100 Linux Distros You Never Knew Existed

An Unreliable But True Guide, Part Two

With digressions where the history deserves them.

Saying there are "many" Linux distributions is a bit like saying the ocean is "fairly wet."

100 distros · 15 chapters

Compiled July 2026 · © SplitZrate 2026

Contents

1. Security, Anonymity & Escaping Surveillance 2
2. Extreme Minimalism 3
3. Built by Rebels & Schisms 4
4. State-Controlled & National Distros 5
5. Radically Different Architecture 6
6. Jokes, Satire & Passion Projects 7
7. Retro Looks & Transition Distros 8
8. Gaming & Entertainment 8
9. Containers, Cloud & the Invisible Layer 9
10. Education & Humanitarian Purpose 10
11. Forensics, Rescue & Specialized Tools 10
12. Independent & Rolling-Release Desktop Projects 11
13. Mobile & De-Googled Phones 13
14. NAS, Home Servers & Networking 14
15. How to Build Your Own Linux Distro 15

1. Security, Anonymity & Escaping Surveillance

1.1 Tails — The operating system that forgets it ever existed

Tails (The Amnesic Incognito Live System) runs entirely from a USB stick and deliberately leaves no trace on the computer it's used on — on shutdown, everything is wiped from memory as if the session never happened. All internet traffic is forced through Tor. The system had its real breakthrough into public awareness in 2013, when it was revealed that Edward Snowden used it to communicate securely with journalists during the NSA document leak — turning an obscure live distro into a household name overnight.

1.2 Qubes OS — The computer that pretends to be several computers at once

Developed by security researcher Joanna Rutkowska around the principle of "security by compartmentalization": instead of trusting that a single environment is secure, Qubes runs every activity (browser, email, banking) in its own isolated virtual machine. If one individual "qube" is compromised, the damage stays there — an architectural response to the realization that perfect security is impossible, so why not build the system to survive its own breaches.

1.3 Whonix — Two computers in one, solely to protect your IP address

Whonix consists of two separate virtual machines: a "gateway" that only talks to the Tor network, and a "workstation" that can only reach the internet through the gateway. Even if the workstation half were fully compromised, malware couldn't leak your real IP address, because architecturally there's no other way out.

1.4 Kodachi — Privacy in layer upon layer upon layer

Routes your traffic through VPN, Tor, and the I2P network simultaneously, automatically wipes DNS leaks, and leaves no trace on the host computer — a distro built by a single developer for users who find plain Tor a bit too relaxed.

1.5 Subgraph OS — A security project aimed at journalists that never quite got finished

Developed with support partly tied to the same environment that funded the Tor project, with the goal of giving journalists and activists a hardened, sandboxed desktop system. Development has come in waves and at times stalled entirely — a good example of how hard it is to keep an ambitious security project alive without commercial backing.

1.6 Discreete Linux — For anyone who takes "air gap" completely seriously

Formerly known as Ubuntu Privacy Remix. Designed to actively prevent sensitive documents from ever reaching the internet: networking is effectively disabled, and the whole idea is to run the system on a machine that deliberately never gets connected — a system design choice that's almost defiantly old-fashioned in a connected world.

1.7 Kali Linux — The pentester's standard toolbox, born from another distro

Kali succeeded the earlier distribution BackTrack in 2013 and bundles hundreds of pre-installed security tools for penetration testing. These days used just as often by IT security students as by actual professionals — and unfortunately, occasionally by someone who should probably read up on unauthorized-access law first.

1.8 Parrot Security OS — Kali's Italian, somewhat more all-around cousin

Similar in purpose to Kali but also emphasizes privacy, development, and digital forensics within the same distribution, with a somewhat lighter resource footprint — proof there's more than one way to package the same underlying toolbox.

1.9 BlackArch — Arch Linux for anyone who thinks Kali has too few tools

Built on top of Arch Linux with its own repository that by now spans several thousand security tools — probably the most tool-dense security distribution in existence, with the corresponding learning curve Arch always comes bundled with.

1.10 Pentoo — Gentoo, but with exploits instead of performance tuning

A security-focused live distribution built on Gentoo, inheriting Gentoo's philosophy of compiling and fine-tuning everything yourself — applied to penetration testing instead of pure hardware optimization.

1.11 Kali NetHunter — The pentest lab that fits in your pocket

A mobile variant of Kali Linux that runs on top of Android, making it possible to run the same kinds of security tests directly from a phone or tablet — the pentester's toolbox, compressed down to something that fits in a jeans pocket.

