Rocky Linux 9 Administration

Deploying, Securing, and Managing Enterprise Linux Systems

Preface

When CentOS—long the backbone of countless enterprise environments—shifted its focus from a stable, downstream rebuild of Red Hat Enterprise Linux to the rolling-release CentOS Stream, the Linux community faced an inflection point. System administrators, DevOps engineers, and organizations around the world needed a reliable successor: a community-driven, enterprise-grade distribution they could trust in production. **Rocky Linux** emerged as that successor, founded by Gregory Kurtzer, one of the original creators of CentOS, with a clear mission—to provide a free, open-source, production-ready enterprise Linux platform that would never waver from its commitment to stability and compatibility.

This book, *Rocky Linux 9 Administration: Deploying, Securing, and Managing Enterprise Linux Systems*, is written for anyone who wants to master Rocky Linux 9 as a platform for real-world infrastructure. Whether you are a newcomer stepping into Linux administration for the first time or an experienced professional migrating workloads from CentOS, RHEL, or another distribution, this book provides the practical knowledge and hands-on guidance you need to deploy, secure, and manage Rocky Linux systems with confidence.

What This Book Covers

The journey begins with **Chapter 1**, which introduces Rocky Linux 9—its origins, its relationship to the broader enterprise Linux ecosystem, and why it has rapidly become a trusted choice for production environments. From there, **Chapters 2 through 6** walk you through installation, initial setup, user and group administration, package management, disk and filesystem management, and the fundamentals of LVM and RAID. These chapters establish the core competencies every Rocky Linux administrator needs.

Chapters 7 and 8 turn to networking and service management, covering Rocky Linux's networking stack and the essential skill of managing services with `systemd`. **Chapters 9 and 10** move into deployment territory, guiding you through standing up web servers, databases, and application services on Rocky Linux—tasks that form the daily work of infrastructure teams worldwide.

Security is not an afterthought in this book; it is a central theme. **Chapters 11 and 12** provide thorough coverage of firewall configuration, network security, and SELinux—Rocky Linux's powerful mandatory access control framework. **Chapters 13 and 14** address logging, monitoring, backup, and system recovery, ensuring you can maintain visibility into your systems and recover gracefully when things go wrong.

The final chapters look forward. **Chapter 15** distills production best practices for running Rocky Linux at scale, while **Chapter 16** challenges you to think beyond day-to-day administration and grow into the role of an infrastructure engineer. The appendices—including a Rocky Linux command cheat sheet, service configuration templates, a security hardening checklist, a troubleshooting guide, and a curated learning path—serve as enduring references you will return to long after your first read.

Who This Book Is For

This book is designed for system administrators, DevOps practitioners, IT students, and anyone responsible for managing Linux infrastructure. No prior Rocky Linux experience is required, though basic familiarity with the Linux command line will help you move through the material more quickly.

How to Use This Book

The chapters are structured sequentially, building on one another, so reading from start to finish will give you the most cohesive learning experience. However, each chapter is also designed to stand on its own, making the book equally useful as a desk reference when you need to solve a specific problem on your Rocky Linux systems.

Acknowledgments

This book would not exist without the vibrant Rocky Linux community—the developers, contributors, and advocates who have built and sustained a distribution that thousands of organizations now depend on. I am grateful to the technical reviewers whose sharp eyes and deep expertise improved every chapter, and to the readers whose curiosity and dedication to learning make writing worthwhile.

Rocky Linux represents something important: the power of open-source collaboration to meet enterprise needs without compromise. I hope this book equips you to harness that power fully.

Let's get started.

