How to Set Boot Parameters for the Different Linux Distributions

Planning / Implementation

There are lots of components during the Linux boot/installation process and each component has its own parameters. The following figure shows the high-level stages of a typical Linux boot process.

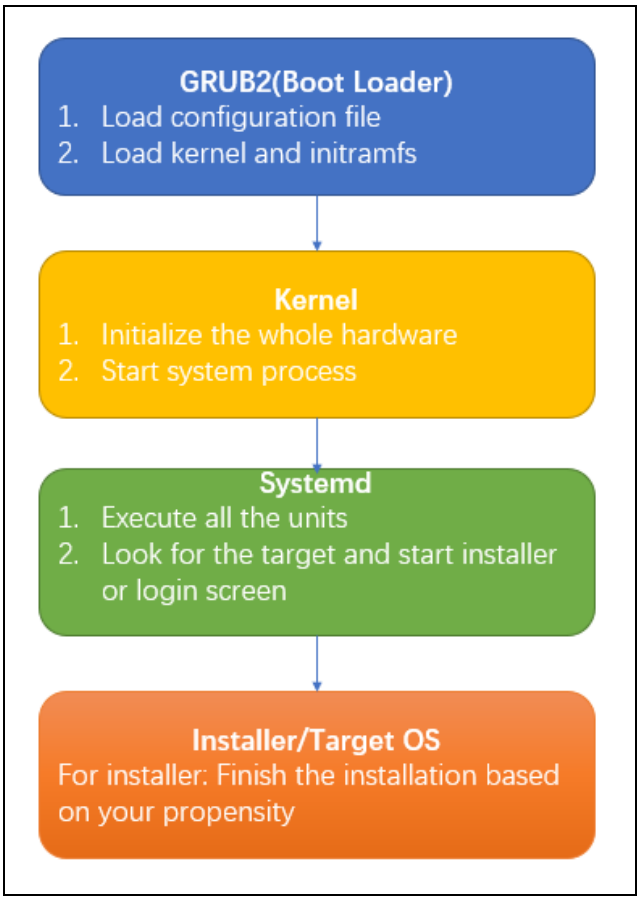


Figure 1. High-level stages of a typical Linux boot process

For the different Linux distributions, some parameters of above components are identical. However, different parameters among those distributions may have the same functionality. To avoid incorrect use of parameters, this chapter describes how to catch and set the relevant parameters about the different units.

The use of GRUB2

GRand Unified Bootloader version 2 (GRUB2) is a bootloader that allows user to quickly, easily and painlessly control the boot sequence of their systems. It is easy to get the usage with the following command.

```
Ubuntu OSes:        # info grub
RHEL and SUSE OSes: # info grub2
```

Alternatively, you can view the user manual on the GNU web site:
http:// https://www.gnu.org/software/grub/manual/grub/

The parameters of the installer

In this section, we introduce the parameters used in the different Linux installers.

- Anaconda
- Linuxrc
- Ubuntu installer

Anaconda

Anaconda is an installation program used by Red Hat Enterprise Linux, Fedora and some other distributions such as Asianux, Qubes OS, etc.

As different OS version contains different version of installer, download the relevant anaconda source package for the correct parameter. If you do not get the source package, refer to the close tag of https://github.com/rhinstaller/anaconda to find the parameter. Below is an example on RHEL 9.

1. Download the source package from https://access.redhat.com/.
2. Extract the file “boot-options.rst” from source package.

```
[root@rhel9 anaconda]# rpm2cpio anaconda-34.25.1.14-1.el9.src.rpm |cpio -div
anaconda-34.25.1.14.tar.bz2
anaconda.spec
7021 blocks
[root@rhel9 anaconda]# tar -jxf anaconda-34.25.1.14.tar.bz2
[root@rhel9 anaconda]#
[root@rhel9 anaconda]# find . -name boot-options.rst
./anaconda-34.25.1.14/docs/boot-options.rst
[root@rhel9 anaconda]#
```

Figure 2. Extract the file “boot-options.rst” from source package

3. Check the file boot-options.rst to find out the required parameter and the corresponding definition. The parameter “ip” is an example different from SUSE installer.