1.12 IprediaOS — Anonymity built on I2P instead of Tor

A Fedora-based system that forces all traffic through the lesser-known I2P network instead of Tor — an alternative anonymization network with a completely different technical design, these days remembered mostly as one of the earlier, now largely dormant, attempts to build an entire distribution around that specific technology.

---

2. Extreme Minimalism

2.1 Tiny Core Linux — An entire desktop in 16 megabytes

The core system fits in under 16 MB and can boot to a graphical desktop in seconds, with the rest of the functionality loaded as removable "extensions" you choose yourself. Proof that almost everything the modern computer "needs" is actually optional.

2.2 Damn Small Linux — Designed to fit on a business card

One of the earliest and best-known examples of extreme size minimization: the entire distribution originally fit on a 50 MB CD, small enough to burn onto a round "business card CD" format — a literal marketing gimmick that was also a genuine technical feat.

2.3 SliTaz — French, fast, and almost nobody's heard of it

A French minimalist distribution with its own package manager (`tazpkg`) that fits in a few dozen megabytes, aiming to be fully usable even on genuinely old hardware — developed by a small team without the resources larger distributions have at their disposal.

2.4 Puppy Linux — The distribution that lives entirely in RAM

Loads the entire system into memory at startup, making it blazing fast even on ancient machines and letting you remove the install media entirely without the system crashing. Has spun off dozens of variants over the years, built by different community members, all sharing the same core philosophy of extreme reuse of old hardware.

2.5 antiX — Puppy's more "grown-up" relative, without systemd

Based on Debian but deliberately built without systemd, aimed at older hardware and users who want full control without unnecessary complexity — one of several distributions that exist specifically as a response to the systemd controversy.

2.6 4MLinux — Four Ms, one philosophy

The name refers to the distribution's four main use cases: maintenance, multimedia, myserver, and miniserver — an unusually clearly articulated identity for such a small, niche distribution.

2.7 Slax — Modular, portable, and built to run straight off a USB stick

Based on Debian but built around removable modules you can add or remove without reinstalling the whole system — an early and influential example of the "live USB" philosophy, long before it became standard in most distributions.

2.8 Sabotage Linux — Minimalism as self-flagellation

Built around musl libc and BusyBox instead of GNU's standard tools, with the goal of being as minimal and controllable as possible — a project whose name honestly sums up the experience of actually using it as a daily driver.

2.9 Absolute Linux — Slackware with the wheels already attached

Based on Slackware but with a pre-configured IceWM desktop and graphical settings tools, aimed at users who like Slackware's philosophy but don't necessarily want to configure every detail by hand.

2.10 Alpine Linux — The tiny distro that became the entire container world's standard

Built around musl libc and BusyBox instead of glibc/GNU, with a base image of just a few megabytes. That small size alone made Alpine the default choice as a base image for a huge share of the world's Docker containers — a distribution most developers use every single day without ever installing it on an actual computer.

---

3. Built by Rebels & Schisms

3.1 Devuan — Debian, minus the one controversial ingredient

When Debian switched to systemd as its default init system in 2014, a group of developers calling themselves "Veteran Unix Admins" refused to accept the change. The result was Devuan: a complete Debian fork whose sole real reason for existing is to offer the same system without systemd. The project's slogan, "Init Freedom," says it all about how much of a principled stand this was — an entire operating system founded, at its core, on a single architectural objection.

3.2 Artix Linux — Arch without systemd, for the same principled reasons

Same core motivation as Devuan, just applied to Arch instead of Debian: all of Arch's package selection and philosophy, but with alternative init systems like OpenRC or runit instead.

3.3 MX Linux — Two discontinued projects that merged and became more popular than either

MX Linux was born from a collaboration between the remnants of the antiX community and the former MEPIS project, after MEPIS itself shut down. The merger paradoxically became more popular than either founding project ever was on its own.

3.4 BunsenLabs — The community that refused to let a dead project stay dead

When the popular, minimalist Debian+Openbox distribution CrunchBang was discontinued in 2015 by its lone developer, a group of users simply took over the codebase and kept developing it under the name BunsenLabs — a clear example of how open source means no project ever has to stay truly dead as long as someone still cares.

