aws

Migration Guide

# AWS Elemental Server

# AWS Elemental Server: Migration Guide

# Table of Contents

# About this guide

This guide describes how to upgrade AWS Elemental Server and AWS Elemental Conductor File. It describes how to perform an upgrade to version 2.18 or higher of the AWS Elemental software, and how to downgrade to a version below 2.18.

This special guide exists because this software upgrade requires that you install the RHEL 9 version of the Linux operating system. A software downgrade requires that you re-install RHEL 7 or CentOS 7.

**Important**

We strongly recommend that you test the entire migration procedure in your lab. This strategy lets you test the migration process itself, and test the entire workflow on the new software.

**Note**

For assistance with your AWS Elemental appliances and software products, see the AWS Elemental Support Center.

# Migrating to AWS Elemental Server version 2.18

This procedure describes how to upgrade an appliance that is running AWS Elemental Server from a version below 2.18.0 and migrate them to version 2.18.0 or higher.

> **Important**
>
> We strongly recommend that you test the entire migration procedure in your lab. This strategy lets you test the migration process itself, and test the entire workflow on the new software.

In this procedure, we show how to upgrade AWS Elemental Server version 2.17.5 to version 2.18.0. Modify the commands you enter to match your versions.

**Topics**



# Step A: Get ready for AWS Elemental Server migration

## Read the essential notes

Refer to the essential notes in the AWS Elemental Server Release Notes to identify key changes to the behavior of AWS Elemental Server.

## Modify your automation system for HTTPS

After a node has been migrated, it uses HTTPS. By default, the nodes are set up with self-signed certificates. Make sure of the following points:

- You might need to change your automation system to use HTTPS.

## Verify installer type

Use a software installer that aligns with your AWS Elemental Server type: GPU-accelerated or CPU-only.

Use the same type of installer for your Conductor File. For example, if your AWS Elemental Server has a GPU, use the GPU installer for your Conductor File.

To determine whether you have a GPU-accelerated or CPU-only system, run the following command on your AWS Elemental Server:

**lspci | grep NVIDIA**

If the output is empty, your system is CPU-only.

If the output is similar to the following, your system is GPU-accelerated:

```
VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
```

## Create a boot USB drive

On Dell hardware, you have the option to install RHEL 9 by using a boot USB drive or by using iDRAC (for Dell). (Note that SuperMicro hardware, you can only install RHEL 9 by using IPMI.)

If you want to use a boot USB drive with Dell, you should make the drive now. You might want to make several drives, depending on how many people will be performing the migration tasks.

To create a boot drive, follow these steps:

1. Obtain the RHEL 9 `.iso` file from AWS Elemental Software Download page.

   Find the AWS Elemental product and version they are planning to use. The appropriate ISO file appears beside that version.
2. At your workstation, use a third-party utility (such as PowerISO or ISO2USB) to create a bootable USB drive from your `.iso` file. For help, see the knowledge base article Creating Bootable Recovery (kickstart) Media.

### Verify space on the node

As part of the upgrade, you create a backup of the data on the node. You must make sure that you have enough free space for the backup. Follow these guidelines:

- The backup for a freshly kickstarted and licensed appliance generates a small backup directory zipped version of the backup (`<hostname>_lifeboat-archive.zip`).
- Your configuration will generate larger files because of the data that you create, so review available space before starting.
- Check the contents of the `/home` partion, and clear out old files, unnecessary files, and old installers.

## Step B: Prepare the AWS Elemental Server node for migration

### Upgrade to the latest 2.17 minor version

To upgrade to version 2.18.0 or higher, the software currently installed on the node must be version 2.17.0 or higher. For instructions about how to upgrade to that version, see the https://docs.aws.amazon.com/elemental-onprem/latest/pdf/elemental_server_upgrade_guide_2.17.pdfServer 2.17 Upgrade Guide.

### Verify access to the BMC on the appliances

Make sure that you have access to the BMC on each appliance:

- On a Dell server, make sure that iDRAC is installed and that you can start it.
- On an SMC server, make sure that IPMI is installed and that you can start it.