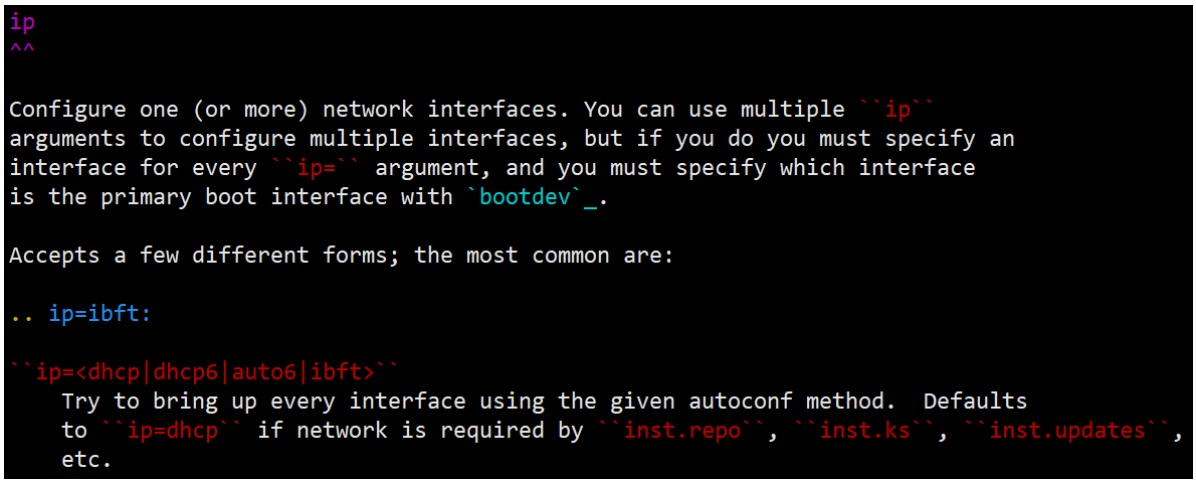


Figure 3. ip parameter

Based on the guide of anaconda, the bootup is handled by dracut, so we can assume that the dracut options are also part of the anaconda parameters.

Those options can be shown with the command "man dracut.cmdline" or the following website: https://mirrors.edge.kernel.org/pub/linux/utils/boot/dracut/dracut.html#dracutcmdline7