Lucas Winfield

Table of Contents

Chapter 1: Introduction to Rocky Linux 9

The story of Rocky Linux begins not with a technical specification or a corporate boardroom decision, but with a community rallying together in response to an unexpected shift in the enterprise Linux landscape. When Red Hat announced in December 2020 that CentOS Linux, the beloved community rebuild of Red Hat Enterprise Linux, would shift its focus to CentOS Stream, a rolling-release distribution that sits upstream of RHEL rather than downstream, the reverberations were felt across data centers, development teams, and system administration departments worldwide. Thousands of organizations had built their infrastructure on CentOS precisely because it offered a free, binary-compatible alternative to RHEL with the same stability, the same package versions, and the same long support lifecycle. The transition to CentOS Stream meant that this reliable, predictable foundation was changing into something fundamentally different: a testing ground for future RHEL releases rather than a proven, stable rebuild.

Gregory Kurtzer, the original founder of the CentOS project, recognized the void that this decision would create. Within days of the announcement, he launched the Rocky Linux project, naming it after his late colleague Rocky McGaugh, one of the early CentOS co-founders. The mission was clear and unambiguous: to produce a community-driven, enterprise-grade operating system that would be bug-for-bug compatible with Red Hat Enterprise Linux, freely available to everyone, and governed by the community rather than a single corporate entity. This was not merely a technical exercise in recompiling source packages. It was a statement about the importance of open-source principles, community ownership, and the idea that enterprise-quality software should be accessible to all.

Rocky Linux 9, released in July 2022, represents the maturation of this vision. Built from the source code of Red Hat Enterprise Linux 9, Rocky Linux 9 inherits a wealth of modern technologies, security enhancements, and performance improvements while maintaining the stability and predictability that enterprise environments demand. It ships with Linux kernel 5.14, which brings improved hardware support, better performance for modern processors, and enhanced security features. The distribution includes GCC 11 as its default compiler, Python 3.9 as the system Python, and a modernized software stack that reflects the current state of enterprise computing.

Understanding where Rocky Linux 9 fits in the broader Linux ecosystem requires appreciating the relationship between several interconnected projects. At the foundation sits Fedora Linux, a community-driven distribution sponsored by Red Hat that serves as the upstream source for new technologies and innovations. Fedora moves quickly, adopting new software versions and experimental features with each release, typically every six months. Red Hat takes selected Fedora releases, stabilizes them, applies extensive testing and quality assurance, adds enterprise-specific features, and produces Red Hat Enterprise Linux. RHEL is a commercial product with paid subscriptions that include technical support, certified hardware and software compatibility, and security updates for an extended lifecycle. Rocky Linux then takes the publicly available source code of RHEL, removes Red Hat's trademarks and branding, recompiles the packages, and produces a freely distributable operating system that is functionally identical to RHEL.

This chain of derivation means that Rocky Linux 9 benefits from the enormous investment that Red Hat makes in developing, testing, and maintaining RHEL. Every security patch, every bug fix, every performance optimization that Red Hat engineers produce for RHEL 9 is reflected in Rocky Linux 9 through the recompilation process. The practical result is that administrators who deploy Rocky Linux 9 in their environments can expect the same behavior, the same package versions, the same kernel features, and the same application compatibility that they would find in RHEL 9, without the requirement of a paid subscription. This does not mean that Rocky Linux is a replacement for RHEL in every scenario. Organizations that require vendor support, certified configurations, or compliance with specific regulatory frameworks may still need RHEL subscriptions. But for development environments, testing infrastructure, web servers, container hosts, and countless other use cases, Rocky Linux 9 provides an outstanding foundation.

The Rocky Linux project is governed by the Rocky Enterprise Software Foundation (RESF), a public benefit corporation established to ensure that the project remains community-owned and community-directed. The RESF operates with transparency, publishing its governance documents, holding open meetings, and welcoming contributions from anyone who wishes to participate. This governance model directly addresses one of the concerns that arose from the CentOS situation: the risk that a single corporate entity could unilaterally change the direction of a project that millions of people depend on. By establishing the RESF as a legal entity with a clear charter and community accountability, the Rocky Linux project has created structural safeguards against such scenarios.