You can install iDRAC or IPMI even when AWS Elemental Server is running events.

### Note the network adapter for the management interface

Make a note of the management network device listed in the web UI under **Settings**, **Network**, **Network Settings**, **Current Settings**, **Network Devices**. By default, eth0 is the management network device, but this may differ on your system. You'll need to know this adapter later during the migration process.

### Update firmware

Both the BIOS firmware and the BMC firmware (IPMI for SuperMicro, iDRAC for Dell) must be at the latest versions available from the manufacturer. They must be at the latest versions before you can set the boot mode to UEFI.

We recommend that you update the firmware on all your nodes at the same time. We also recommend that you perform this update during a maintenance window. After you install the firmware, you must reboot the node. For more information, see the section called "Update firmware".

### Move custom files

You might have custom files in `/opt/elemental_se/scripts` on the node. These are files that you created. They aren't part of the installation of the AWS Elemental Server software, and they aren't backed up and restored.

Copy these files to storage off the node, so that you can copy the files back to the node after you've upgraded it.

## Step C: Stop running jobs

Stop, or complete, all the running jobs on the node.

## Step D: Create an AWS Elemental Server backup

You must create a backup of the data on the node. You back up data using the special lifeboat script.

> **Important**
>
> The lifeboat script creates a backup of multiple files that are relevant to the AWS Elemental software. These files might include credentials and other sensitive system information. Handle the backup according to your organization's best practices for handling sensitive data.

## About the backup process

The script backs up the following data:

- Licenses.
- Network settings for the node, including Ethernet configurations, DNS information, and host addresses.
- Timecode configuraton such as NTP, PTP, and chronyd.
- Firewall settings.
- SSL certificates.
- Optionally, the user credentials used in various components on the node. It is convenient to include these credentials, if your organization's policies allow them to be handled in this way.
- Configuration files for features of the AWS Elemental software.
- Remote storage mounts.
- Node data. Data such as data about the events and MPTSes.

## Step 1: Download the lifeboat script

You must copy the lifeboat script onto every node.

1. Download the latest version of the lifeboat script from https://a.co/ElementalRHEL9Lifeboat to your laptop. The lifeboat file is called `elemental_lifeboat_el.tar`.

   > **Important**
   >
   > Download the script just before you are ready to create the backup. AWS Elemental is continually making improvements to the script, therefore you want to make sure that you always have the latest version.

2. Copy the lifeboat file to the `/home/elemental` directory.
3. From the Linux prompt, use the *elemental* user to start a remote terminal session with the node. Don't log in as sudo.
4. Untar the lifeboat file.

```
[elemental@hostname ~]$ cd /home/elemental && tar xvf elemental_lifeboat_el9.tar
```

# Create the backup

> **Important**
>
> Make sure that you have stopped the node. We recommend that you don't run the script on an active node. The script temporarily stops elemental_se and httpd services.

Enter the backup command as follows:

```
[elemental@hostname ~]$ ./lifeboat.sh --backup --include-creds
```

Where `--include-creds` (optional) includes the following credentials in the backup: SSH, AWS, SMB/CIFs.

**Results of the backup**

The script creates the following assets:

- Asset 1. One version of the data that is compatible with 2.18.0 or later. When you restore the backup after you've installed RHEL 9, the lifeboat script will automatically select and copy over this version.
- Asset 2. One version of the data that is compatible with 2.17.x and earlier. You might later decide to downgrade a node back to a version below 2.18.0. When you restore the backup after you've installed RHEL 7 or CentOS 7, the lifeboat script will automatically select and copy over this version.
- Asset 3. An MD5 checksum of the contents of asset 3.
- Asset 4. A SHA1 checksum of the content of asset 3.