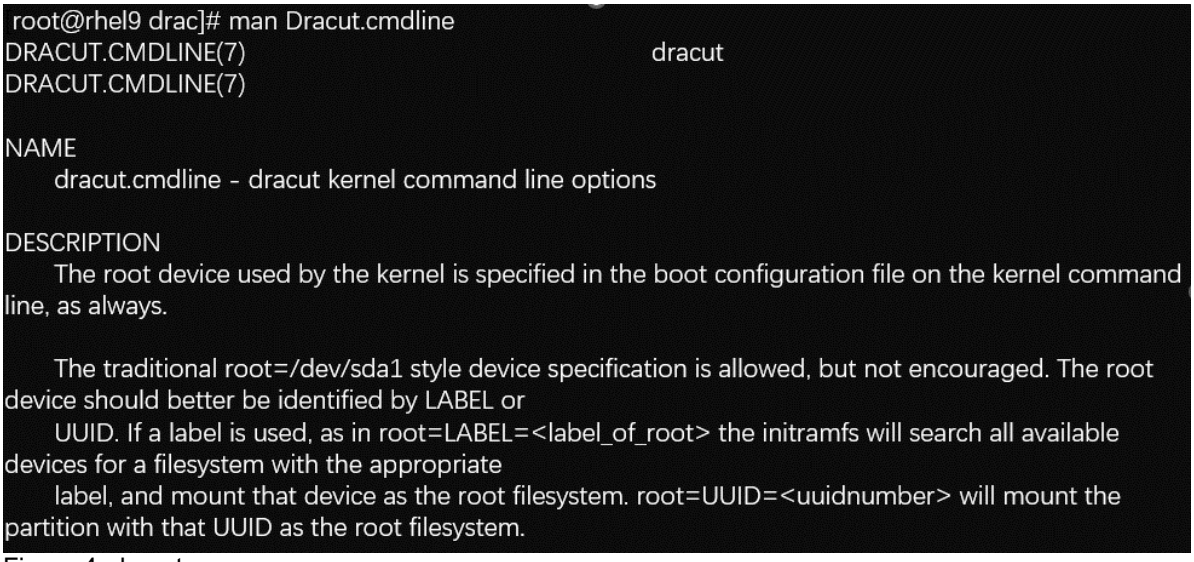
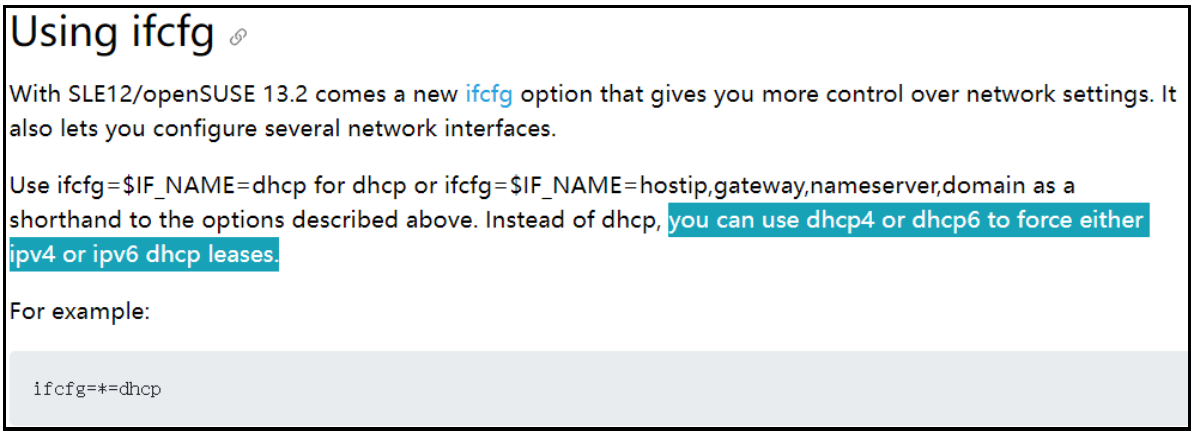


Figure 4. dracut man page

Linuxrc

Linuxrc is a small program that runs before the actual installation program YaST starts. The supporting parameter is included in the file linuxrc.html from the package “linuxrc”. You can also refer to web page **en.opensuse.org/SDB:Linuxrc** directly, however it is better to choose the version close to the old OS in the “History” item. The following figure is the example different from RHEL ip parameter.



Using ifcfg

With SLE12/openSUSE 13.2 comes a new ifcfg option that gives you more control over network settings. It also lets you configure several network interfaces.

Use ifcfg=$IF_NAME=dhcp for dhcp or ifcfg=$IF_NAME=hostip,gateway,nameserver,domain as a shorthand to the options described above. Instead of dhcp, you can use dhcp4 or dhcp6 to force either ipv4 or ipv6 dhcp leases.

For example:

```
ifcfg=*=dhcp
```

Figure 5. ifcfg parameter

Ubuntu installer

The default installer of Ubuntu 18.04 or earlier is Debian-installer(d-i). However, Ubuntu Server 20.04 or later introduces a new installer ‘subiquity server installer’ instead of the Debian-installer. As the d-i is about to be abandoned, this section mainly focuses on the 'subiquity'.

Subiquity contains three parts:

- Casper
- Curtin
- Cloud-init

The main boot options reside in Casper.

Refer to https://manpages.ubuntu.com/ for detailed information.

Kernel parameters

For kernel parameters, most of them are still consistent, such as nomodeset, iommu, acpi,etc. With the development of new features, new parameters have to be imported, such as no5lvl and hugetlb_free_vmemmap.