Rocky Linux 9 introduces several significant changes compared to Rocky Linux 8, reflecting the evolution of the underlying RHEL platform. The following table provides a comprehensive comparison of key components between the two major releases:

Component	Rocky Linux 8	Rocky Linux 9	Significance of Change
Kernel Version	4.18	5.14	Improved hardware support, better performance for AMD and Intel processors, enhanced eBPF capabilities, and improved security features including core scheduling for SMT
Default Compiler	GCC 8	GCC 11	Support for newer C and C++ standards, improved optimization capabilities, better diagnostics and error messages
System Python	Python 3.6	Python 3.9	Access to modern Python features including dictionary union operators, string methods, and improved type hinting
OpenSSL Version	1.1.1	3.0	Major architectural change with provider-based cryptography, FIPS module improvements, and deprecation of legacy algorithms

Default Filesystem	XFS	XFS	Continues with XFS as the default, with improvements to scalability and performance
Init System	systemd 239	systemd 250	Enhanced service management, improved logging, better container integration, and new features like portable services
Firewall Manager	firewalld with nftables	firewalld with nftables	Continued use of nftables backend with improved rule management and performance
Container Runtime	Podman 3.x	Podman 4.x	Improved pod management, better Kubernetes YAML support, and enhanced rootless container capabilities
Network Configuration	NetworkManager	NetworkManager	Enhanced with improved WiFi support, better VPN integration, and team/bond configuration improvements
Package Management	DNF 4 (YUM 4)	DNF 4 (YUM 4)	Continued use of DNF with modularity improvements and better dependency resolution

SELinux	Enforcing (targeted)	Enforcing (targeted)	Updated policies, improved container SELinux support, and better integration with modern workloads
Default Desktop	GNOME 3.32	GNOME 40	Major redesign of the GNOME desktop with improved workflows, new Activities overview, and better multi-monitor support
PHP Version	7.2 (default module)	8.0 (default module)	Significant performance improvements with JIT compilation, named arguments, union types, and attributes
Node.js	10 (default module)	16 (default module)	Access to modern JavaScript features, improved performance, and updated npm package manager
Database Options	MariaDB 10.3, PostgreSQL 10	MariaDB 10.5, PostgreSQL 13	Improved performance, new SQL features, better replication support, and enhanced security

One of the most significant changes in Rocky Linux 9 is the adoption of OpenSSL 3.0, which represents a fundamental architectural shift in how cryptographic operations are handled. OpenSSL 3.0 introduces a provider-based architecture where different cryptographic implementations can be loaded as modules. This has practical implications for administrators because some legacy cryptographic algorithms that were available in OpenSSL 1.1.1 are now deprecated or disabled by default. Applications that rely on older algorithms such as MD5 for certificate verification or legacy TLS versions may require configuration adjustments when migrating to Rocky Linux 9. The system-wide cryptographic policy framework, managed through the `update-crypto-policies` command, provides a centralized way to manage these settings across all applications on the system.

Rocky Linux 9 also brings important changes to the way software is packaged and distributed through the Application Stream (AppStream) repository. The AppStream concept, introduced in RHEL 8 and carried forward into Rocky Linux 9, allows multiple versions of user-space software to be available simultaneously through modular packaging. This means that administrators can choose between different versions of programming languages, databases, and web servers without waiting for a new major release of the operating system. For example, while Rocky Linux 9 ships with Python 3.9 as the default system Python, additional Python versions can be installed through AppStream modules without conflicting with the system installation.

The system requirements for Rocky Linux 9 reflect the modern hardware capabilities that the distribution is designed to leverage. The following table outlines the minimum and recommended specifications for different deployment scenarios:

Requirement	**Minimum Specification**	**Recommended for Server**	**Recommended for Desktop**
Processor	64-bit x86_64, aarch64, ppc64le, or s390x	2 or more cores	4 or more cores
RAM	1.5 GB	4 GB or more	4 GB or more

Disk Space	10 GB	20 GB or more (depending on packages)	20 GB or more
Network	Not required for installation from local media	Gigabit Ethernet recommended	Gigabit Ethernet or WiFi
Display	Not required for server installations	Not required for server installations	1024x768 minimum resolution

