Slackware AArch64: Installer

Using the Installer, you can install and configure Slackware how it was designed to be: from the installed package set to the file system and its layout.

Installer uses

The Installer can also be used as:

- a rescue environment
- a development environment to onboard new Hardware Models.
- installing Slackware!

Slackware Installer packages

There are three types of Slackware Installer packaged for the AArch64 architecture. They are described below in the order in which they are created by the Slackware ARM build system.

Naked Installer: Generic Initial RAM Disk

The AiO and Bare Installer packages target specific Hardware Models and enable booting from the SD card directly into the Slackware Installer.

The Naked Installer is not packaged for any specific Hardware Model. The other Installer packages (Bare & AiO) contain this Naked image unmodified within a file system on within the disk image.

It's usually used to develop support for (presently) unsupported Hardware Models. *Installation within QEMU uses this Installer package.*

This Installer image can also be used to boot a Hardware Model over TFTP.

The Naked Installer initrd image is used (unmodified) within the Bare Installers.

Created by	Refresh/update cadence
Slackware ARM Kernel build process	Coupled to Kernel package upgrade (typically fortnightly or more frequently for -current)

Bare Installers for Supported Hardware Models

Bare Installers are identical to the AiO Installers, except that they don't contain the package set. The user must supply the package set on some other media (e.g. USB stick, NFS share, over HTTP/web).

The Bare Installers are approximately 250MB (compressed) in size. Each image is packaged for a particular Hardware Model.

These Installers package the *Naked Installer* in a format bootable on a specific set of supported Hardware Models.

Users may choose to use the Bare Installers when:

- They want the root file system to be on the SD card (this is more tricky when using the AiO images)
- The AiO Installer has a bug/lacks a certain feature, and you'd like to use the more recent Installer before the AiO is refreshed.
- The user already has the Slackware ARM repository locally and does not want to download the larger AiO image.

Created by	Refresh/update cadence
Installer Image building system	Coupled to Kernel package upgrade (typically fortnightly or more frequently for -current)

All in One (AiO) Installer for Supported Hardware Models

The All in One Installers are modified *Bare Installer* images, including an additional partition containing the Slackware packages/installation media.

The Slackware package set are included within the downloadable image. This enables offline installation and the easiest installation option. This mimics a stable release of Slackware - a frozen set of packages are installed, and users upgrade (and continue to update) the OS using `slackpkg` post installation.

The AiO (All in One) Installers are the prescribed installation package, as laid out in the Installation guides.

The AiO images are approximately 5GB (compressed) in size. Each image is packaged for a particular Hardware Model.

All users are encouraged to use the AiO Installers for supported Hardware Models.

Created by	Refresh/update cadence
All in One build system (not presently available)	Typically once a month

Installation Interfaces

You can perform the Slackware installation over three interfaces.

- Local keyboard & monitor: This is the documented method.
- Over a Serial connection: This is useful for remote installations where a serial cable is permanently connected to a remote machine. The end-to-end installation (including accessing the Hardware Model's Boot Loader) can be controlled.
- Over the network, using SSH: This is useful for headless installations where the IP address provided to machine (DHCP is enabled by default) is known.

> If logging in over SSH, the root password is empty by default. To set an Installer root password, you can mount the Bare or AiO Installer image and edit

💡 /boot/extlinux/extlinux.conf. Use the Kernel cmdline operator `instrootpw` parameter (see inline documentation within the extlinux.conf file)

Slackware Installer: Post Installation scripts

The Slackware ARM installer supports an additional set of post installation scripts which help with the the various setup and configuration options.

Slackware ARM post installation script execution flow

Primary Launcher

```
# Script: /usr/lib/setup/armedslack-run-postinstall-scripts
# Sourced from: /usr/lib/setup/SeTconfig
# Runs immediately prior to running the regular non-ARM scripts
```

Runs everything within: /usr/lib/setup/armedslack-postinstall-scripts

Example:

```
$ ls /usr/lib/setup/armedslack-postinstall-scripts/
000-env-postinst-setup  001-hwm-os-configure*  os-initrd-mgr*
removeinstaller*  setconsole*
```

They will be executed in order.

Examining some post install scripts

Post install scripts: Hardware Model Helper scripts

Hardware Model configuration helper script: /usr/lib/setup/armedslack-postinstall-scripts/001-hwm-os-configure

This script executes everything found within: /usr/share/hwm-configure/platform/aarch64/installer/helper.scr/

```
for platformscr in hwm-configure/platform/aarch64/installer/helper.scr/*; do
echo $platformscr;done
hwm-configure/platform/aarch64/installer/helper.scr/000-grub-configure
hwm-configure/platform/aarch64/installer/helper.scr/rk3399
hwm-configure/platform/aarch64/installer/helper.scr/rpi
```

```
# Script......: /usr/share/hwm-
```

```
configure/platform/aarch64/installer/helper.scr/rpi
# Called from : /usr/lib/setup/armedslack-hwm-os-configure
```

Fixes up the extlinux.conf for RPis to enable FDT within the RPi native BL. Dependency: hwm-configure/platform/aarch64/installer/helper.scr/u-boot-configure As this creates the u-boot extlinux.conf config.

This will run in the correct order due to the file naming scheme.

Post Install scripts: Remove Installer


```
/usr/lib/setup/armedslack-postinstall-scripts/removeinstaller
# Called from: /usr/lib/setup/armedslack-run-postinstall-scripts
```

** Notes: this needs to run AFTER the u-boot-configure script due to it removing the OS boot stanza template.

It will run afterwards due to the ordering provided by the file naming scheme within /usr/lib/setup/armedslack-run-postinstall-scripts