In order to check the supported boot parameter of your up-and-running OS, you can install kernel-doc package and get the available parameter list in the file kernel-parameters.txt (RHEL OS).

```
[root@localhost ~]# yum install -y -q kernel-doc

Installed:
  kernel-doc-5.14.0-162.6.1.el9_1.noarch

[root@localhost ~]#
[root@localhost ~]# find /usr/ -name "kernel-parameters.txt"
/usr/share/doc/kernel-doc-5.14.0-162.6.1.el9_1/Documentation/admin-guide/kernel-parameters.txt
[root@localhost ~]#
```

Figure 6. Installing kernel-doc

For SUSE OSes, please refer to https://github.com/openSUSE/kernel/blob/master/Documentation/admin-guide/kernel-parameters.txt, as the link is master branch by default, you need to switch to the relevant branch about OS version shown as follows.

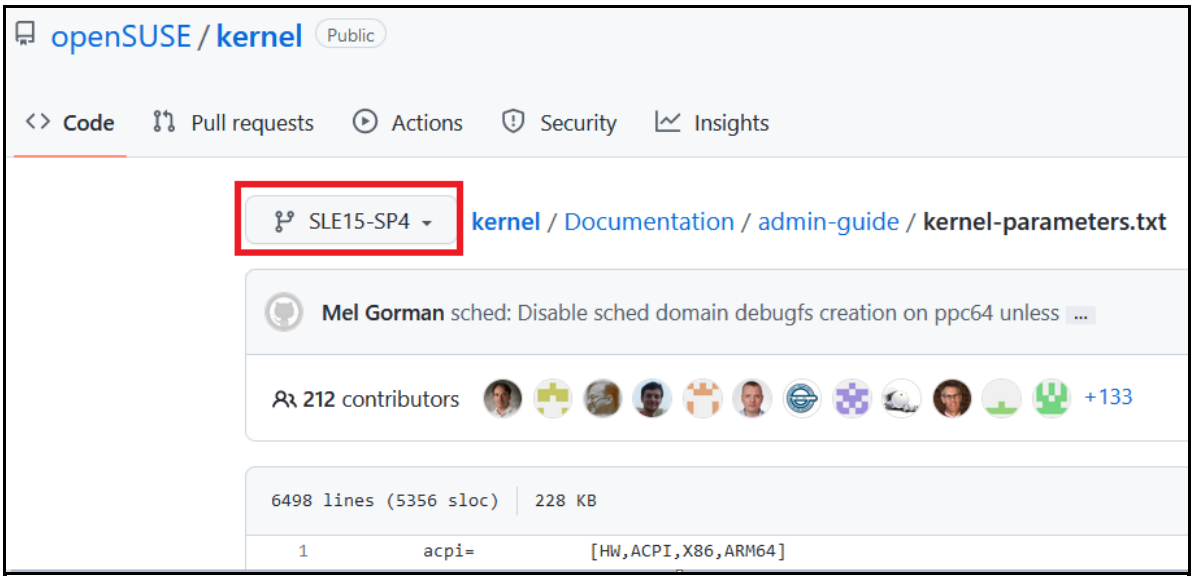


Figure 7. Kernel parameters information in openSUSE

For Ubuntu OSes, you can install linux-doc package and get the available parameter list in the file kernel-parameters.txt.gz.

```
root@ubuntu:~# apt install -y linux-doc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  linux-doc
0 upgraded, 1 newly installed, 0 to remove and 152 not upgraded.
Need to get 44.9 MB of archives.
After this operation, 152 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 linux-doc all 5.4.0-137.154 [44.9 MB]
Fetched 44.9 MB in 1min 40s (448 kB/s)
Selecting previously unselected package linux-doc.
(Reading database ... 109183 files and directories currently installed.)
Preparing to unpack .../linux-doc_5.4.0-137.154_all.deb ...
Unpacking linux-doc (5.4.0-137.154) ...
Setting up linux-doc (5.4.0-137.154) ...
root@ubuntu:~#
root@ubuntu:~# find /usr/ -name kernel-parameters.txt.gz
/usr/share/doc/linux-doc/admin-guide/kernel-parameters.txt.gz
root@ubuntu:~#
```

Figure 8. Kernel parameters information in Ubuntu

Systemd parameters

Systemd is Linux's system and service manager. It is the parent process that starts as PID 1 and acts as an init system which starts and maintains user space services. There are some boot parameters for controlling system behavior.