To verify the version of Rocky Linux installed on a system, administrators can use several commands that provide detailed information about the operating system:

```
cat /etc/rocky-release
```

This command outputs a single line identifying the Rocky Linux release, such as:

```
Rocky Linux release 9.3 (Blue Onyx)
```

For more detailed information, the `os-release` file provides machine-readable data about the distribution:

```
cat /etc/os-release
```

The output includes several fields that are useful for scripting and automation:

```
NAME="Rocky Linux"
VERSION="9.3 (Blue Onyx)"
ID="rocky"
ID_LIKE="rhel centos fedora"
VERSION_ID="9.3"
PLATFORM_ID="platform:el9"
PRETTY_NAME="Rocky Linux 9.3 (Blue Onyx)"
ANSI_COLOR="0;32"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:rocky:rocky:9::baseos"
HOME_URL="https://rockylinux.org/"
BUG_REPORT_URL="https://bugs.rockylinux.org/"
SUPPORT_END="2032-05-31"
ROCKY_SUPPORT_PRODUCT="Rocky-Linux-9"
```

```
ROCKY_SUPPORT_PRODUCT_VERSION="9.3"
```

Note that the `SUPPORT_END` field indicates that Rocky Linux 9 will receive security updates and bug fixes until May 31, 2032, providing a full ten-year support lifecycle that matches RHEL 9. This extended support period is one of the key advantages of Rocky Linux for enterprise deployments, as it allows organizations to plan their infrastructure with confidence that the operating system will be maintained for many years.

The `hostnamectl` command provides another useful view of system information:

```
hostnamectl
```

This produces output similar to:

```
 Static hostname: rocky9-server
       Icon name: computer-vm
         Chassis: vm
      Machine ID: a1b2c3d4e5f6a1b2c3d4e5f6a1b2c3d4
         Boot ID: f6e5d4c3b2a1f6e5d4c3b2a1f6e5d4c3
  Virtualization: kvm
Operating System: Rocky Linux 9.3 (Blue Onyx)
     CPE OS Name: cpe:/o:rocky:rocky:9::baseos
          Kernel: Linux 5.14.0-362.8.1.el9_3.x86_64
    Architecture: x86-64
 Hardware Vendor: QEMU
  Hardware Model: Standard PC (Q35 + ICH9, 2009)
```

For administrators who need to check the kernel version specifically, the `uname` command provides this information:

```
uname -r
```

Output:

```
5.14.0-362.8.1.el9_3.x86_64
```

The kernel version string contains several pieces of information. The `5.14.0` portion indicates the upstream Linux kernel version. The `362.8.1` portion represents the Red Hat kernel build number, which increments with each kernel update. The `el9_3` portion indicates that this kernel is built for Enterprise Linux 9.3, and `x86_64` identifies the processor architecture.

Rocky Linux 9 supports four processor architectures, each serving different segments of the enterprise computing market. The x86_64 architecture, also known as AMD64, is the most widely deployed and covers the vast majority of server and desktop hardware from Intel and AMD. The aarch64 architecture, commonly known as ARM64, has gained significant traction in cloud computing and edge deployments, with providers like AWS offering ARM-based Graviton instances that provide excellent price-performance ratios. The ppc64le architecture supports IBM Power Systems, which are widely used in financial services, healthcare, and other industries that rely on IBM infrastructure. The s390x architecture supports IBM Z mainframes, which remain critical infrastructure for many of the world's largest banks, insurance companies, and government agencies.

The Rocky Linux community maintains several communication channels and resources that administrators should be aware of. The official documentation, available at docs.rockylinux.org, provides installation guides, administration tutorials, and migration instructions. The Rocky Linux Forums offer a place for community discussion and peer support. The Mattermost chat platform and IRC channels on Libera.Chat provide real-time communication with other Rocky Linux users and developers. For organizations considering migration from CentOS Linux 8 to Rocky Linux 9, the community has developed migration tools and documented the process extensively.

