AlmaLinux

- AlmaLinux 9 – Basic To-Dos nach Installation
- CloudStack Management Server (Standalone)
- MariaDB
    - Step-by-Step Guide to Installing MariaDB on AlmaLinux 9

AlmaLinux 9 – Basic To-Dos nach Installation

Dieser Artikel beschreibt die grundlegenden Schritte nach der Installation eines **AlmaLinux 9 Servers**.
Er dient als universelle Basis-Konfiguration für neue Systeme (z. B. Webserver, Management-Server, Backup-Node oder CloudStack-Host).

⚙ System aktualisieren

> AlmaLinux 9 verwendet **DNF** als Paketmanager (Nachfolger von `YUM`).

Unterschied zwischen DNF und YUM

Punkt	YUM	DNF
Basis	Alte Generation (RHEL ≤ 7)	Moderne Version (ab RHEL 8/9)
Geschwindigkeit	Langsamer, kein Caching	Schneller, mit Caching & besserem Dependency-Resolver
API-Fähigkeit	Keine	Vollständige Python-API
Kompatibilität	Alte Syntax	Voll kompatibel zu YUM-Befehlen (`dnf = yum`)

Kurz gesagt: `dnf` **ersetzt** `yum` **vollständig.**
Du kannst auf AlmaLinux 9 zwar noch `yum` eingeben – intern wird aber `dnf` ausgeführt.

Befehle

```
# Systempakete aktualisieren
dnf update -y

# (Optional) Neustart, falls Kernel-Update
reboot
```

Hostname setzen

> Der Hostname identifiziert den Server eindeutig im Netzwerk und wird u. a. für Zertifikate, Logs und SSH-Sessions verwendet.

Hostname anzeigen

```
hostnamectl
```

Hostname ändern

```
# Beispiel
hostnamectl set-hostname server01.example.local
```

Verifizieren

```
hostnamectl status
# oder
hostname
```

Hinweis

- Der Befehl schreibt den neuen Namen direkt in `/etc/hostname`.
- Ein Neustart ist **nicht nötig**, die Änderung greift sofort.
- Wenn du einen FQDN (z. B. `server01.example.local`) verwendest, sollte der Hostname auch in `/etc/hosts` eingetragen werden:

```
127.0.0.1   localhost
192.168.1.10 server01.example.local server01
```

Benutzer & SSH-Absicherung

```
# Neuen Benutzer anlegen
useradd tobias
passwd tobias

# In die sudo-Gruppe aufnehmen
```

```
usermod -aG wheel tobias
```

SSH absichern:

```
# Root-Login deaktivieren
sed -i 's/^#PermitRootLogin.*/PermitRootLogin no/' /etc/ssh/sshd_config

# (Optional) Passwort-Login deaktivieren, wenn SSH-Key genutzt wird
sed -i 's/^#PasswordAuthentication.*/PasswordAuthentication no/' /etc/ssh/sshd_config

# Dienst neu starten
systemctl restart sshd
```

□□ 4□ Basis-Tools installieren

```
dnf install -y vim nano htop git curl wget net-tools \
  bash-completion unzip tar zip rsync chrony
```

□□ 5□ Firewall aktivieren

```
systemctl enable --now firewalld
firewall-cmd --add-service=ssh --permanent
firewall-cmd --reload
```

□□ 6□ Netzwerk konfigurieren

```
nmcli device status
nmcli connection show
```

Beispiel – statische IP

```
nmcli con mod "ens192" ipv4.addresses 192.168.1.10/24
nmcli con mod "ens192" ipv4.gateway 192.168.1.1
nmcli con mod "ens192" ipv4.dns "1.1.1.1 8.8.8.8"
nmcli con mod "ens192" ipv4.method manual
```

```
nmcli con up "ens192"
```

□□ 7□ SELinux & Auto-Updates

```
# SELinux-Status prüfen
sestatus

# (Optional) auf Permissive Mode setzen
setenforce 0
sed -i 's/^SELINUX=enforcing/SELINUX=permissive/' /etc/selinux/config
```

Automatische Sicherheitsupdates aktivieren:

```
dnf install -y dnf-automatic
systemctl enable --now dnf-automatic.timer
```

□□ 8□ Cockpit aktivieren (Web-GUI)

```
dnf install -y cockpit
systemctl enable --now cockpit.socket
firewall-cmd --add-service=cockpit --permanent
firewall-cmd --reload
```

> Zugriff: https://server-ip:9090
> Login mit einem Benutzer aus der `wheel`-Gruppe.