Since it is opensource software, we can check the boot parameters in the https://github.com/systemd/systemd/blob/main/man/kernel-command-line.xml instead of decompressing the source package.

Applying the boot parameters

This section describes how to apply the parameters during installation and boot.

For the installer, the kernel and system apply parameters in the same way, but the server can be set up as a BIOS-based or UEFI-based system, thus you will get a different boot menu interface. Below is setup method on BIOS-based systems.

RHEL

Press the **Tab** key and add custom boot options to the command line, press “Enter” to continue booting, as shown in the following image.

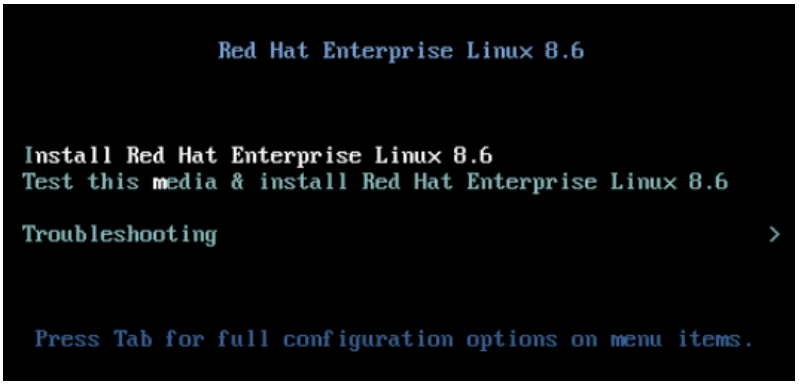


Figure 9. Prompt to press the tab key in RHEL

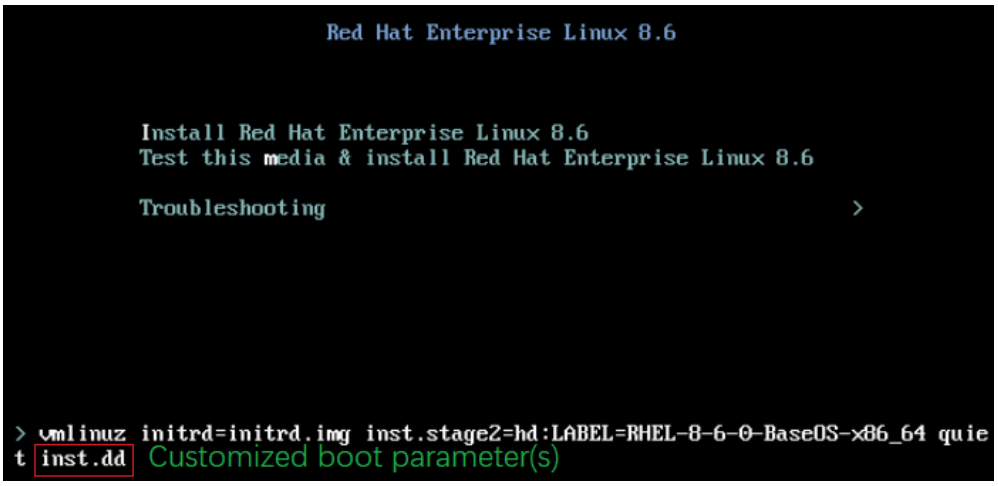


Figure 10. Entering custom boot options in RHEL

SUSE

Add custom boot options to the line “Boot Options” and press the **Enter** key to start the installation.