When comparing Rocky Linux to other RHEL-compatible distributions, it is worth understanding the distinctions. AlmaLinux, another community project born from the CentOS transition, shares a similar goal of providing a free RHEL-compatible distribution. While both projects produce functionally equivalent operating systems, they differ in governance structure, build infrastructure, and community organization. Oracle Linux, produced by Oracle Corporation, is another RHEL-compatible distribution that includes Oracle's own Unbreakable Enterprise Kernel as an alternative to the RHEL kernel. Each of these options has its own strengths, but Rocky Linux distinguishes itself through its community-first governance model, its transparent build process using the Peridot build system, and its commitment to remaining free and open in perpetuity.

As you proceed through this book, every concept, every command, and every configuration example will be demonstrated on Rocky Linux 9. The skills you develop will be directly applicable to RHEL 9 environments as well, given the binary compatibility between the two distributions. Whether you are building a web server, configuring a database cluster, deploying containers, or hardening a system for regulatory compliance, Rocky Linux 9 provides the stable, secure, and well-supported foundation that enterprise workloads demand. The chapters that follow will guide you through installation, initial configuration, user management, storage administration, networking, security, and advanced topics, building your expertise step by step until you are confident in deploying and managing Rocky Linux 9 in any environment.

Note: Throughout this book, when you see commands prefixed with the hash symbol (#), this indicates that the command should be executed as the root user or with sudo privileges. Commands prefixed with the dollar sign ($) can be executed as a regular user. Always exercise caution when running commands as root, as they have the potential to modify critical system files and configurations.

Note: Rocky Linux 9 follows the same versioning scheme as RHEL 9, with minor releases (9.1, 9.2, 9.3, and so on) delivered approximately every six months. Each minor release includes updated packages, security fixes, and occasionally new features, while maintaining backward compatibility within the major release. Administrators should plan to apply minor release updates regularly to ensure their systems benefit from the latest security patches and improvements.

This chapter has established the foundation for understanding what Rocky Linux 9 is, where it comes from, and why it exists. With this context in mind, you are prepared to move forward into the practical work of installing, configuring, and administering Rocky Linux 9 systems. The journey from here takes you through the hands-on experience that transforms theoretical knowledge into practical skill, and Rocky Linux 9 is an excellent platform on which to build that expertise.

Chapter 2: Installation and Initial Setup

The journey into Rocky Linux 9 administration begins with a solid foundation, and that foundation is built during the installation process. While installing an operating system might seem straightforward, the decisions you make during this phase have lasting consequences on system performance, security posture, and administrative overhead. Every choice, from the partitioning scheme to the software selection, shapes the environment you will manage for months or even years to come. This chapter walks you through every critical step of installing Rocky Linux 9, configuring it for first use, and verifying that the system is ready for production workloads. Whether you are deploying Rocky Linux on bare metal hardware, inside a virtual machine, or across a fleet of enterprise servers, the principles and practices covered here will serve as your authoritative guide.

Rocky Linux 9 carries forward the tradition of enterprise-grade stability that administrators have relied upon for decades. As a community-driven, bug-for-bug compatible rebuild of Red Hat Enterprise Linux, Rocky Linux provides the same binary compatibility, the same package ecosystem, and the same lifecycle guarantees that organizations demand. Understanding the installation process deeply ensures that you can reproduce environments consistently, troubleshoot issues that arise during deployment, and tailor systems to meet specific organizational requirements.

System Requirements and Pre-Installation Planning

Before downloading a single ISO file, you must understand the hardware and resource requirements that Rocky Linux 9 demands. Planning ahead prevents wasted time and ensures that the target system can support the workloads you intend to run.

The minimum and recommended system requirements for Rocky Linux 9 are outlined in the following table. These values represent the baseline for a functional system and the recommended specifications for a comfortable administrative experience.

