

PICKING A LINUX DISTRO



Linux/GNU, commonly referred to as Linux, is a free, open-source OS that was created by Linus Torvalds. Because its source code is free and open, people all over the world help maintain, update and create more features for the OS. (New operating systems, like Google's Android, have even been built using the Linux kernel.) As a result, Linux is extremely customizable and can be tailored to all sorts of specific needs. For these reasons, a majority of servers around the world — including Linode's — use Linux as their OS.

Different user communities adapt Linux differently, creating their own distributions — Fedora, Debian, Gentoo — contingent on specific need. Most Linux distributions, or distros, can be used universally, but some are better at specific tasks. For example, Arch is an extremely powerful, detailed distro that grants the user customization over even the smallest functionalities of a solution in the stack. The Ubuntu distro handles many small configuration details by default, rendering installation and configuration easier for newer users.

PAGE

Arch

A minimalist, detail-oriented, bleeding-edge distribution for power users. One of the more complicated distros to run, Arch empowers a user to manually customize every nuance about the server's configuration. Rolling updates keep all Arch versions uniform and consistent.

3

CentOS

Essentially the Red Hat distribution without proprietary services, CentOS is usually selected for its powerful web-hosting capabilities, as it can install and run cPanel. (Red Hat is a distribution well known by enterprises that use Linux servers for their web solutions — it's one of the few distros that comes with a price tag, but also a telephone/ticket call line for customer support.) Administrators who have experience with Red Hat but don't want to pay for persistent licensing can migrate to and use CentOS, instead.

4

Debian

One of the earliest distributions, Debian has become one of the most popular and frequently maintained distros among Linux users. It leverages stable, steady and reliable services with greater deliberation about adopting recently released features available in other distros.

5

Fedora

Accommodates the latest, bleeding-edge software. It supports and runs services prototypes that can be considered unstable.

6

Gentoo

This is a detail-oriented, challenging distro. Its source code is compiled by the user to fit exact needs with installed hardware. By proxy, this would mean users are intimately knowledgeable about their now optimized server. Like Arch Linux, Gentoo follows rolling updates.

7

OpenSUSE

This is a detail-oriented, challenging distro. Its source code is compiled by the user to fit exact needs with installed hardware. By proxy, this would mean users are intimately knowledgeable about their now optimized server. Like Arch Linux, Gentoo follows rolling updates.

8

Slackware

This is a detail-oriented, challenging distro. Its source code is compiled by the user to fit exact needs with installed hardware. By proxy, this would mean users are intimately knowledgeable about their now optimized server. Like Arch Linux, Gentoo follows rolling updates.

9

Ubuntu

One of the easiest distros to learn and operate, Ubuntu is arguably the most popular. Derived from Debian, Ubuntu uses the same package management, but its own repositories. Ubuntu gets updates throughout each year, with longer-lasting LTS versions released every two years.

10



In 2001, Judd Vinet began work on what would become Arch in 2001 and released its incipient version a year later. Since then, Arch's user base and community have grown steadily.

In 2007, Vinet retired from his role as lead Arch developer and passed the responsibility to Aaron Griffin, who still heads the Arch Linux project today.

What Makes Arch Stand Out?

Arch Linux has earned a reputation for speed, customization and simplicity because installation starts at the command line and finishes at a shell prompt, where a user must load preferred software to attain a practical - and exclusive - working environment.

This minimal, tailored foundation supersedes the vanilla installations of other distros that are cluttered with unnecessary software, services and resource consumption. Arch supplies all the tools and instructions necessary to build an installation to exact specifications.

This do-it-yourself philosophy of user centrality is a touchstone along "The Arch Way."

Still, Arch's keystone remains simplicity. Because it is a rolling release (with a monthly base installer update), Arch never needs to be reinstalled, only updated with its package manager, Pacman.

Pacman works entirely with .xz files as opposed to using RPM, DEB or Snap packages. XZ file compression uses the LZMA2 algorithm which results in smaller downloads and faster decompression than other packaging container formats.

And because it's comprised of the latest, stable, upstream "bleeding-edge" packages, Arch makes available all that upstream resources provide.

The core repositories maintained by the Arch developers serve as Arch's primary software sources. Also, the popular, community-driven Arch User Repository (AUR) contains tens of thousands of PKGBUILDs, shell scripts which contain information for building packages with the makepkg tool and installing them with Pacman. Using PKGBUILDs, the AUR can install a wealth of apps which can't be found in other distros' repositories.