3.5 Trisquel — The distribution Richard Stallman actually uses himself

One of very few distributions officially endorsed by the Free Software Foundation as "fully free," with absolutely no proprietary drivers or firmware blobs included — a deliberate, strictly principled choice that makes certain modern hardware components (like wifi) unreliable or entirely unusable out of the box.

3.6 Parabola GNU/Linux-libre — Arch, but with the FSF's stamp of approval

Just as Trisquel relates to Ubuntu, Parabola relates to Arch: the same foundation, but with every trace of proprietary code carefully surgically removed, at the cost of narrower hardware support.

3.7 gNewSense — One of the very earliest "fully free" distributions

Originally based on Ubuntu, with the same strict philosophy of including no proprietary code whatsoever. The project is now dormant, but for a long time it was one of the clearest examples of how far some in the community were willing to go for the principle.

3.8 GNU Guix System — The Nix idea, with GNU's own brand stamped on it

A functional, declarative operating system where every system configuration is described as code and can be reproduced exactly, built and maintained directly by the GNU project — a variation on the same idea as NixOS, but with GNU's stricter stance on free software baked in from the ground up.

---

4. State-Controlled & National Distros

4.1 Red Star OS — North Korea's own operating system, built to monitor its users

Based on Fedora but redesigned to visually resemble macOS, Red Star OS is used on state-approved computers in North Korea. Security researchers who've analyzed leaked copies have shown that the system secretly watermarks files and USB drives connected to the computer with traceable information — a feature whose only plausible purpose is making it possible to track how information spreads between citizens. Perhaps the clearest example there is that the same open source code giving us privacy tools like Tails can just as easily be turned in the opposite direction.

4.2 Astra Linux — The Russian military's Windows replacement

Developed by the Russian company RusBITech and certified to handle state secrets, including a special "Smolensk" edition approved for the highest security classification. The Russian military and several state agencies have gradually migrated away from Windows to Astra Linux, partly for pure sovereignty reasons around who controls the underlying code in critical systems.

4.3 UOS / Deepin — China's polished, state-backed desktop experience

Deepin, known for its unusually refined and aesthetically pleasing desktop, forms the basis for UOS (Unity Operating System) — a version with explicit state backing, intended to reduce China's technical dependence on foreign operating systems like Windows.

4.4 ROSA Linux — Yet another Russian distribution, with roots in France's Mandriva

Built by former Mandriva developers after the French company went bankrupt, with financial backing from Russian investors — a striking example of how European open source projects sometimes live on in unexpected geopolitical arrangements after the original company ceases to exist.

4.5 BOSS Linux — India's answer to "why pay licensing fees abroad"

Developed by India's National Resource Centre for Free and Open Source Software and widely used within Indian government administration — part of a broader global trend of countries deliberately opting out of foreign proprietary systems for their public infrastructure.

4.6 Pardus — Turkey's own distribution, built to cut licensing costs in schools and government

Developed by the Turkish research council TÜBİTAK, originally standalone and later rebuilt on a Debian base, aiming to reduce the country's dependence on foreign software licenses in the public sector and education.

4.7 Nova — Cuba's own distribution, built with a little help from abroad

Developed at a Cuban university, partly with support from China's equivalent state Linux efforts, as a way for Cuba to reduce its dependence on American software during decades of trade embargo.

---

5. Radically Different Architecture

5.1 NixOS — The operating system where nothing really overwrites anything else

Built on the Nix package manager, where every package and every system configuration is stored in isolation under a unique hash-based name. The result is a system where upgrades can never destroy a previously working configuration — you can literally roll back the entire operating system to exactly the state it was in last week, with a single command.

5.2 GoboLinux — The distribution that threw out the entire Unix file structure

Most Linux distributions share the same basic folder hierarchy (`/usr`, `/etc`, `/var`) dating back to early Unix systems. GoboLinux threw all of that out and built its own structure where every program literally lives in its own folder under `/Programs` — the same underlying Linux kernel, but

a completely rebuilt surface that first confuses, then often impresses, experienced Unix administrators.

5.3 Bedrock Linux — The distribution that lets you have your cake and eat it too

Bedrock solves a problem nobody knew they had: the ability to mix packages and tools from several different Linux distributions within a single running system at once, through a technical trick that links together multiple distributions' root filesystems in parallel. Want Ubuntu's hardware support, Arch's package availability, and Alpine's minimalism all at once, on the same install? Bedrock is pretty much the only project that even attempts it.