□□ 9□ EPEL & CRB Repository aktivieren

```
dnf install -y epel-release
dnf config-manager --set-enabled crb
```

□□ □□ Abschluss

- System ist aufgesetzt
- Benutzer, Sicherheit & Netzwerk konfiguriert
- Bereit für Rollenzuweisung (z. B. Web, DB, RMM, CloudStack)

Zusammenfassung wichtiger Befehle

Zweck	Befehl
Hostname setzen	`hostnamectl set-hostname <name>`
Systemupdate	`dnf update -y`
Benutzer anlegen	`useradd <name>`
Firewall starten	`systemctl enable --now firewalld`
Cockpit aktivieren	`systemctl enable --now cockpit.socket`

Last Update: {{DATE}}
Autor: CoreStream Group – Internal KB

CloudStack Management Server (Standalone)

Prepare the Operating System

The OS must be prepared to host the Management Server using the following steps. These steps must be performed on each Management Server node.

1. Log in to your OS as root.
2. Check for a fully qualified hostname.

   ```
   hostname --fqdn
   ```

   This should return a fully qualified hostname such as “management1.lab.example.org”. If it does not, edit /etc/hosts so that it does.
3. Make sure that the machine can reach the Internet.

   ```
   ping cloudstack.apache.org
   ```

4. Turn on NTP for time synchronization.
   Note
   An NTP daemon is required to synchronize the clocks of the servers in your cloud.
   Install chrony.
   In RHEL or CentOS:

   > ```
   > $ yum install chrony
   > ```

   n RHEL or CentOS:
   Yum repository information is found under `/etc/yum.repos.d`. You’ll see several `.repo` files in this directory, each one denoting a specific repository.
   To add the CloudStack repository, create `/etc/yum.repos.d/cloudstack.repo` and insert the following information.
   In the case of RHEL being used, you can replace ‘centos’ by ‘rhel’ in the value of baseurl

   ```
   [cloudstack]
   name=cloudstack
   baseurl=http://download.cloudstack.org/centos/$releasever/4.21/
   enabled=1
   ```

```
gpgcheck=0
```

Now update your local apt cache.

```
sudo apt update
```

Your DEB package repository should now be configured and ready for use.

Install on CentOS/RHEL

```
yum install cloudstack-management
```

Step #2: Install MariaDB

AlmaLinux 9 provides a MariaDB package directly in its official repositories. To install the MariaDB server with all necessary dependencies, execute the following command:

> *sudo dnf install mariadb-server -y*

Open the MySQL configuration file. The configuration file is `/etc/my.cnf.d/mariadb-server.cnf`.
Insert the following lines in the `[mysqld]` section.
You can put these lines below the datadir line. The max_connections parameter should be set to 350 multiplied by the number of Management Servers you are deploying. This example assumes one Management Server.

```
innodb_rollback_on_timeout=1
innodb_lock_wait_timeout=600
max_connections=350
log_bin=mysql-bin
binlog_format=ROW
```

Start and Enable the MariaDB Service

Once you have installed the MariaDB package, start and enable the service to launch automatically on the system boot.

> *# sudo systemctl start mariadb*
>
> # *sudo systemctl enable mariadb*

Run the following command to secure your installation. You can answer “Y” to all questions.

```
mysql_secure_installation
```

loudStack can be blocked by security mechanisms, such as SELinux. Disable SELinux to ensure + that the Agent has all the required permissions.
Configure SELinux (RHEL and CentOS):

1. Check whether SELinux is installed on your machine. If not, you can skip this section. In RHEL or CentOS, SELinux is installed and enabled by default. You can verify this with:

   ```
   rpm -qa | grep selinux
   ```

   1. Set the SELINUX variable in `/etc/selinux/config` to “permissive”. This ensures that the permissive setting will be maintained after a system reboot.
      In RHEL or CentOS:

      ```
      vi /etc/selinux/config
      ```

      Change the following line

      ```
      SELINUX=enforcing
      ```

      to this:

      ```
      SELINUX=permissive
      ```

   2. Set SELinux to permissive starting immediately, without requiring a system reboot.

      ```
      setenforce permissive
      ```

      In a production environment, selinux should be set to enforcing and the necessary selinux policies are created to allow the services to run.

MariaDB

Step-by-Step Guide to Installing MariaDB on AlmaLinux 9

Even new businesses or applications can generate enough data to require a dedicated data handling and management system. This system should efficiently manage, store, and retrieve data for various processes and components.

MariaDB, a high-performance and versatile relational database management system (RDBMS), is a leading choice for its speed, reliability, and ease of use.

By leveraging MariaDB, businesses can streamline their data operations, enhance efficiency, and make informed decisions based on accurate and accessible data.

In this tutorial, we will discuss how to install MariaDB on AlmaLinux 9, a community-driven version of the enterprise-level Red Hat Enterprise Linux (RHEL).

Let us quickly examine the prerequisites and then go into the details of the installation process.

The Prerequisites

Before you dive into the installation process, ensure you have the following:

- You have AlmaLinux 9 installed and running on your machine.
- A user account with sudo or root privileges

Let us discuss the steps in detail.