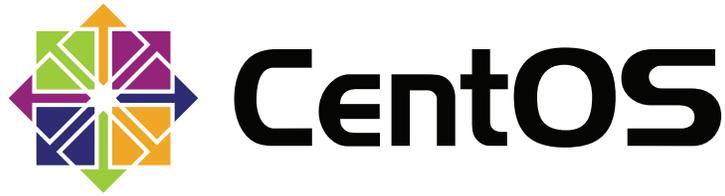
For guidance, Arch offers its ArchWiki, a documentation library that gives new Arch users all instructions and tips necessary to install, configure and maintain the distro.

Who Would Benefit Most using Arch?

Arch's beauty is that it can function equally as a server, a primary workstation or anything in-between. Arch's online forums recount the distribution being used exclusively on desktops, laptops, virtual servers and embedded hardware.

Consider, however, that an incipient Arch user must be willing to read and learn. Installation and major upstream changes can frustrate prospective users who encounter a steep learning curve. Fair or not, Arch has a reputation for being newbie-unfriendly, and better targeted at veteran Linux users.

Regardless of a user's experience, installing, configuring and using Arch will teach a tremendous amount about Linux. If that is also a goal, you could be very satisfied with Arch Linux.



CentOS is a younger distro in the Linode lineup. Created as the development platform for CAOS Linux, CentOS eventually shunted to its own project path with its first release, version 2, in 2004.

The name CentOS is an acronym for Community Enterprise Operating System. The project is overseen by the CentOS Governing Board, which integrates with Special Interest Groups, or SIGs. SIGs are teams within the community that focus on specific areas in the project, such as documentation or alternate architectures.

As of 2014, the CentOS Project is under the wing of Red Hat as part of its Open Source and Standards Team. Some CentOS members work for Red Hat, which sponsors some of the Project's resources.

What Makes CentOS Stand Out?

One of CentOS' biggest attractions is its ten-year support lifespan, and what happens within those ten years (or rather, what doesn't). Major features and package versions are only introduced with new milestone releases (CentOS 6, 7, etc.). Unless a significant issue arises, this is a distro maintained primarily for stability and security; you won't find the the newest Linux components in CentOS.

Yet, that the intended advantage. A conservative approach to new software adoption is a major factor in enterprise environments where reliability and compatibility with custom tools are essential.

A CentOS milestone release will receive general support for seven years, after which it will receive only critical security fixes for an additional three years. This is known as the Production 3 Phase of the RHEL life cycle, which CentOS follows, although it's recommended to upgrade before this phase.

RHEL undergoes hardware validation by manufacturers to ensure the operating system runs optimally on the respective equipment of each. Because CentOS is packaged from RHEL's publicly available source code, hardware validation is indirectly a feature of CentOS as well, although some Red Hat binaries, such as drivers and utilities, aren't available in CentOS.

CentOS is an RPM-based distribution which uses yum as its package manager, systemd as its init system, and implements SELinux enforcement by default. The distribution is available in a variety of options and configurations, from a minimal .iso to the Everything image, including purpose-built Gnome and KDE live images.

The primary CentOS architecture is x86-64, but ARM is one of several alternates available. CentOS accommodates container images for Docker, Vagrant and the like. CentOS Atomic has been designed specifically as a host system for Docker containers.

The CentOS Project also provides a GPG signed checksum list for its CentOS 7 images, and online key fingerprints.

Who Would Benefit Most using CentOS?

CentOS tracks the public source of Red Hat Enterprise Linux, so the end result is an operating system very close to RHEL. If that's an environment you're looking for, or if you need strong compatibility with RHEL, CentOS could be a great fit for you.

However, CentOS is not only for those with Red Hat experience. Its controlled, paced version upgrades make it well suited for any software stack where reliability is priority.

Moreover, packages which aren't publicly available upstream in RHEL won't be installed in CentOS; the distribution is fully free and open source by default, but there are third-party repositories for additional software, such as media codecs.

Both RHEL and CentOS are used for large-scale, enterprise-level servers and workstations. Once a milestone is released, new features are rarely added; excepting updates for security issues and bug fixes.

You will not find bleeding-edge software on CentOS. What you will find is your software stack operating for up to a decade on a stable, reliable and robust operating system.



debian

Debian is one of the oldest Linux distributions available and is the foundation for many other distros from Finnix to Ubuntu. The Debian Project was founded in 1993 by Ian Murdock with the vision of creating a non-commercial operating system which values open development, Linux and GNU, yet is competitive with commercial systems.