The script also creates the following files:

- File 1. A file that contains assets 1 and 2. The file has this name, where `hostname` is the name of the current node:

  `<hostname>_lifeboat-archive.zip`
- File 2. A file that contains assets 3 and 4. The file has this name, where `hostname` is the name of the current node:

`<hostname>_lifeboat-archive_export-checksum.txt`

- File 3. A file that contains assets 1, 2, 3 and 4. The file is stored on the current node at this location:

  `/opt/upgrade-backups/system-backup.tar.gz`

### Verify the backup

Verify the integrity of the backup archive. This step is optional but we strongly recommend that you follow it because the restore operation that you later perform might fail if the backup file is corrupted.

You verify the integrity by comparing the checksum that the backup script creates to the checksum that you perform on the `<hostname>_lifeboat-archive_export-checksum.txt` file. You can compare an MD5 or a SHA1 checksum.

1. Enter the `cat` command to view the checksums currently listed in the checksum file.:

   ```
   ~]$ cat <hostname>_lifeboat-archive_export-checksum.txt
   ```

   The `cat` command simply displays the file contents on your screen. For example:

   ```
   md5sum
         d41d8cd98f00b204e9800998ecf8427e
   sha1sum
       e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   ```

2. Now run a checksum command on the `lifeboat-archive.zip` file. For example:

   ```
   ~]$ md5sum /home/elemental/<hostname>_lifeboat-archive.zip
   ```

   Or

   ```
   ~]$ sha1sum /home/elemental/<hostname>_lifeboat-archive.zip
   ```

3. Compare the results from step 1 to the results from step 2. If the checksums don't match, copy the archive file again.

### Store the backup archive

Copy the `<hostname>_lifeboat-archive.zip` file to storage off the node, so that you can copy it back to the node when you want to perform the restore operation.

> **Important**
>
> The lifeboat script creates a backup of multiple files that are relevant to the AWS Elemental software. These files might include credentials and other sensitive system information. Handle the backup according to your organization's best practices for handling sensitive data.

# Step E: Switch boot mode to UEFI

RHEL 9 requires that the boot mode for the appliance is UEFI. You can change the boot mode from BIOS (or Legacy mode) to UEFI.

### Topics



## Switch to UEFI on a Dell

There are three ways to switch the boot mode from Legacy mode to UEFI.

### Topics



### Switch using the IDRAC user interface

iDRAC is a system for controlling Dell servers remotely. It is already installed and enabled on the Dell server. However, you might need to configure it. For more information about configuring iDRAC, see the official Dell iDRAC User Guide.

This procedure is identical to the procedure for switching to BIOS, except that you choose **UEFI** instead of **BIOS**.

1. Log into the iDRAC user interface as an administrative user.
2. On the iDRAC menu, choose **Configuration, then BIOS Settings**, then **Boot Settings.**
3. On the **Boot Mode** line, change the **Current Value** from **BIOS** to **UEFI**.
4. Scroll down to the **Apply** button and choose that button. The **Pending Value** changes to **UEFI**.
5. Scroll down to the bottom of the page and choose **Apply And Reboot**.

The system reboots. UEFI is now enabled.

## Switch using RACADM

You can switch to UEFI mode by logging into RACADM, which is the iDRAC command line interface.

This procedure is identical to the procedure for switching to BIOS, except that you specify **UEFI** instead of **BIOS**.

1. Start a Linux session and log into the iDRAC command line interface as a Linux Admin user. For example:

   ```
   ssh ADMIN@<iDRAC hostname or IP>
   ```

   The IDRAC command line interface appears, with the **racadm>>** prompt.
2. To verify that the current boot environment is BIOS, enter this command:

   ```
   get BIOS.biosBootSettings.BootMode
   ```

   If the environment is BIOS, a message similar to the following appears:

   ```
   [Key=BIOS.Setup.1-1#biosBootSettings]
   BootMode=Bios
   ```
3. Set the **BIOS settings** to **UEFI**:

   ```
   set BIOS.BiosBootSettings.BootMode Uefi
   ```
4. Apply and reboot:

```
jobqueue create BIOS.Setup.1-1 -r Forced
```

The system reboots. UEFI is now enabled.

### Switch using the F2 boot menu

You can use the boot menu from a direct connection to the server, or through the IDRAC virtual console.

This procedure is identical to the procedure for switching to BIOS, except that you specify **UEFI** instead of **BIOS**.

1. This step applies only if you want to use the virtual console: log into the iDRAC user interface and launch the Virtual Console.
2. Reboot the appliance.

   ```
   sudo reboot
   ```
3. The appliance starts to reboot using BIOS, which is currently enabled.
4. As soon as the reboot starts, repeatedly press **F2** on the keyboard, until the message **Entering System Setup** appears. Then wait for the **System Setup** screen to appear.
5. Choose **System BIOS**, then choose **Boot Settings**.
6. On the **Boot Mode** line, choose **UEFI**.
7. Choose the **Exit** option and follow the prompts to save. At the success message, choose **OK**.

The system reboots. UEFI is now enabled.

## Switch to UEFI on a SuperMicro

To switch the boot mode from BIOS (Legacy mode) to UEFI, you can use the IPMI interface, or you can work when directly connected to the server.

### Step 1: Install Java applet

Decide if you want to use the IPMI management console, or if you plan to connect directly to the server. If you want to use the console, decide if you want to use the Java remote console applet to access the console, or if you want to use HTML5.

If you want to use the IPMI management console and you want to use the Java remote console applet, you must install the applet.

1. Make sure you have the IP address of the IPMI. If you don't have it, connect to the appliance using SSH, then type the following command:

   ```
   sudo ipmiutil lan | grep Param\(3\)
   ```

   The IP address appears in the response. For example:

   ```
   Lan Param(3) IP address: 10 4 130 12
   ```

2. Log in to the IPMI management console via a web browser. Use the ADMIN credentials, with the user name entered in uppercase.
3. From the menu bar, choose **Console Redirection**, then **Launch Console**. The download of a JNLP file starts.
4. When the download is complete, open the applet. The applet is self signed. Typically, this file is already associated with Java so you should just be able to open it directly.
5. Change the security level in the Java control panel in order for the applet to run:
   a. In Windows, open **Control Panel**, **Programs**, and then **Java**.
   b. Click the **Security** tab. Move the slider to the lowest setting: **Medium**.
   c. Click **OK**.

You can now open the remote console window.

## Step 2: Change the mode to UEFI

This procedure is nearly identical to the procedure for switching to BIOS. You change the same fields on the **Setup Utility** screen.

1. From the IPMI management console, sign in to the server as the *elemental* user.
2. Reboot the system:

   ```
   [elemental@hostname]$ sudo reboot
   ```

   The system starts to reboot. The window size might change as the system is rebooting.

3. While the system is rebooting, repeatedly press the **Delete** key on the keyboard (or the **del** button on the virtual keyboard). The **Setup Utility** screen appears.

   You can use these keys to work on the screen:

   - The arrow keys
   - Enter to select
   - ESC to return to the previous screen.

4. On the main menu, choose **Advanced**.
5. In **sSATA Configuration**, look for fields that have one of these values:

   - **BIOS**
   - **DUAL**
   - **Legacy**
   - **Legacy BIOS**

   Change the value to **EFI**. If there are no fields with these values, go to the next step.

6. In **PCIe/PCI/PnP Configuration**, find every field that has one of these values:

   - **BIOS**
   - **DUAL**
   - **Legacy**
   - **Legacy BIOS**

   In each of these fields, change the value to **EFI**.

7. On the main menu, choose **Boot**. In **Boot Mode Select**, change the value from **DUAL** to **UEFI**.
8. Select **F4**. On the **Save & Exit** dialog, choose **Yes**.

# Step F: Install RHEL 9 on a AWS Elemental Server node

**Topics**

# Install on a Dell

You can install RHEL 9 on a Dell chassis either from the iDRAC interface or using a USB stick.

## Install using the iDRAC interface

### Get Ready

1. Make sure that there are no physical USB drives plugged into the system.
2. Make sure that you are at a workstation that has direct access to the network that the iDRAC interface is on. (So don't use a VPN connection.)
3. Log into iDRAC through the web interface. Use an administrative username and password.
4. Launch the Virtual Console. On the main menu, select **Virtual Media**. On the next screen, select **Connect Virtual Media**. The **Virtual Media** screen appears.
5. In the **Map CD/DVD** section, in **Image File**, click **Choose File**. In the window that appears, navigate to the kickstart .iso file, select it, and click **Open**. The **Image File** field in the **Virtual Media** screen now specifies the image file.
6. Click **Map Device**. Then at the bottom of the screen, click **Close**.

The kickstart .ISO image file is now mapped to the virtual CD/DVD drive.

1. On the main menu of the Virtual Console, click **Boot**. On the **Boot Controls** list, click **Virtual CD/DVD/ISO**. Then at the **Confirm Boot Action** prompt, click **Yes**.
2. On the main menu of the Virtual Console, click **Power**, then click **Reset System (warm boot)**, and at the **Confirm** prompt, click **Yes**.

The system reboots into the kickstart .iso. Lines of text appear, and finally the prompt **Enter the server complete hostname** appears.

### Install the operating system

1. At the **Enter the server complete hostname** prompt, enter the hostname that already applies to this node, then press **Enter**. The installation starts.
