



I want to make an image file that will have a fat32 EFI and a ext4 MX23 partition. The EFI will be 1075 MiB and the MX23 will be 258186 MiB. How do I do that using MX23 Linux?

Sources

Create a FAT file system image on Linux



1

create-efi-linux-image/efi_img_creator at master · ...



2

Dynamically creating an IMG file with a FAT32 par...



3



View 12 more

Answer

Creating a Multi-Partition Image with FAT32 EFI and EXT4 MX23 on MXLinux

Creating an image file with multiple partitions, including a FAT32 EFI partition and an EXT4 MX23 partition, involves several steps. Here's a detailed guide on how to achieve this using MXLinux.

Understanding the Task

We aim to create an IMG file containing two partitions:



Ask a follow-up

☐ Copilot



Home



Discover



Library



- ``fdisk``: For creating and managing disk partitions.
- ``mkfs.fat`` and ``mkfs.ext4``: For formatting FAT32 and EXT4 file systems respectively.
- ``dd``: For creating the image file.

You can install these tools using the following command:

```
sudo apt-get update && sudo apt-get install fdisk mkfs-fat32 mkfs-ext4
```

Step-by-Step Guide

1. Create a Temporary File for the Image

First, create a temporary file that will serve as the base for our image.

```
fallocate -l 259261M /tmp/image.img
```

This command creates a 259261 MiB file named ``image.img`` in the ``/tmp`` directory 1.

2. Create Partition Table and Partitions

Use ``fdisk`` to create a partition table and define our two partitions.

```
sudo fdisk /tmp/image.img <<EOF
n   # New partition
p   # Primary partition
1   # First partition (EFI)
+1075M   # Size of 1075 MiB
n   # New partition
p   # Primary partition
2   # Second partition (MX23)
EOF
```

Copilot



Home



Discover



Library



```
sudo mkfs.fat -F32 /tmp/image.img -n "EFI" <<< $(sudo blkid -s UUID -o value /dev  
sudo mkfs.ext4 /tmp/image.img -F /dev/loop0p2
```

4. Create the Image File

Now, create the final IMG file with the specified partitions.

```
sudo losetup --partscan /tmp/image.img  
sudo partprobe /tmp/image.img  
sudo e2fsck -f /dev/loop0p2  
sudo resize2fs /dev/loop0p2
```

Finally, unmount and remove the loopback device:

```
sudo umount /dev/loop0p*  
sudo losetup -d /tmp/image.img
```

Verifying the Image

You can verify the created image file using tools like ``fdisk`` or ``gparted``. The image should contain two partitions: a FAT32 EFI partition and an EXT4 MX23 partition with the specified sizes.

1

``fallocate``: A command that creates or extends a file to a specified size.

 Rewrite



Related

Copilot

How can I verify the created image file using gparted?



Home



Discover



Library



Search images



Search videos



Copilot



Home



Discover



Library