What Makes Debian Stand Out?

There are three different branches of Debian: Stable, Testing and Unstable. The Debian Project supplies about three years of security support and important fixes to Stable, at which point the Debian LTS team furnishes support for about another two years.

Stable has earned a reputation for being a solid, reliable operating system, so it's no surprise to find it often used in production environments. Debian and CentOS have traded places as the most widely used Linux distribution running the world's web servers. This stability does come at the cost of packaging newer software because it's a difficult feat to combine extreme stability with bleeding-edge packages.

Depending on how you use Debian, it can be considered both a rolling and standard release. To upgrade Debian versions without a reinstallation - say, from Debian 8 to a Stable 9, just change the source repositories to the next codename and run a dist-upgrade.

However, if you edit your sources.list file to "track" the Testing or Unstable branches, then a dist-upgrade would jump your installation to Debian Testing or Unstable. From there, you will perpetually pull from the whichever of those repository branches you chose, regardless of which codename is currently in them.

Another advantage of Debian is the presence of abundant architecture ports. Probably the most well known non-x86 implementation is Raspbian, the Debian-powered operating system for ARM-based Raspberry Pi computers. Debian packages use the deb format, while Apt and aptitude are the primary package management options with various frontend controllers available.

The Main repository contains about 99% of the software available to Debian. Contrib is an additional repository of

software that also complies with the Debian Free Software Guidelines and is intended to work with the Debian distribution. However, Contrib requires software outside of the distribution to either build or function, such as needing packages from the non-free repository. Non-free is where you'll find software which doesn't meet the DFSG, apps like Adobe Flash and proprietary media codecs.

Packages from Main are officially part of and maintained by the Debian Project, whereas the contrib and non-free repositories are not. It's also the Main repo which is curated by the Debian LTS team after official support has ended for a Debian version.

Who Would Benefit Most using Debian?

If you're looking to run a server for a web application, database or networked storage, Debian Stable is a natural choice. It is equally comfortable powering your VPN as it is your laptop. With server, desktop and minimal installers available, you have the choice of starting small and building your own system, or beginning with everything and removing unnecessary components as you go.

Debian's easy upgrade procedure lends itself to those who don't want to reinstall after each new milestone release. Debian also provides GPG-signed checksum files for verifying your image's integrity before committing it to media.

Debian Testing and Unstable versions are best for those users comfortable with either sacrificing some system stability for newer software adoption, or adventuring with development branches. Please note: security updates for Stable are priority while those for the Testing branch are not. Another consideration: Unstable has no security team support.

Debian stands as a free and open-source stable operating system for multiple architectures. With about five years of support (three years of security support and important fixes plus two additional years of security support for certain architectures), easy upgrades, and various compile-time security features, Debian advances towards fully reproducible package builds.



Fedora began as a grassroots, university computer-science project. The Fedora Project, started by Warren Togami in 2002 as a community-maintained package repository for Red Hat Linux, predated the distributed operating system. The Project's purpose was to provide tried-and-tested software which wasn't available in RHL.

When Red Hat decided to focus on enterprise customers with its Enterprise Linux, the Fedora Project merged with the former RHL and became the community upstream. In early 2004, the first Fedora Core release went public.

What Makes Fedora Stand Out?

Fedora is available in three branches: Workstation, Server and Cloud. The default desktop environment is Gnome Shell, but Fedora Spins can serve alternative desktops. Fedora Labs are supplemental spins for preconfigured software collections, such as gaming, forensics and astronomy. Fedora has ARM ports as well.

The lifespan of a Fedora release is about 13 months and, similar to Debian, upgrading Fedora versions is a matter of changing the repository source and using the package manager; no need to reinstall.

Fedora is an RPM-based distribution which aims to offer the newest stable packages while working closely with upstream package maintainers to commit bug fixes and patches. Fedora is another distribution with extensive wiki documentation, which applies to more than Fedora. Ask Fedora, the project's official forum, offers ready and ample community support for the distro and its spins.

Current apps unique to Fedora are DNF and firewalld. DNF is the evolution of the Yum package manager, created originally as a fork to correct various issues with Yum, ranging from a more comprehensive and usable API, to extensibility, to dependency resolution. Firewalld is a new dynamic firewall daemon which emphasizes different zones based on levels of trust.

Fedora Server also includes Cockpit by default, which provides a graphical management console via web browser login.