5.4 CRUX — Lightweight, source-based, and explicitly unwilling to be beginner-friendly

Created by Per Lidén with a BSD-inspired "ports" system for building packages from source, aimed specifically at experienced users — CRUX makes zero effort to hide its complexity behind a friendly interface, and is proud of it.

5.5 Source Mage GNU/Linux — Everything compiled from source, called "spells"

A package system where every install feels more like casting a spell than running an ordinary command — packages are called "spells" and collected in "grimoires." Technically it's really just source compilation with extra steps, but the playful terminology has earned the project a surprisingly devoted little following.

5.6 Void Linux — Its own package system, built from scratch with no Debian or Red Hat lineage

Unlike most independent distributions (which often still lean on apt or rpm under the hood), Void developed its own package manager, XBPS, and its own init system based on runit — one of very few larger projects that consistently opted out of every established building block in favor of building absolutely everything themselves.

---

6. Jokes, Satire & Passion Projects

6.1 Hannah Montana Linux — A joke that refuses to die

Originally released in 2009 as a pink, Disney-themed Ubuntu remix named after the teen TV star Hannah Montana, mostly intended as an inside joke within the community. The project was shut down, then unexpectedly resurrected decades later — as recently as July 2026 a new release appeared built on KDE Plasma 6, still consistently pink, still with no real explanation for why it keeps existing beyond the fact that someone, somewhere, refuses to let the joke die.

6.2 Ubuntu Satanic Edition — Ubuntu, but with pentagrams and death metal

An Ubuntu-based remix aimed at extreme metal fans, with a dark theme, pentagram-inspired graphics, and pre-installed metal-related screensavers — a clear example of just how broadly "customize Linux to fit your identity" can actually be taken.

6.3 Sabily (formerly Ubuntu Muslim Edition) — Ubuntu with the call to prayer built in

An Ubuntu remix aimed at Muslim users, with pre-installed tools for prayer times, Quran reading, and qibla direction (the direction toward Mecca) — an example of how specific, practical religious needs can genuinely be solved with the right pre-installed software.

6.4 Ichthux — Ubuntu for the Christian user

Aimed similarly at Christian users, with pre-installed Bible study tools and Christian media — the name references "ichthys," the classic fish symbol for Christianity.

6.5 Emmabuntüs — Built by a French charity to give old computers new life in Africa

Developed by the French nonprofit Emmaüs specifically to make it easy to refurbish and donate used computers to those in need, with a particular focus on Africa. The distribution deliberately ships with extra accessibility tools and a wide range of pre-installed educational software — probably the only distribution on this list whose primary reason for existing is charitable work rather than technology for technology's sake.

---

7. Retro Looks & Transition Distros

7.1 Q4OS — The Trinity desktop that refuses to leave the XP era

Uses the Trinity Desktop Environment, a continuation of the classic KDE 3, to deliver an interface that deliberately feels like an early-2000s Windows install — aimed squarely at users who want to leave Windows without having to relearn their entire muscle memory at the same time.

7.2 Zorin OS — Built explicitly to calm nervous Windows refugees

Offers layout options that mimic Windows or macOS depending on what the user is used to, aiming to make the transition to Linux as frictionless as possible for people who've never opened a terminal before and would ideally like to keep it that way.

7.3 Winux (formerly Linuxfx, formerly Windowsfx) — The distribution Microsoft wasn't too fond of

Built an interface so strikingly similar to Windows that Microsoft sent a formal objection to the original name "Windowsfx" in 2023 — the project renamed itself Linuxfx, and has since rebranded again to **Winux**, its current name as of 2024–2025, while keeping the familiar-looking surface throughout.

7.4 elementary OS — A desktop so polished people think it's macOS

Built on Ubuntu but with a completely custom, consistently designed desktop (Pantheon) frequently compared visually to macOS — one of the clearest examples of how much pure visual polish can shape how a Linux distribution is perceived by new users.

7.5 PCLinuxOS — The survivor from a time when rolling updates were unusual

An independent, rolling-release distribution that's existed since the early 2000s, with a dedicated community that sticks to a surprisingly traditional, Windows-like KDE interface in an industry constantly chasing the latest design trend.

---

8. Gaming & Entertainment

8.1 Lakka — Your entire computer becomes a retro game console

Built around the RetroArch emulation framework, Lakka turns essentially any computer or Raspberry Pi into a dedicated retro game console on boot — no desktop, no distractions, just emulators

and your game collections.