Resource	**Minimum Requirement**	**Recommended Specification**	**Notes**
Processor	64-bit x86_64 or ARM (aarch64)	Multi-core 64-bit processor	Rocky Linux 9 dropped support for 32-bit architectures entirely
RAM	1.5 GB	4 GB or more	Graphical installation requires at least 1.5 GB; server workloads benefit from significantly more
Disk Space	10 GB	20 GB or more	This is for a minimal installation; full environments with development tools need 40 GB or more
Network	Not strictly required	Gigabit Ethernet adapter	Network access enables repository updates and remote management during installation

Display	VGA-capable display	1024x768 or higher resolution	Required for graphical installer; text mode works without a display
Optical/USB	Bootable USB or DVD drive	USB 3.0 for faster media creation	Network boot via PXE is also fully supported

Planning also involves deciding the installation type. Rocky Linux 9 offers several ISO images, and choosing the correct one depends on your deployment scenario.

ISO Type	**Approximate Size**	**Description**
Boot ISO (boot.iso)	Approximately 900 MB	Contains only the installer and requires a network connection to download packages during installation
Minimal ISO	Approximately 1.7 GB	Includes a minimal set of packages for a basic server installation without a graphical desktop
DVD ISO	Approximately 9 GB	Contains the full repository of packages, enabling offline installation with all software groups available

For enterprise deployments where network connectivity during installation cannot be guaranteed, the DVD ISO is the safest choice. For environments with reliable network infrastructure, the Boot ISO keeps your installation media small and ensures you install the latest packages directly from the repository.

Obtaining and Verifying the Installation Media

Rocky Linux installation images are available from the official Rocky Linux website at rockylinux.org. The download page presents options for different architectures and ISO types. Always download from official mirrors or the primary download page to ensure authenticity.

After downloading, verifying the integrity of the ISO file is a critical security practice. Rocky Linux provides SHA256 checksums and GPG signatures for every release. To verify the checksum on an existing Linux system, you would use the following command.

```
sha256sum Rocky-9.3-x86_64-dvd.iso
```

Compare the output of this command against the checksum published on the Rocky Linux download page. If the values match exactly, the file is intact. If they differ, the download was corrupted or tampered with, and you should download the file again.

For GPG verification, which provides a stronger guarantee of authenticity, you first import the Rocky Linux signing key and then verify the signature file.

```
gpg --import RPM-GPG-KEY-Rocky-9
gpg --verify CHECKSUM.sig CHECKSUM
```

The output should indicate a good signature from the Rocky Linux Release Engineering team. This two-step verification process, checking both the checksum and the GPG signature, represents best practice for any enterprise deployment.

Creating bootable installation media from the verified ISO depends on your platform. On a Linux system, the `dd` command writes the ISO directly to a USB drive.

```
dd if=Rocky-9.3-x86_64-dvd.iso of=/dev/sdb bs=4M status=progress
oflag=sync
```

Note: The `of` parameter must point to the raw device (such as `/dev/sdb`), not a partition (such as `/dev/sdb1`). Double-check the target device with the `lsblk` command before executing `dd`, as this operation destroys all existing data on the target device.

On Windows systems, tools like Rufus or balenaEtcher can create bootable USB media from the ISO file. On macOS, the `dd` command works similarly, though the device path follows the `/dev/diskN` convention.

Walking Through the Installation Process

Boot the target system from the installation media. The initial boot screen presents several options. Select "Install Rocky Linux 9" to launch the Anaconda installer, which is the same installer used by Red Hat Enterprise Linux and other distributions in the same family.

The Anaconda installer uses a hub-and-spoke model, meaning that all configuration categories are presented on a single summary screen, and you can address them in any order. This design is intentional and allows administrators to jump directly to the sections that matter most for their deployment.

Language and Locale Selection

The first screen prompts you to select the language used during installation. This choice also sets the default system locale, though you can change it later. For most enterprise deployments, English (United States) is the standard choice, as it ensures log messages and documentation references align with the majority of available resources.