The Fedora Project is heavily security-focused, to the extent that the entire getfedora.com domain uses TLS 1.2 with DNSSEC and DANE. Memory and execution protections are implemented. PGP keys with online fingerprints are available for verifying installer checksum files; while by default, SELinux is set to enforce mode and firewalld is enabled. Red Hat has set the longterm goal of reproducible builds for Fedora's RPM binaries, and Fedora also supports Secure Boot.

Who Would Benefit Most Using Fedora?

If you're already comfortable with Red Hat, then of course you'll likely feel right at home with Fedora. But this distro isn't limited to a RHEL partiality. Whether the default offering or a spin, Fedora gives you a powerful workstation or server operating system, or lets you install a minimal base from which you can build.

Moreover, third-party software sources, which Fedora must exclude for licensing reasons, and Fedora's rolling-releases make the distro a viable option for those who don't want to reinstall a system with each new release.

The Fedora Project's efforts are highlighted by their values of free-and-open-source software, mindfulness of security, and ease of use. The result is a distribution that has become very popular on the desktop and is also used on the servers of some powerhouse companies, small and large.

Whether in the form of sponsorship or community support, Fedora receives accolades from big names in the Linux world and will happily adapt to whichever use case you throw at it.



Gentoo is yet another Linux success story. In 1999, Daniel Robbins released Enoch Linux, the precedent to today's Gentoo. Shortly thereafter, Enoch was renamed in honor of the Gentoo penguin, the fastest underwater penguin. Next, Robbins released version 1.0 on October 4, 1999, the date he registered the domain gentoo.org.

What Makes Gentoo Stand Out?

Gentoo is a rolling-release distro and self-described meta-distribution, lacking a default or standardized installation. One of the Gentoo Project's highest values is to provide the framework for creating an operating system tailored specifically to a user's preferences. With a base system large enough to run the kernel and vital drivers, everything else — from desktop environment, to init system, to specific compiler flags — is open to user choice.

You can build the distro entirely from source code on a local machine. Gentoo uses Portage (with its command-line tool emerge) for package management and bucks Linode distro tradition by not relying on binary software repositories.

Fear not, however. A repository exists of ebuilds which contain metadata, dependencies and scripting, necessary tools to build a tailored package. Portage resolves these dependencies, downloads source code, compiles and merges a package into the operating system. In addition, package sources are mirrored.

Gentoo sports differing terminology. You merge or unmerge software using ebuilds instead of installing or uninstalling a binary as you would with a .deb or .rpm distro. Besides merging from an ebuild, Portage can build and install local binaries.

If choosing your own USE flags sours you on Gentoo, know that its profiles are Portage blueprints for software sets and make configurations. You'll also find profiles for Gnome and KDE desktops, developer tools, and inclusion of SELinux or systemd.

If you want extraordinary security, hardened Gentoo is a base profile to compile packages using more secure GCC flags than a normal installation, and adds features like PaX, SELinux, and a Grsecurity-compiled kernel. Also available to all images and tarballs are OpenPGP-signed digest files containing each SHA checksums, with fingerprints verifiable online.

Who Would Benefit Most Using Gentoo?

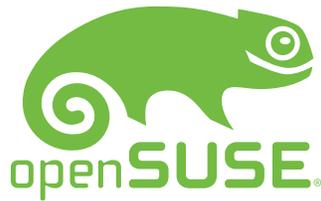
While Gentoo can challenge Linux newbies, it is well documented in its wiki. The Gentoo Handbook contains necessary info to load and run the distro. Discussion forums and mailing lists for asking questions help get you on track.

Gentoo has access to all the usual software you'd expect on Linux, so it could power your VPN just as easily as your desktop. Unless you perform major hardware changes, the installation is a one-timer. Subsequent Gentoo updates require the updating of its repository and commanding Emerge to run a sync.

Gentoo generally appeals most to those who need to accomplish a task that's more out-of-the-box than that typically found with a binary distro. If bleeding-edge packages are a passion -- e.g., a non-standard libc implementation? -- Gentoo can satisfy.

The security-minded will appreciate Gentoo's hardened profile and base tarballs. In fact, Google's Chromium Project adopted Gentoo and Portage as the framework for ChromeOS.

If you're willing to invest the time installing Gentoo, it can reward you with an extremely customized and streamlined experience. Along the way, Gentoo will also teach you a tremendous amount about Linux and the basics of software compilation.



S.u.S.E. started in 1992 not as a Linux distribution, but as a German software development company whose name Software und System Entwicklung evolved into the acronym and today's SUSE and openSUSE (pronounced "zoo-zah") Linux distros.

What Makes openSUSE Stand Out?