8.2 Batocera.linux — Lakka's more all-encompassing sibling

A similar concept to Lakka but with broader hardware support and more pre-installed emulators and interface options — one of the most popular "plug and play" solutions for home-built retro game consoles.

8.3 SteamOS — Valve's own distribution, built to make Linux a real game console

Originally launched for Steam Machines, SteamOS had its big breakthrough as the operating system behind the Steam Deck — a handheld game console that runs Windows games through the Proton compatibility layer, and has effectively proven that "Linux can't run my games" no longer holds the way it did ten years ago.

8.4 Drauger OS — An Ubuntu remix with exactly one focus: gaming

Strips away most of the general desktop clutter and instead optimizes explicitly for gaming performance, with pre-installed tools for Steam and other gaming platforms ready right at install.

---

9. Containers, Cloud & the Invisible Layer

9.1 Flatcar Container Linux — CoreOS, life after death

After Red Hat acquired and eventually shut down the original CoreOS project, the company Kinvolk stepped in and kept developing the codebase under the name Flatcar — a container-optimized, automatically self-updating system with no traditional package manager, built exclusively to run containers, nothing else.

9.2 Talos Linux — The server you never log into

Talos Linux deliberately has no SSH access and no traditional shell whatsoever — the entire system is managed exclusively through a secure API, specifically designed to run Kubernetes clusters. The philosophy is that if no human can ever log in and change something manually, the system can also never drift from its intended, declared state — administration by removing the possibility of administration.

9.3 Bottlerocket — Amazon's answer to the same idea

A minimal, container-focused operating system developed by AWS, built around the same core idea as Talos and Flatcar: strip away everything not directly needed to run containers, including traditional package managers and shells, to reduce both attack surface and maintenance burden.

9.4 RancherOS — An operating system where even the system services run as containers

A now-discontinued but conceptually interesting project where practically the entire operating system, including the Docker engine itself, ran as containers on top of an extremely minimal Linux kernel — containers all the way down.

9.5 Photon OS — VMware's own minimal container base

Developed by VMware specifically to run containerized workloads efficiently within VMware's own virtualization environments, sharing the same minimalist philosophy as the rest of this category.

9.6 Clear Linux — Intel's discontinued performance-tuned distribution

Built by Intel with aggressive compiler optimizations tailored to the company's own hardware, aimed at performance-sensitive cloud workloads. **Intel discontinued Clear Linux in July 2025**, halting updates and archiving its repositories — worth knowing before recommending it for anything new, though its optimization techniques remain an interesting case study of what a hardware maker building its own distribution can look like.

---

10. Education & Humanitarian Purpose

10.1 Sugar — The operating system built for "One Laptop per Child"

Developed as the interface for the ambitious OLPC project, which aimed to provide children in developing countries with cheap, rugged laptops. Sugar replaces the entire traditional desktop metaphor with a playful, activity-based interface designed specifically for children who've never seen a computer before.

10.2 Kano OS — Build your own computer, then learn to code on it

Developed to pair with Raspberry Pi-based kits, where children literally assemble their own computer before diving into built-in, gamified coding exercises right in the operating system. **Kano Computing, the company behind it, went into administration in 2023**; its project repositories have seen little activity since, so treat Kano OS as a product to research current status on before recommending it as an active option.

10.3 Skolelinux / Debian Edu — Norway's national school distribution

A Debian-based distribution developed specifically to pre-configure entire school IT environments — servers, student computers, and network authentication packaged and pre-installed, with the goal of letting a single, non-technical teacher manage an entire classroom's IT without outside consulting help.

10.4 Endless OS — Built for households without reliable internet access

Ships with a huge amount of content — encyclopedias, educational apps, offline maps — preinstalled directly on disk, explicitly aimed at users in regions where internet access is expensive, slow, or unreliable.

10.5 Edubuntu — The official Ubuntu branch for the classroom

An official Ubuntu variant pre-packaged with teaching tools and classroom management software, intended to be installed directly in a school environment without extensive extra configuration.

---

11. Forensics, Rescue & Specialized Tools

11.1 CAINE — Digital forensics, runnable straight off a USB stick

"Computer Aided Investigative Environment" bundles tools for digital evidence collection in a way that's carefully designed to avoid accidentally overwriting or altering evidence — legal rigor built directly into the system architecture.

11.2 DEFT Linux — Yet another forensic toolkit, with Italian roots

Similar in purpose to CAINE, originally developed in Italy for investigators and security researchers who need to examine a compromised or suspect machine without risking damage to evidence in the process.