Step #1: Update System Package Repository

Before the installation process, it’s crucial to update your system package repository to ensure all packages are up-to-date.

> *# sudo dnf update -y*

Here, the **-y** flag automatically answers **YES** to any prompts, streamlining the update process.

The command ensures you have the latest security updates and software improvements.

Step #2: Install MariaDB

AlmaLinux 9 provides a MariaDB package directly in its official repositories. To install the MariaDB server with all necessary dependencies, execute the following command:

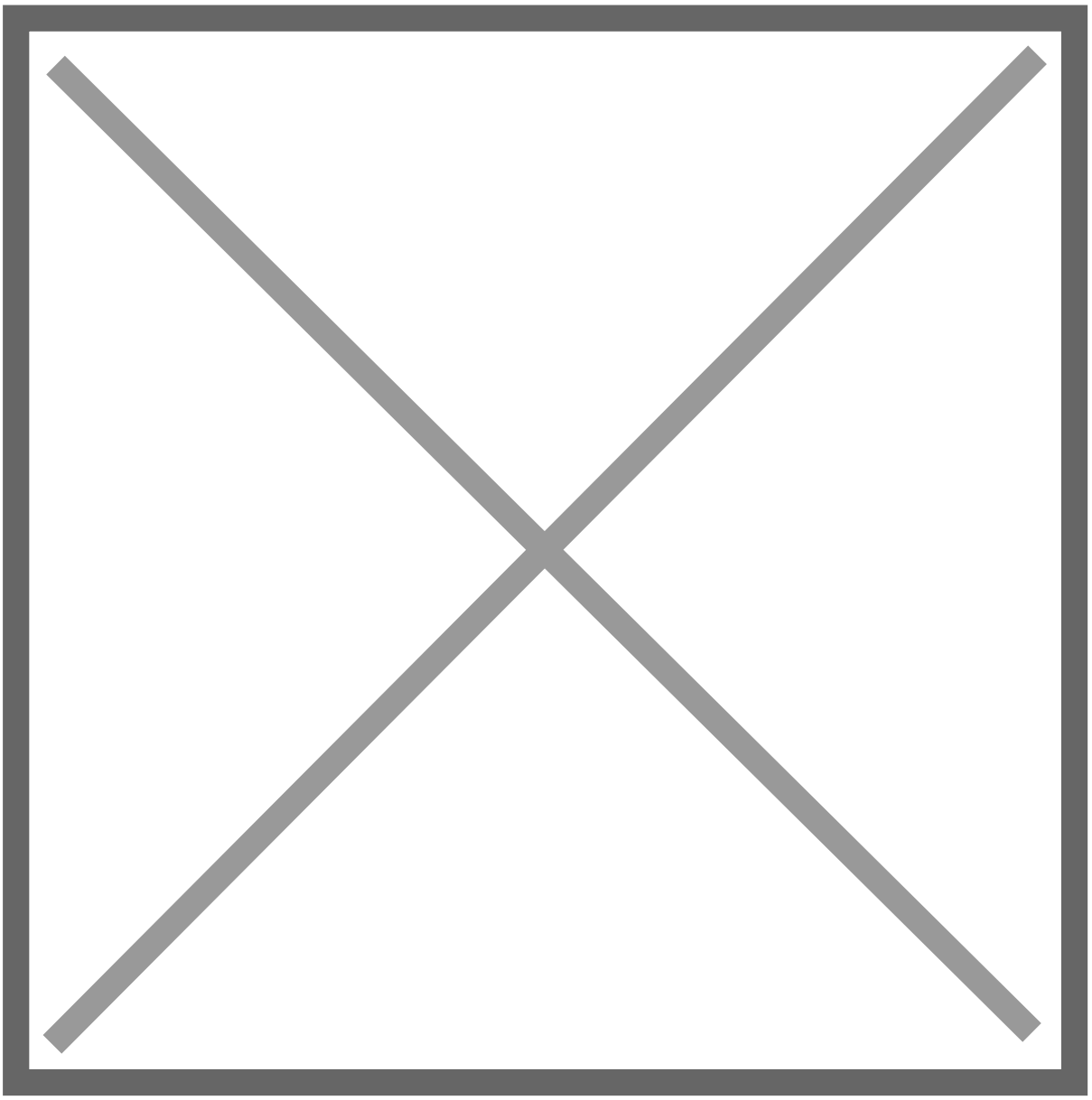
> *# sudo dnf install mariadb-server -y*

Step #3: Start and Enable the MariaDB Service

Once you have installed the MariaDB package, start and enable the service to launch automatically on the system boot.

> *# sudo systemctl start mariadb*
>
> *# sudo systemctl enable mariadb*

Press enter or click to view image in full size



Step #4: Secure MariaDB Installation

It’s important to enhance the security of your MariaDB installation by setting a strong root password, removing anonymous users, disallowing remote root login, and removing test databases.