OpenSUSE is the community-maintained development base for SUSE Linux Enterprise, the future of which was reassessed in 2015 when it was decided impractical to chase a single branch that tried to focus on both stability suitable for servers and workstations, and upstream packages suitable for bleeding edge development.

Consequently, openSUSE has diverged into two distinct branches as the 13.x releases began phase out. The Tumbleweed branch is a rolling release which aims to provide the latest stable versions of its package base. Leap branch is a free, enterprise-level version with SUSE Linux Enterprise at its core. It focuses on stability and reliability, combines with community packages and holds a minimum, three-year support lifespan.

OpenSUSE has traditionally focused on the KDE desktop environment and become the distro of choice for a matchless KDE Plasma experience. Today, the distro's performance gets equal accolades for use with Gnome, XFCE and LXDE, while alternatives like Cinnamon and MATE are available from the repositories.

The Btrfs filesystem and Snapper have become openSUSE staples and OpenQA is used to smooth out wrinkles in image builds leading up to their release. OpenSUSE has broad documentation in various formats if online HTML pages don't fit your needs.

AppArmor is openSUSE's preferred, mandatory access control and it's enforcing the default-supplied policies out of the box, although SELinux is available as well. SuSEfirewall2 is openSUSE's iptables wrapper, enabled by default for easy firewall management and protection. OpenSUSE is also Secure Boot compliant.

Unique to openSUSE is the SUSE Studio, an online tool for customizing SUSE and openSUSE Leap installation images. You add the packages you want, configure services and export to an image for optical or USB media, virtualization or containerization.

SUSE Studio also contains downloadable image appliances, which other users have created and maintained. The backend to SUSE Studio is Kiwi, a CLI tool for building SUSE and openSUSE appliances from a Linux command line.

YaST2, openSUSE's graphical configuration manager, is a platform of modules that manage everything on the system, from sudo users to peripheral devices to system services, without touching a config file. YaST executes in openSUSE's installer and sets SSH access, firewall, and kernel and bootloader parameters, etc.

In addition, AutoYaST let's you make more granular changes and eases multi-machine installation, akin to Fedora's Kickstart or the Debian family's preseed files. You don't need a desktop to use YaST, either. If you're running openSUSE on a headless server, YaST is fully functional as an ncurses menu system launched from the command line.

Who Would Benefit Most Using openSUSE?

Between SUSE Studio and YaST, openSUSE could be an administrator's panacea, or even a win for a Linux neophyte. One of YaST's most valuable attributes: simplifying system configuration beyond the desktop environment's standard settings panel.

Similar to the relationship between CentOS and RedHat, openSUSE is also ideal for those interested in SUSE Enterprise Linux or in carrying home SLE's workflow from the office or job site. The distro performs equally well running a casual home computer, a powerful scientific workstation or a populated database. Community support remains current in the official online forums and mailing list.

OpenPGP fingerprints and SHA256 to verify an installer, Btrfs snapshots, support for multimedia codecs, and Adobe Flash during installation are just a few more reasons to try openSUSE.

With rolling releases and enterprise-stable branches to choose from, openSUSE let's you prioritize either system stability or bleeding-edge package execution without ever leaving its ecosystem.

slackware

l i n u x

Whether adding packages, changing themes or building modules for obscure hardware, the ultimate purpose behind modifying a distro after a fresh installation is to make the system work better for the user. And that's how Slackware began: as a university student's collection of post-install notes and fixes for the Softlanding Linux System.

What Makes Slackware Stand Out?

Slackware's prominence arises from its near-UNIX-like characteristics, a pillar of the distro's philosophy. Slackware's init system is a hybrid of System V and BSD methods.

Slackware strives to render a complete system, including several apps unavailable as defaults in other distros. Development environments, mail, web and FTP servers, diagnostics tools, even kernel source code are all included.

Slackware comes in two branches: stable and current, both for 32-bit x86, 64-bit x86_64, and ARM. Stable is offered as a solid, thoroughly tested release for desktops or servers; while current is the development branch which leverages the latest software with reduced reliability. Slackware is also known for long lifespans. Milestone releases dating back to mid-2009 still have no planned end of life.

Slackware's maintainers desist from making changes to upstream software. Infrequent patching is done to alter software from its release state. This purity is often cited as a most desirable feature.

Pkgtool, Slackware's package management toolkit, packages software into XZ archives using LZMA2 compression, so downloads and extractions are as efficient as possible.