11.3 SystemRescue — The toolkit you wish you'd had before your computer crashed

A live distribution packed with disk, partition, and recovery tools, meant to always sit ready on a USB stick in a drawer until the exact day your main computer refuses to boot.

11.4 Clonezilla — The open source answer to expensive commercial disk-imaging tools

A tool for cloning entire disks or partitions, widely used by IT departments that need to quickly set up large numbers of identical computers without installing each machine by hand.

11.5 GParted Live — An entire distribution built around a single tool

Exists practically just to give you access to the graphical partitioning tool GParted even on a computer that currently lacks a working operating system — an extremely narrow purpose, executed extremely well.

11.6 Rocks Cluster Distribution — Build a supercomputer cluster without being a supercomputer expert

Packages everything needed to set up and manage a high-performance computing cluster, historically popular in academic research as a way for smaller university departments to run clusters without a dedicated HPC team behind them.

11.7 Scientific Linux — CERN and Fermilab's joint Red Hat clone

Built jointly by two of the world's heaviest research institutions as a free, rebuilt variant of Red Hat Enterprise Linux for research environments — discontinued in 2019 in favor of CentOS, but for a long time the backbone of a significant share of the world's particle physics data.

---

12. Independent & Rolling-Release Desktop Projects

12.1 Solus — The distribution that nearly died with its founder

Built entirely from scratch with its own package manager (`eopkg`) and its own desktop (Budgie), without leaning on Debian, Arch, or Fedora. The project hit an existential crisis in 2018 when founder Ikey Doherty suddenly vanished from development without explanation — and has since gone dormant and been revived several more times, a project permanently balanced between being alive and having just barely avoided dying.

12.2 Manjaro — Arch Linux with the wheels already attached, if not always smoothly

Gives beginners access to Arch's massive package selection and rolling updates without having to install everything manually through Arch's own notoriously unwieldy install process. Has made headlines for the wrong reasons a few times, including when an expired SSL certificate temporarily blocked users from updating their systems — a reminder that "easier than Arch" doesn't always mean "without new problems of its own."

12.3 EndeavourOS — The phoenix that rose from Antergos's ashes

When the previously popular Arch-based distribution Antergos was discontinued without warning in 2019, a group of its users quickly banded together and launched EndeavourOS as an explicit spiritual successor — same target audience, same Arch foundation, just under a new name and a new set of maintainers.

12.4 Garuda Linux — Arch with extra everything, in neon colors

An Arch-based distribution known for its bold, colorful default theme and its focus on gaming performance, with pre-installed system tweaks and an advanced tool for easily rolling the system back when something breaks — Arch for people who want the power without having to build every piece themselves.

12.5 KDE Neon — The KDE project's own showcase distribution

Built on Ubuntu's stable foundation but with the absolute latest KDE Plasma desktop on top, updated considerably more often than the desktop environment in typical Ubuntu-based distributions — intended less as a distribution for everyday use and more as a showcase for what KDE is actually capable of right now.

12.6 Nitrux — A desktop built around AppImages instead of traditional packages

Uses its own KDE-based desktop (NX Desktop) and deliberately bets on the AppImage format for applications instead of traditional package management — an attempt to sidestep dependency problems by simply avoiding traditional packages where possible.

12.7 Peppermint OS — Built around the idea that web apps are apps enough

A lightweight system with a tool called "Ice" that lets you package any website as if it were a standalone desktop app — built for an era where more and more actual applications are just websites in another disguise anyway.

12.8 Bodhi Linux — The distribution that got its own desktop environment after a schism

Built around the Enlightenment desktop environment in its classic form. When the main Enlightenment project moved forward with changes the Bodhi team didn't want to follow, they simply forked the older version and renamed it Moksha — a desktop now maintained solely by and for Bodhi.

12.9 Feren OS — Yet another Windows-friendly, KDE-based transition distribution

Similar in ambition to Zorin and Q4OS: a familiar, Windows-like interface built on top of KDE, aimed at users who want Linux's performance and security without sacrificing the visuals they're already used to.

12.10 Chimera Linux — A modern from-scratch project with no GNU or glibc at its base

A relatively young, independent project built around musl libc and LLVM instead of the GNU toolchain, with its own init system — one of very few new "completely from scratch" projects to appear in recent years in an industry where most new distributions are content to be just another Ubuntu remix.