Installation Destination

This is arguably the most consequential decision in the entire installation process. Clicking on "Installation Destination" allows you to select the target disk and configure the partitioning scheme.

Rocky Linux 9 offers automatic partitioning, which creates a reasonable default layout, and custom partitioning, which gives you full control. For production servers, custom partitioning is strongly recommended because it allows you to separate system, application, and data volumes according to organizational standards.

A well-designed partition layout for a Rocky Linux 9 server might look like the following.

Mount Point	Suggested Size	Filesystem Type	Purpose
/boot	1 GB	xfs	Stores kernel images and bootloader files; kept small and separate for reliability
/boot/efi	600 MB	vfat	Required for UEFI systems; stores the EFI bootloader
/ (root)	20 GB	xfs	Contains the operating system, libraries, and system binaries
/home	10 GB or more	xfs	User home directories; separate partition prevents users from filling the root filesystem
/var	15 GB or more	xfs	Variable data including logs, spool files, and package caches; grows unpredictably
/var/log	10 GB	xfs	Isolating logs prevents runaway logging from consuming space needed by other services
/tmp	5 GB	xfs	Temporary files; isolating this partition allows mounting with noexec for security
swap	2 to 4 GB	swap	Virtual memory; size depends on RAM and workload characteristics

Rocky Linux 9 uses LVM (Logical Volume Manager) by default when automatic partitioning is selected. LVM is highly recommended even for custom layouts because it allows you to resize volumes, create snapshots, and manage storage flexibly after installation.

Note: The XFS filesystem is the default for Rocky Linux 9 and is optimized for large files and high-throughput workloads. While ext4 is also supported, XFS provides better performance characteristics for most enterprise scenarios.

Software Selection

The Software Selection screen determines which packages and package groups are installed. Rocky Linux 9 offers several base environments.

Base Environment	Description	Typical Use Case
Server with GUI	Full graphical desktop with server tools	Workstations and servers requiring a graphical interface
Server	Command-line server environment with essential tools	General-purpose servers where a GUI is unnecessary
Minimal Install	Bare minimum packages for a functional system	Containers, appliances, and highly customized deployments
Workstation	GNOME desktop environment for end-user computing	Developer workstations and desktop deployments
Custom Operating System	Almost nothing installed; maximum customization	Advanced administrators who want complete control over every installed package

For most server deployments, the "Server" or "Minimal Install" base environment is appropriate. The principle of least privilege applies to software as well: install only what you need, and add additional packages later as requirements emerge. Each additional package represents a potential attack surface, so minimalism improves security.

Additional software groups can be selected alongside the base environment. Common add-ons include "Development Tools" for compiling software, "System Tools" for administrative utilities, and "Security Tools" for auditing and hardening.

Network and Hostname Configuration

Click on "Network & Host Name" to configure network interfaces. Toggle the network adapter to the "ON" position to activate it. For servers, configuring a static IP address is standard practice. Click "Configure" next to the adapter and set the following parameters under the IPv4 Settings tab, changing the method from "Automatic (DHCP)" to "Manual."

Parameter	Example Value	Description
IP Address	192.168.1.100	The static address assigned to this server
Netmask	255.255.255.0	Defines the network boundary
Gateway	192.168.1.1	The default route for traffic leaving the local subnet
DNS Servers	8.8.8.8, 8.8.4.4	Name resolution servers; use internal DNS servers in enterprise environments

Set the hostname in the field at the bottom of the screen. Use a fully qualified domain name (FQDN) such as `rocky-server01.example.com`. A meaningful hostname simplifies identification in logs, monitoring systems, and network management tools.

Date, Time, and NTP Configuration

Accurate timekeeping is essential for log correlation, certificate validation, authentication protocols, and scheduled tasks. Under "Time & Date," select the appropriate time zone and enable Network Time Protocol (NTP) by toggling the "Network Time" switch. Rocky Linux 9 uses Chrony as its NTP implementation, and it is configured automatically when NTP is enabled during installation.

Root Password and User Creation