Third-party software packages are managed through volunteer-driven SlackBuilds.org, or SBo, which maintains a repository of build scripts to download, compile and install various applications into the distro, akin to Arch's AUR. SBo is unaffiliated with the Slackware project, but users can find thousands of GPG-signed SlackBuild scripts

with concise HOW-TO install or write-your-own guides. Slackware handles package dependencies uniquely. All dependencies of the recommended full installation are satisfied, but the responsibility falls to the end user to ensure additional software dependencies are met. Albeit a somewhat controversial decision, this policy stands as another pillar of Slackware's core principles.

Slackware follows no set release schedule; new milestones are made available only when they're ready. Slackware has no official policy regarding support lifespan, either, so long-ago milestones remain fully supported, both in terms of new features and security patches. Slackpkg handles Upgrading to a new release, averting full reinstallation.

Who Would Benefit Most using Slackware?

While Slackware provides one of the larger default software sets compared to others, it doesn't aim to satisfy everyone. Its team never intended to create a distro for the masses.

Experienced Linux users will get the most from using Slackware. Various wikis and documentation sites exist to help. Should difficulties arise, Slackware sub-forums at LinuxQuestions.org, a disparate IRC channel, and a newsgroup exist online to help.

If you're looking for a server or desktop distribution that you can slim down to a minimal installation for as lean a system as possible, give Slackware a try.

If you prefer configuring the system with text files, a simple init system and deliberate lack of upstream changes — along with the assurance that these traits will not change — then, give Slackware a try.

Still unsure? Peruse the Slackware Way. Note the available features bulleted. If you like what you see, give Slackware a try.

ubuntu

Ubuntu is a South African word which loosely translates to one of two meanings: “humanity to others” and “I am what I am because of who we all are.” As a Linux operating system, the inaugural Ubuntu 4.10 launched in 2004. South African entrepreneur Mark Shuttleworth code-named his creation Warty Warthog.

What Makes Ubuntu Stand Out?

Ubuntu ranks among the most visible and popular desktop Linux distributions available. Canonical has put substantial effort into making the distribution an easy and smooth experience to install, maintain and upgrade; so, it’s universality is no surprise.

Ubuntu releases come in two support lifespans. Regular releases are made every six months and receive support for a subsequent nine months. Long-term support (LTS) releases are made every two years and each is supported for the following five years.

Ubuntu is one of the most often forked distributions and is the base for distros such as Linux Mint, Zorin OS and Google’s internal OS Goobuntu.

In addition to the standard Unity version provided by Canonical, Ubuntu comes in many flavors. These are pre-built images using different desktop environments (KDE, XFCE, etc.) and while not Canonical products, many are supported in part by Canonical and receive direct contribution by the Ubuntu project.

A significant amount of work with Ubuntu Touch has delivered Ubuntu to tablets and smartphones. Unity 8 will finally bring Convergence to Ubuntu for a seamless experience between desktop and mobile devices, and Canonical has been expanding Ubuntu’s reach to Internet-of-Things, cloud solutions, and smart televisions. Even in-car electronics have been discussed.

Because it’s such a popular distribution, community support is ready. Answers are abundant. Ubuntu’s Personal Package Archives makes it easy to install software

which isn’t available in the repositories, and to package your own projects or builds. Canonical has also worked with Microsoft to build an Ubuntu userspace in Windows 10, with a full Bash shell.

For security features, Ubuntu has detailed security matrix which includes things like memory protection, kernel hardening and no-listening network services. The FTP archives contain OpenPGP signed checksum files for verifying installation images.

Who Would Benefit Most using Ubuntu?

Ubuntu is a bit of an “every-man” distro. It is largely considered very friendly to Linux newcomers. The source for Ubuntu’s package base is Debian’s Unstable repository (Sid). Depending on where Debian and Ubuntu are in their release cycles, an Ubuntu LTS could suit you well if you want long term support and newer packages than the current Debian Stable, but without leaving the DEB ecosystem.

If you already have knowledge of Debian, Ubuntu’s workflow is synonymous, with slight differences regarding providing package versions versus native properties of the OS.

Ubuntu’s assortment of installers caters to various needs: from building your own operating system from a minimal installation base with the mini.iso to server, to the general desktop installer, which can automatically configure disk partitioning for installs, to a dual-boot configuration with another operating system.

Ubuntu has become the most available multi-platform Linux distribution and is becoming more accessible as an OEM-supplied operating system. If you’re interested in an Ubuntu phone, tablet or other Smart device, Ubuntu on your desktop or server could be a perfect complement.