12.11 Vanilla OS — Ubuntu-based, but with atomic, fail-safe updates

Uses its own tool (ABRoot) to apply system updates atomically — either the entire update succeeds, or it's rolled back completely and automatically, a safeguard against the classic scenario where a half-finished update leaves the system in a broken in-between state.

---

13. Mobile & De-Googled Phones

13.1 postmarketOS — "Give your old phone ten more years of life"

Built on Alpine Linux with an explicit goal: making it possible to keep using old Android phones long after the manufacturer has stopped shipping security updates for them. The philosophy is essentially an environmental protest disguised as an operating system — every phone spared from premature disposal is, according to the project itself, a small win against the built-in obsolescence of the phone industry.

13.2 Mobian — Debian, but in your pocket

Adapts Debian to run directly on phones like the PinePhone, aiming to give mobile users exactly the same package repository and software freedom as a regular Debian computer — the phone just happens to be small, rather than being a separate ecosystem entirely.

13.3 Sailfish OS — What was left after Nokia abandoned MeeGo

Developed by Jolla, a company founded by former Nokia employees after Nokia shut down its MeeGo effort (whose only phone, the N9, was critically acclaimed but never got a successor). Sailfish inherited much of MeeGo's underlying technology and lives on as one of very few genuinely independent mobile operating systems that's neither Android nor iOS.

13.4 UBports — The community that rescued Ubuntu's phone project when Canonical gave up

When Canonical officially shut down its Ubuntu Touch mobile effort in 2017, a group of volunteer developers took over the entire project and continued it under the name UBports — another example of how a company's abandoned project can get a second life the moment the source code is open.

13.5 LineageOS — The phoenix that rose from CyanogenMod's ashes

When the company behind the hugely popular Android modification CyanogenMod shut down its entire operation in late 2016, the core developers immediately took over the code and continued it under the new name LineageOS — effectively the same project, the same community, just without the company that eventually became its Achilles' heel.

13.6 /e/OS — Android without Google, for anyone who wants the apps but not the tracking

Builds on the same core idea as LineageOS but goes a step further by actively replacing Google's services with its own privacy-focused alternatives — the same familiar app ecosystem, explicitly less inclined to report back on you to an ad platform.

---

14. NAS, Home Servers & Networking

14.1 Proxmox VE — Debian at its core, an entire virtualization center in practice

One of the most popular tools among hobbyists building their own home server clusters, built on top of Debian but packaged as a complete platform for managing virtual machines and containers through a web-based interface — enterprise-grade virtualization without the enterprise-grade price tag.

14.2 OpenMediaVault — NAS software built by a former FreeNAS developer

Created by one of the original developers behind the FreeBSD-based FreeNAS, but explicitly built on Debian instead — a deliberate choice to bring the same concept (simple, web-based network storage management) over to the Linux ecosystem.

14.3 TrueNAS SCALE — Where the FreeBSD legacy finally met Linux

iXsystems, the company behind the classic FreeBSD-based TrueNAS, launched SCALE as a Linux-based parallel version — a sign that even long-running BSD projects eventually bow to the Linux ecosystem's broader hardware support and container tooling.

14.4 IPFire — An entire distribution with a single job: being your firewall

Designed exclusively to run as a dedicated firewall/router appliance, with a modular add-on system for intrusion detection and content filtering — no desktop, nothing unnecessary, just network traffic being carefully inspected before it's let through.

14.5 OpenWrt — The router firmware your ISP didn't want you to know existed

An embedded Linux system designed to replace the often-mediocre stock firmware on routers, giving back features (advanced firewall, VPN, traffic shaping) that consumer routers rarely offer by default — an entire subgenre of hobbyism built around making cheap home networking gear disproportionately capable.

14.6 Univention Corporate Server — Germany's open answer to Active Directory

A Debian-based distribution built specifically around identity and access management for businesses and government agencies, intended as an open source alternative to Microsoft's Active Directory ecosystem — popular in German-speaking Europe among organizations that explicitly want to avoid locking their entire user management into a single American vendor.

14.7 openSUSE MicroOS — The server that fixes itself, or gives up and rolls back

An immutable operating system where the entire system updates atomically via Btrfs snapshots: either the update goes through completely cleanly, or the system automatically reboots into the last known working snapshot — the same core philosophy as Talos and Vanilla OS, applied to a general-purpose server OS rather than a narrow niche need.