Use the security script provided by MariaDB to secure the installation. Start by launching the script with this command:

> *# sudo mysql_secure_installation*

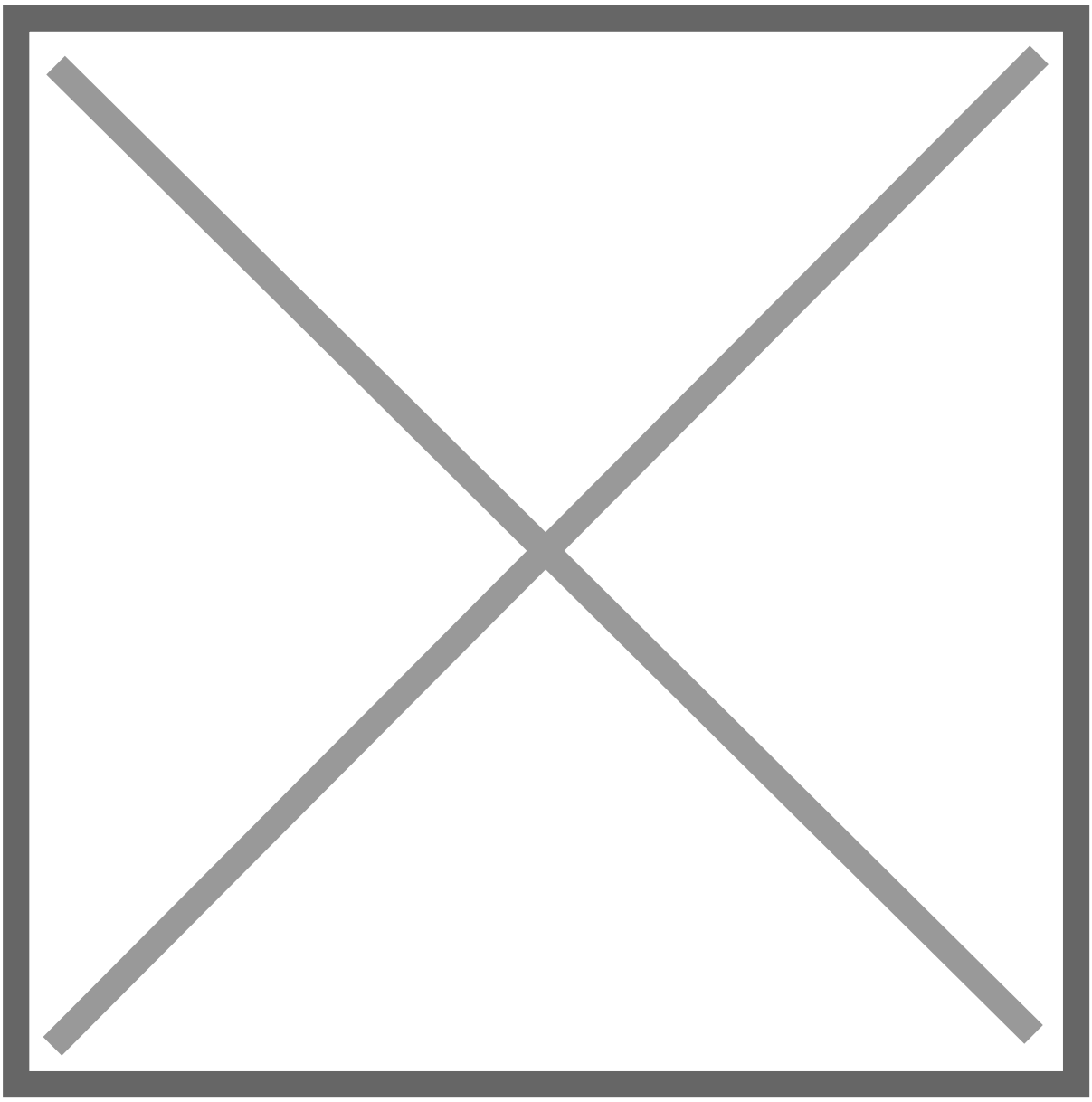
Follow the prompts to set a root password, remove anonymous users, disallow root login remotely, and remove test databases. This step enhances the security of your database server.

Step #5: Configure MariaDB (Optional)

If you need to customize MariaDB settings based on your specific requirements and system resources, open the configuration file in a text editor like Vi or Nano.

> *# sudo vi /etc/my.cnf.d/mariadb-server.cnf*

Press enter or click to view image in full size



You can adjust settings like **max_connections** and **innodb_buffer_pool_size** according to your system’s capabilities and your requirements.

Step #6: Test MariaDB

To verify if MariaDB is installed appropriately and running, connect to the MariaDB server using the following command:

> *# mysql -u root -p*

Press enter or click to view image in full size