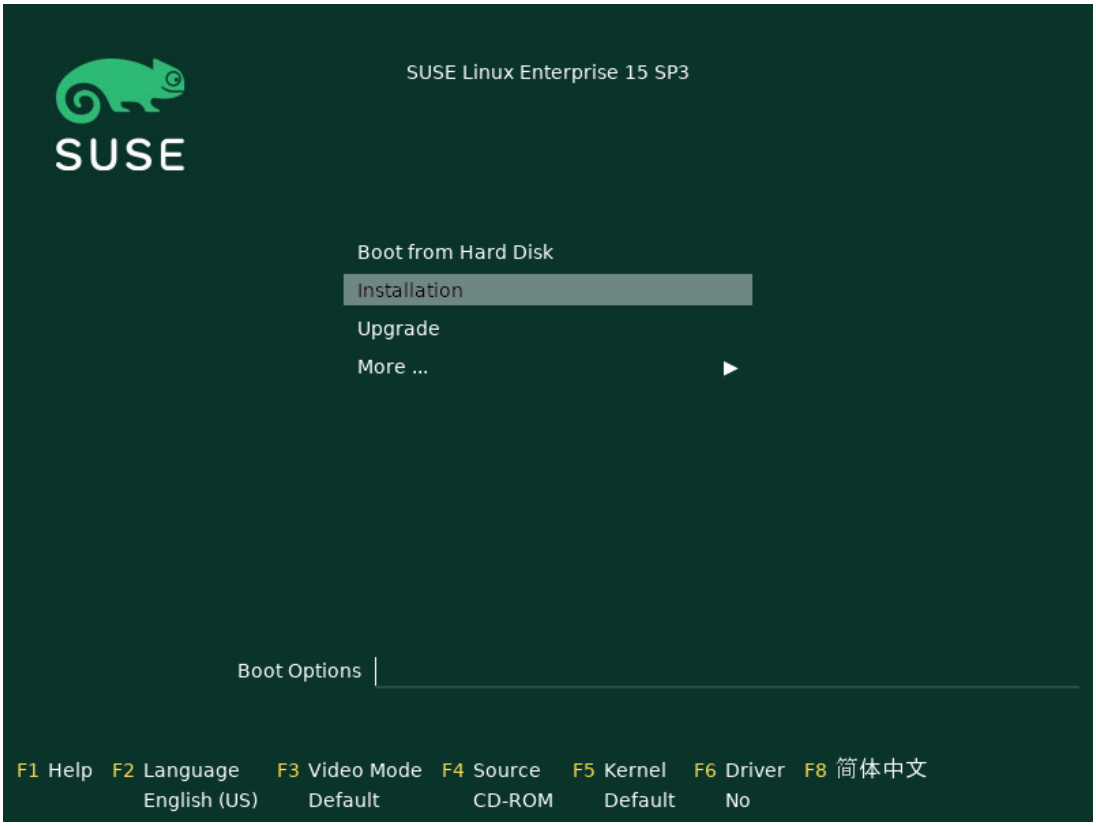


Figure 11. Custom boot options in SUSE

Ubuntu 18.04 & 20.04

For 22.04 refer to the next figure for UEFI-based systems below.

The line “Boot Options” is not displayed by default, it appears when the F6 key is pressed, adding custom boot options between “Boot Options” and text “---”.

