

SDT-PXE(1) SUSE Driver Tools

Scott Bahling

19-05-2013

NAME

sdt pxe - create `linux` kernel and `initrd` images for PXE installer

SYNOPSIS

sdt pxe [*options*]

DESCRIPTION

PXE

The `pxe` command creates an updated `initrd` image based on the `initrd` file contained on the media specified by the `-b/--base` option with updates applied from the files contained in the directory specified by the `-u/--updates` option.

In addition, a Linux kernel image will be extracted from either the base media or a kernel package if provided as an update.

The `initrd` image will be placed the specified output directory as the file `initrd`. The Linux kernel image will be placed in the same directory as the file `linux`.

These files can be deployed on a PXE boot server for booting systems using the updated kernel or kernel modules.

Creating PXE boot files

To create the PXE boot files use the `pxe` sub-command of the `sdt` tool.

```
sdt -O PXE-files pxe -u updates -b ./distro/sles11sp1-x86_64
```

The `-O/--outputdir` option of the `sdt` command specified the location where the PXE files should be placed.

The `-b/--base` option points to the distro installation media used as a basis of the bootable driver kit.

The `-u/--updatedir` option specifies a directory where the `sdt` tool will find the files to be added to the Update Medium. Any KMPs, kernel packages, or other files found in the directory will be added to the `initrd` appropriately, and the `'linux'` file exported will be based on any *default* flavor kernel package provided. If no updated kernel package is provided, the `linux` kernel image will be exported from the base media.

OPTIONS

-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.

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

--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`.

-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.

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)