2. When the installation is complete, press **Enter** to quit and reboot.
3. You can now install any third-party packages. AWS Elemental maintains an RPM repository for use with RHEL 9. The repository contains the following types of third-party packages:

- Packages that are stored in the Red Hat BaseOS repository, and that are required to run AWS Elemental software.
- Packages that are stored in the Red Hat AppStream repository, that aren't required but that you want to include.

For more information about the packages that you must obtain from the AWS Elemental RPM repository, and for instructions about configuring the repository, see the knowledge base article Advisory.

### USB stick

1. Make sure that you have created a boot USB drive. See the section called “Create a boot USB drive”.
2. Insert the USB drive into an available USB port. You might need to press **F2** while booting in order to select the boot device. The recovery (kickstart) screen appears.
3. Enter the hostname that already applies to this node, then press **Enter**. The installation starts.
4. When the installation is complete, remove the USB drive from the system and store it in a secure location.
5. Then on the screen, press the reboot button shown or press the **Enter** key.
6. You can now install any third-party packages. AWS Elemental maintains an RPM repository for use with RHEL 9. The repository contains the following types of third-party packages:
   - Packages that are stored in the Red Hat BaseOS repository, and that are required to run AWS Elemental software.
   - Packages that are stored in the Red Hat AppStream repository, that aren't required but that you want to include.

   For more information about the packages that you must obtain from the AWS Elemental RPM repository, and for instructions about configuring the repository, see the knowledge base article Advisory.

## Install on an SMC

You install RHEL 9 on an SMC chassis from the IPMI interface.

1. Install the Java applet and change the security level, if necessary. For information, see the section called “Step 1: Install Java applet”.
2. Make sure that there are no physical USB drives plugged into the system.
3. Make sure that you are at a workstation that has direct access to the network that the IPMI interface is on.

   > **Note**
   >
   > Don't use a VPN connection.

4. Copy the ISO file for RHEL 9 to your laptop.
5. Open the IPMI remote console viewer. On the main menu, choose **Virtual Media** or **Media**, then choose **Virtual Storage/Virtual Media Wizard**.
6. Choose **CD/ISO media** and browse to the ISO that you want to use. Choose **Connect/Plug in**.
7. Reboot the system. The image should start to boot.

   If the image does not start to boot, click the **F11** key while the splash screen is displaying. Then when the **Please select boot device** prompt appear, choose **UEFI: Virtual CDROM**. Move this item to the top of the list by pressing the **+** key repeatedly.
8. The installer starts. At the prompt, enter the hostname of the appliance and press **Enter**. The installation starts and takes 20 to 30 minutes.
9. When the installation completes, press the **Enter** key to reboot.
10. Plug Out the ISO before it reboots or it takes you back into the kickstart menu.
11. You can now install any third-party packages. AWS Elemental maintains an RPM repository for use with RHEL 9. The repository contains the following types of third-party packages:

    - Packages that are stored in the Red Hat BaseOS repository, and that are required to run AWS Elemental software.
    - Packages that are stored in the Red Hat AppStream repository, that aren't required but that you want to include.

    For more information about the packages that you must obtain from the AWS Elemental RPM repository, and for instructions about configuring the repository, see the knowledge base article Advisory.

# Step G: Install worker software on a AWS Elemental Server node

This install procedure isn't the same as the install procedure on a newly obtained appliance (as described in AWS Elemental Server Installation Guide). You don't have to configure the node.

1. From the Linux command line, log in to the worker node. Use the **elemental** user credentials.
2. Check routing table by running the following command:

   ```
   ip r show
   ```

   The system returns something similar to the following:

   ```
   default via 10.x.x.x dev eth0 proto dhcp src 10.x.x.x metric 103
   10.x.x.x/x dev eth1 proto kernel scope link src 10.12.107.43 metric 102
   ...
   ```

   To proceed with the upgrade, your management interface must be listed for the first route. To find which network interface is your system's management interface, see: Note the network adapter for the management interface.
3. If your management interface isn't listed for the first route, you must update the default route.:
4. Run the installer. Use the appropriate command:
   - For GPU versions of the software (for AWS Elemental Server only):

     ```
     [elemental@hostname ~]$ sudo sh ./elemental_production_server_2.18.n.nnnn.run --
     skip-all --start --xeula
     ```

   - For CPU-only versions of the software:

     ```
     [elemental@hostname ~]$ sudo sh ./elemental_production_server_cpu_2.18.n.nnnn.run
      --skip-all --start --xeula
     ```

   Where:

   `--skip-all` skips all the prompts. There is no need to view prompts about configuration because when you restore the database to the node, all the configuration data is copied over and overwrites any configuration data already on the node.

`--start` restarts the software after installation.

`--xeula` skips the display of the license agreement. There is no need to view this prompt because you have previously accepted the agreement.

5. When the installation is complete, restart the node:

```
[elemental@hostname ~] sudo reboot
```

## Step H: Restore the database on a AWS Elemental Server node

You restore data using the same lifeboat script that you used to create the backup.

> **Important**
>
> Make sure that you have stopped the node. Don't run the script on an active node.

1. Download the lifeboat script, following the procedure that you followed when you created the backup.

   > **Important**
   >
   > Make sure that you have latest version of the script. AWS Elemental is continually making improvements to the script.

2. Enter the restore command.

```
[elemental@hostname ~]$ ./lifeboat.sh --restore --import-database
```

   The script tries to extract the version of the backup that is stored in this folder: `/opt/upgrade-backups/system-backup.tar.gz`

   This file was created when you created the backup. The script automatically copied it to this directory. The installation of RHEL 9 should not have deleted this file. Therefore, it should be in this location. If this file doesn't exist or if there is a problem with it, the scripts stops. See the recovery steps below to continue.

> **Important**
>
> If you updated the routing table in earlier, add the `--exclude netscripts-rest` flag when you run the lifeboat.sh script:
>
> ```
> [elemental@hostname ~]$ ./lifeboat.sh --restore --import-database --exclude
>  netscripts-rest
> ```

3. After the restore has succeeded, reboot the node:

```
[elemental@hostname ~]$ sudo reboot
```

### Recovery steps

1. Locate the other copies of the backup and of the checksum files that you should have copied to storage off the node. The files to locate are:
   - `<hostname>_lifeboat-archive.zip`
   - `<hostname>_lifeboat-archive_export-checksum.txt`
2. Copy the files to `/home/elemental`
3. Enter the restore command again:

```
[elemental@hostname ~]$ ./lifeboat.sh --restore
```

This time the script looks for the files that are in `/home/elemental`, and restores those files.

# Document history for migration guide

The following table describes the main changes to this guide.

| Change | Description | Date |
| --- | --- | --- |
| New guide | First release of this guide | June 7, 2024 |