, component compatibility and app requirements differ among devices. Modifying the system can be a valuable learning exercise, but doesn't automatically guarantee **better security** or longer support. Some apps check device integrity and may refuse to run on modified setups. As aboard **NCC Linux**, greater freedom often carries greater maintenance responsibility.

The grand fleet inspection and the charging report

At the end, the Android ships lined up for a group photograph. They couldn't all fit in the frame. One had a folding screen, another a huge battery, another was affordable and simple, a fourth was built for harsh conditions, and a fifth was busy installing an update.
— Captain Android — said the **HMS macOS** commander — our equipment and appearance are more predictable.
— I know. Our users have an enormous range of devices and prices to choose from.
— WHICH IS BETTER?! — yelled the **Windows** captain.
— It depends on your needs — they both replied.
— I HATE THAT ANSWER!
— It's still correct — remarked the **NCC Linux** captain.

Android combines the Linux kernel with a mobile application platform and an enormous choice of devices. Its strengths include variety, availability across many kinds of hardware and the ability to match a phone to different needs. Trade-offs include differences in user interfaces and support periods, manufacturer restrictions and the complexity of device security. Before buying, it's worth looking beyond the number of camera lenses to the promised update lifetime, functions you need and how you can back up your data.

— COMPUTER! FINAL REPORT! — demanded the **USS Windows** captain.
— Android fleet inspected. Essential systems operational.
— SO WE CAN DEPART?!
— Yes. Just disconnect the devices from their chargers.
— HOW MANY?!
— Three thousand.
— WHO HAS THAT MANY CABLES?!
Greg raised his hand.
— I have adapters.
The **HMS macOS** captain looked impressed.
— This time they even match the color scheme.

---

*Operating Systems Fleet Command reports: MSS Android and the other vessels have successfully departed. One ship stayed behind because its manufacturer hid the “Disconnect Charger” option in a different menu. The **USS Windows** captain requested one common manual. The reply read: “Please specify the model”.* 🚀🤖🤣❤️