

SDT-DRIVERKIT(1) SUSE Driver Tools

Scott Bahling

23-07-2013

NAME

sdt driverkit - tool for creating driver kits

SYNOPSIS

sdt driverkit [*options*]

DESCRIPTION

Driver Kits

A Driver Kit is an add-on product that delivers kernel drivers for a specific target distribution and is the recommended way to deliver drivers to end users. Being a standard SUSE add-on product, the driver kit can be delivered on CD/DVD or as a package repository via network or local file system.

Creating driver kits

To create a driver kit use the `driverkit` sub-command of the `sdt` tool.

```
sdt driverkit -u updates
```

The `-u/--updatedir` option specifies a directory where the `sdt` tool will find the files to be added to the Update Medium.

Creating bootable driver kits

To create a bootable driver kit use the `-b\--base` option to point to the distro installation media used as a basis of the bootable driver kit.

```
sdt driverkit -u updates -b ./distro/sles11sp1-x86_64
```

This will copy the boot structure from the distro installation media to the newly created driver kit. In addition, any KMPs, kernel packages, or other files found in the directory specified by `-u\--updatedir` option will be added to the boot structure appropriately.

Creating a *merged* media

To create an ISO image that *merges* the driver-kit with the base SUSE product onto a single media, use the `-M/--merge` option:

```
sdt driverkit -M -u updates -b ./distro/sles11sp1-x86_64
```

This will generate the driver-kit as usual and copy it along with the base SUSE Product repository to the new ISO image. The ISO boot structure will be updated to include any updated kernel, kernel modules or other updates just as with the bootable driver kit.

This option can also be used to merge an existing driver-kit (or any add-on product) with a SUSE product installation media as follows:

```
sdt driverkit -M -a addon -b ./distro/sles11sp1-x86_64
```

where `addon` points to the path of the add-on repository to be merged with the SUSE Linux Enterprise installation media.

OPTIONS

- u** *UPDATEDIR*, `--updatedir=UPDATEDIR` Directory to be scanned for update files to be added to driver kit. Directory will be scanned for kernel packages, KMP packages, YaST2 modules (.ybc), and tar files (.tgz).
- w** *WORKDIR*, `--workdir WORKDIR` Work directory. Created if non-existent and remains after completion. If this option is not specified, temporary directory is created and deleted on exit.

-m MODSPEC, --modspec MODSPEC Additional module to add to module.config such that it's added to initrd and linuxrc considers it while probing hardware. MODSPEC is of the following format:

SECTION:MODULE,DESCR,PARAM,PRE_INST,POST_INST,INITRD,AUTO

If the kernel doesn't contain any new driver which should be loaded for installation compared to the kernel contained in the base ISO, this option isn't necessary.

-a ADD-ON_DIR, --addon ADD-ON_DIR Path to add-on product to be used as basis of a driverkit. Path will be scanned same as update directory for kernel or other package updates. When used together with the **-b|--base** option, the add-on product will be copied without modification together with the boot structure from the base media thereby generating a bootable driver kit.

-s {u,p,f,up,pf,upf}, --steps {u,p,f,up,pf,upf} Specifies steps to perform. STEPS is any combination of "u", "p" and "f". "u" stands for Unpack, "p" for Prepare and "f" for Finish. The default is to perform all steps, that is, "-s upf".

Unpack step copies necessary directories and files from base ISO under workdir/boot.staging, unpacks initrd from it under workdir/initrd.staging and kernel package under kernel.staging. module.config is copied to module.config under workdir and --module options are applied.

Prepare step installs the new kernel, modules and kernel package under appropriate staging directories.

Finish step packs the initrd and then produces the ISO.

This option allows manual customization or fix up of media. Note that --workdir must be specified when this option is used.

--no-uni-dud By default DUDs are generated to work with both SUSE Linux Enterprise Server and Desktop products. Use this option to disable this and have the DUD structure created to only work with the specified base product.

-o OUTPUT, --output OUTPUT Output file to be created. Defaults to out.iso.

-V VOLUME, --volume VOLUME ISO9660 Volume Identification for the ISO image.

-p PREPARER, --preparer PREPARER ISO9660 Preparer Identification for the ISO image.

-P PUBLISHER, --publisher PUBLISHER ISO9660 Publisher Identification for ISO image.

-A APPLICATION, --application APPLICATION ISO9660 Application spec for the ISO image.

OPTIONS FOR BOOTABLE DRIVER KIT

-b *BASE_MEDIA*, **-base *BASE_MEDIA*** Points to the contents of the initial installation media of the SUSE Linux Enterprise Product that the driver kit is designed to work with. Option can point to either a directory containing the contents of the installation media, or the media ISO image itself.

This option will cause the driver kit to be built as a bootable driver kit.

-M, **-merge** Merge driver kit with base SUSE Linux Enterprise installation media. A new ISO image is created containing the standard SUSE Linux Enterprise installation repository in addition to the driver kit repository along with the boot loader and installer. The installation process will automatically add the driver kit as an add-on product to be installed in tandem with the base SUSE Linux Enterprise product.

-boot-param *BOOTPARAMS* Specify a parameter to add to the isolinux.cfg/elilo.conf file. Format is “label:param” where label is the label of the section in the config file to add the param to.

Example: Add “install=” param to the section labeled “linux”:

```
--boot-param="linux:install=http://installserver/repo"
```

-no-initrd-mod-updates Do not update the kernel modules in the initrd. When updating the initrd, the kernel modules and related files will be copied as-is from the base media to the bootable driverkit initrd.

-E, **-no-efi** Do **not** generate efi boot loader. Only applies to x86_64 arch.

-e, **-embed-dk** Embed driver kit repository into initrd. This can make the initial ram disk quite large, and cause long ram disk load times.

-D, **-no-dk** Exclude driver kit repository on base of media. Driver Kit will be embedded in initrd.

FILES DETECTED IN UPDATEDIR

When scanning the *UPDATEDIR* for files to be added to the driver kit, certain files will be automatically detected and specific actions taken. See sdt-file-detection (7) for more information on how files are detected.

SEE ALSO

sdt(1), sdt-file-detection(7)