14.8 Security Onion — An entire distribution built to catch intruders

Combines a range of network monitoring and intrusion detection tools into a pre-configured package, used by security teams that need to monitor network traffic for signs of a breach — a distribution whose users hopefully never actually need to use it for anything dramatic.

---

15. How to Build Your Own Linux Distro

If the list above has made one thing clear, it's that the bar for "creating your own Linux distribution" is actually surprisingly low — someone did, after all, build an entire operating system exclusively dedicated to a Disney star. Here's the short, honest truth about what the process actually looks like, for anyone who's felt inspired.

15.1 Step 1: Decide whether you're building a remix or something entirely your own

The overwhelming majority of distributions on this list are, at their core, an existing distribution (usually Ubuntu, Debian, or Arch) with your own package selection, your own theme, and your own installer layered on top — not a completely custom kernel or package manager. Building a remix is a weekend-to-month-long process; building something entirely from scratch, the way Void or GoboLinux did, is a multi-year commitment. Be honest with yourself about which one you're actually signing up for.

15.2 Step 2: Choose your base carefully, because you inherit its problems too

Pick Debian or Ubuntu as your foundation and you inherit apt, a massive package repository, and stable hardware compatibility — but also Debian's release cycles and conventions. Pick Arch and you inherit rolling updates and pacman, but also a less beginner-friendly culture. Fedora gives you dnf and Red Hat's more enterprise-friendly package structure. This choice affects practically every decision that follows, so make it deliberately rather than out of pure habit.

15.3 Step 3: Build your root filesystem with tools made specifically for this

For Debian-based projects, `debootstrap` is the standard tool for building a minimal, working root filesystem from scratch. Arch-based projects use the equivalent `pacstrap`. This step creates the actual filesystem your new operating system will consist of — the kernel, basic system tools, and the package database, with nothing extra layered on top yet.

15.4 Step 4: Choose your kernel, init system, and the truly foundational building blocks

Most remix projects simply reuse the base distribution's kernel and init system (usually systemd, unless you deliberately want to take the Devuan-style route and swap it out). If you're building something entirely your own, you now have to make decisions that would otherwise already have been made for you: which libc (glibc or the lighter musl), which init system, and which set of basic command-line tools (GNU coreutils or BusyBox).

15.5 Step 5: Curate your packages — this is where your distribution actually gets an identity

Which packages are pre-installed, which are removed, and which default settings are chosen is, in practice, the most identity-defining part of the entire process. A gaming-focused distro like Drauger OS and a privacy-focused distro like Tails can share the exact same Debian base, yet be perceived as completely different products purely through package selection and default configuration.

15.6 Step 6: Design the desktop environment and the visual theme

Choose a desktop environment (GNOME, KDE Plasma, Xfce, or something more niche like Trinity if you deliberately want to evoke nostalgia) and build your own theme, icon set, and wallpapers. This step requires the least technical skill but has a disproportionately large effect on how professional the end result feels — just compare elementary OS's polished surface to most anonymous Ubuntu remixes.

15.7 Step 7: Build an installer, or a bootable live image

For Debian-based projects, `live-build` is the standard tool for packaging the entire system into a bootable ISO file, often with a graphical installer like Calamares (used by MX Linux and several other independent distributions, among others) layered on top. Arch-based projects have the equivalent `archiso`. Tools like Cubic make it possible to build a custom Ubuntu-based ISO without having to memorize the whole process by hand.

15.8 Step 8: Test obscenely thoroughly, ideally in virtual machines first

Install your new distribution over and over in a virtual environment (QEMU/KVM or VirtualBox) before you even think about testing it on real hardware. Pay particular attention to whether the install process itself works flawlessly — a broken installer is the fastest way to ensure nobody ever gives your distribution a second chance.

15.9 Step 9: Decide how you'll handle updates long-term

Do you inherit the base distribution's package repository as-is, or run your own repository? Most smaller projects rely entirely on the base distribution's upstream packages and add only a handful of their own on top — maintaining a fully independent package repository is a commitment in itself, requiring its own infrastructure and ongoing operation.

15.10 Step 10: Publish, but be ready for the maintenance to be the real job

Building version 1.0 is the fun part. Security updates, bug reports, hardware compatibility, and a community that expects answers are what actually determine whether your distribution is still around in five years or simply vanishes from DistroWatch, like so many hundred projects before it. If you still decide to go ahead — welcome to the list. There's plenty of room for a hundred more.

© SplitZrate 2026 · 100 Linux Distros You Never Knew Existed
