Installing Slackware on UEFI-based hardware

Slackware setup supports UEFI installs. To get Slackware to boot on UEFI machines, bypass the LILO installation and select ELILO installation when prompted during the install.

Installing Older Slackware on UEFI-based hardware

Some modern computers have started to offer motherboards that use Unified Extensible Firmware Interface (UEFI) as a replacement for the traditional BIOS. Slackware is perfectly capable or running on such machines, albeit with the provisos that before Slackware 14.1 the official install media could not be booted via UEFI directly and no UEFI bootloader was provided as part of the official selection of packages.

If you do wish to install Slackware on a UEFI based machine it is **strongly** recommended that you use Slackware64 14.1 or above, thus allowing you to simply boot the official install media.

If you have a requirement for Slackware 14.0, you will need to follow one of our approaches:

- Enabling legacy BIOS support
- Creating your own install media and booting from this
- Using another distro's install media to bootstrap the Slackware installation (this includes using modern Slackware 14.1 or above media to install Slackware 14.0)

(**Note:** These last two methods are *unsupported*!

Legacy BIOS support

The majority of new UEFI-based machines have options to simulate a traditional BIOS. This is often referred to Legacy BIOS support. If your machine supports this it should be your first choice, as once enabled you can continue to install Slackware as you have always done, using the official install media and setup instructions. How you enable this mode varies from manufacturer to manufacturer. Refer to your vendor's documentation on how to enable this mode.

Creating your own install media for older Slackware releases

The easiest way to make your own Slackware UEFI-capable install media is to create a USB boot disk. To do so you may need to use a second machine. Alternatively you can use the target machine directly if it has another OS installed that provides a GPT (GUID Partition Table) capable partitioning tool or you have UEFI-capable Live media (CD, DVD, USB, etc.) provided by another Linux distro, which includes such a utility. Suitable partitioning tools include GPT fdisk (a.k.a gdisk) and GNU

parted, both of which are provided by recent versions of Slackware.

The USB disk you are going to use as install media will need to be partitioned with a GPT scheme. It will need to have a UEFI System Partition as a bare minimum. This system partition is used by UEFI to hold a UEFI-capable bootloader(s) and configuration file(s), along with one or more kernels. The system partition must use a FAT32 filesystem (or it can be HFS+ if your machine is using Apple-based hardware) and must be big enough to hold the Slackware install media's huge kernel, initrd and a boot loader. It is therefore recommended that you make the partition at least 50Mb in size (slightly less could be used but 50Mb gives you a little extra free space to work with). Of course more space will allow for a greater range of options (e.g. including additional kernels or alternative bootloaders). The system partition must be marked as such via the partitioning tool. If you use GPT fdisk this is done by using the partition type code EF00. If you are using Slackware to create this partition, formatting can be done via the mkdosfs utility.

Once the system partition has been created add a ./EFI/BOOT/ directory structure to this partition, into which you place an EFI-capable bootloader, which should be named BOOTX64.EFI if your PC is x86_64 capable. This exact naming (./EFI/BOOT/BOOTX64.EFI) is not an absolute requirement. You can use a different directory structure and name for your bootloader, however conforming to this suggestion will simplify things for you as UEFI systems consider a bootloader named in this way to be the default.

Whilst you can use any EFI-capable bootloader, it is recommend that you use Elilo. Elilo is very simple to use and configure, it is reliable and works in a similar manor to Lilo. As such it should feel familiar to most Slackware users, given that Lilo is Slackware's default bootloader. Elilo does not need to be compiled as the binaries provided by the upstream project should work directly. For example, if you plan to install Slackware64, download elilo-3.14-all.tar.gz, extract elilo-3.14-x86_64.efi and place it on your system partition as ./EFI/BOOT/BOOTX64.EFI.

Into the same directory on your system partition (./EFI/BOOT) copy over the following two files from the official install media: kernels/huge.s/bzImage and isolinux/initrd.img (just the files, not their directory structures). Plus create an elilo.conf, that looks like this:

```
prompt
timeout=50
default=install

image=bzImage
  label=install
  initrd=initrd.img
  append="load_ramdisk=1 prompt_ramdisk=0 rw printk.time=0
SLACK_KERNEL=huge.s"
```

You should now have a USB disk that you can boot from, though you may need to configure your UEFI to consider this the first boot disk (refer to your manufacturer's documentation on how to do this) or use the EFI Shell to specify this disk and/or bootloader.

In addition to the system partition you may wish to have a second partition that holds the rest of the Slackware installation files. This can be formatted with any filesystem supported by the the huge.s kernel. This partition can be mounted after you have booted from your install media and used as a source of packages during installation. Alternatively you can insert an official Slackware CD or DVD

once setup is up and running, or perform a network installation.

Bootstrap the Slackware installation environment via another distro's install media

Making your own boot media is almost always preferable but if this does not suit you for some reason, you can often use the UEFI-capable install media from another distro. The steps are as follows:

- Boot the other distro's install media
- Switch to a virtual terminal
- Bring up the network if a connection is not already configured
- Use wget or curl to fetch the initrd.img from the isolinux subdirectory of the Slackware version you wish to install, using a nearby mirror
- Unpack initrd.img (using zcat and cpio) into a newly created directory (e.g. /Slackware)
- Chroot into that directory (after first using mount to bind /dev, /sys and /proc from the host system and copying over /etc/resolv.conf)
- Setup the $PATH and other variables needed to run the installer (by sourcing the chooted Slackware profile, i.e. ". /etc/profile")

Note: If you use Slackware64 14.1 install media, you could opt to do a network install and select the mirror for an earlier release. Be aware however that this remains *unsupported*!

Tips on install when not using BIOS legacy mode

Once you have the Slackware installation environment up and running, remember to use a GUID Partition Table rather than MBR Partitioning and include a suitable system partition on the primary local hard disk. During the setup phase you should skip Lilo configuration, since you will not be using it. Instead configure your EFI-capable bootloader after setup has finished and before your first reboot. To do this mount the local system partition and setup your boot environment by adding an EFI-capable bootloader, kernel and any configuration files. If you used the above instructions for creating your own install media, simply copy over the files from your boot media to your local system partition, then adjust the local elilo.conf as follows (this assumes you placed Slackware's root partition on /dev/sda2):

```
prompt
timeout=50
default=huge

image=bzImage
  label=huge
  read-only
  root=/dev/sda2
```

After umounting and removing the install media you should now be able to reboot into your new

Slackware install (you may need to change the boot order in your UEFI again if you adjusted it previously).

Finally, after booting into Slackware remember to configure an initrd for the generic kernel and copy both over to the local system partition. A further adjustment to the elilo.conf to account for this could look something like the following (assuming a 3.2.29 kernel):

```
prompt
timeout=50
default=generic

image=vmlinuz-generic-3.2.29
  label=generic
  initrd=initrd.gz
  read-only
  root=/dev/sda2

image=bzImage
  label=huge
  read-only
  root=/dev/sda2
```

Sources

- Originally written by ruario
- Further UEFI Linux booting options: http://www.rodsbooks.com/efi-bootloaders/

howtos, uefi, efi, boot, install, hardware, author ruario

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:slackware_admin:installing_on_uefi_hardware

Last update: **2018/10/29 19:46 (UTC)**