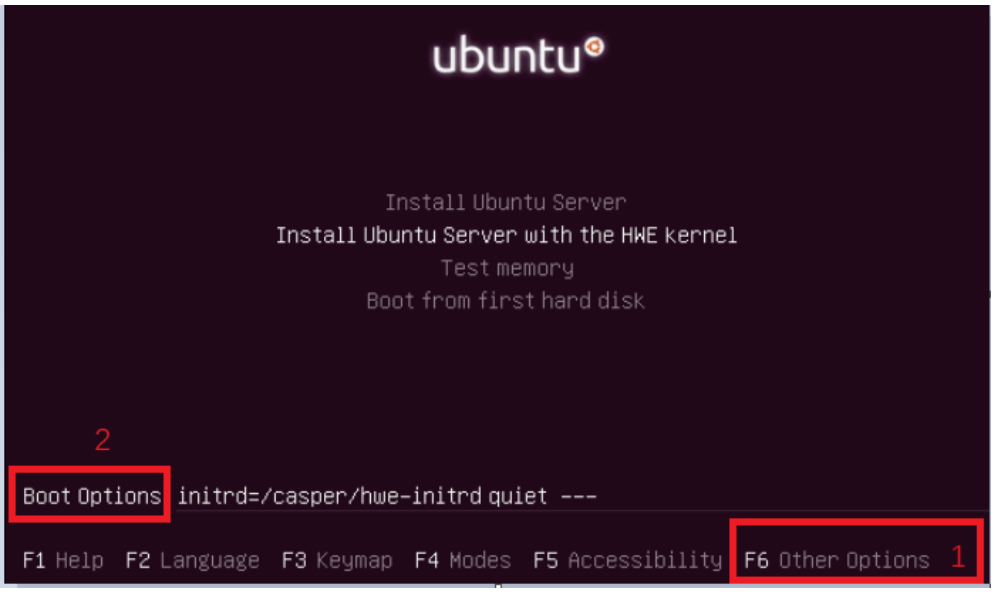


Figure 12. Custom boot options in Ubuntu

UEFI-based systems

Below is setup method on UEFI-based systems. It works in the same way for RHEL/SUSE/Ubuntu server (18.04,20.04,22.04[Both UEFI and BIOS]) OSes, with SLES 15.3 used below as an example.

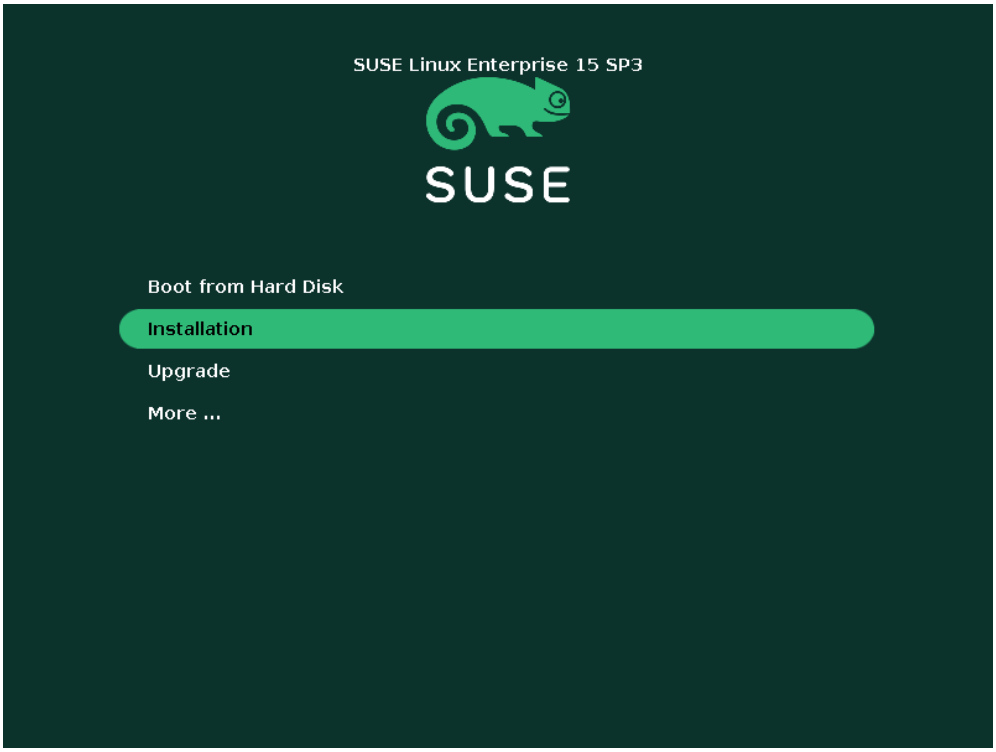


Figure 13. SLES boot menu

Use the arrow keys to highlight the **Installation** entry and press “e” to add parameters to the end of the line that starts with `linuxefi` or `linux`, as shown below. To boot the edited entry, press F10 or other combination key based on the help at the bottom.

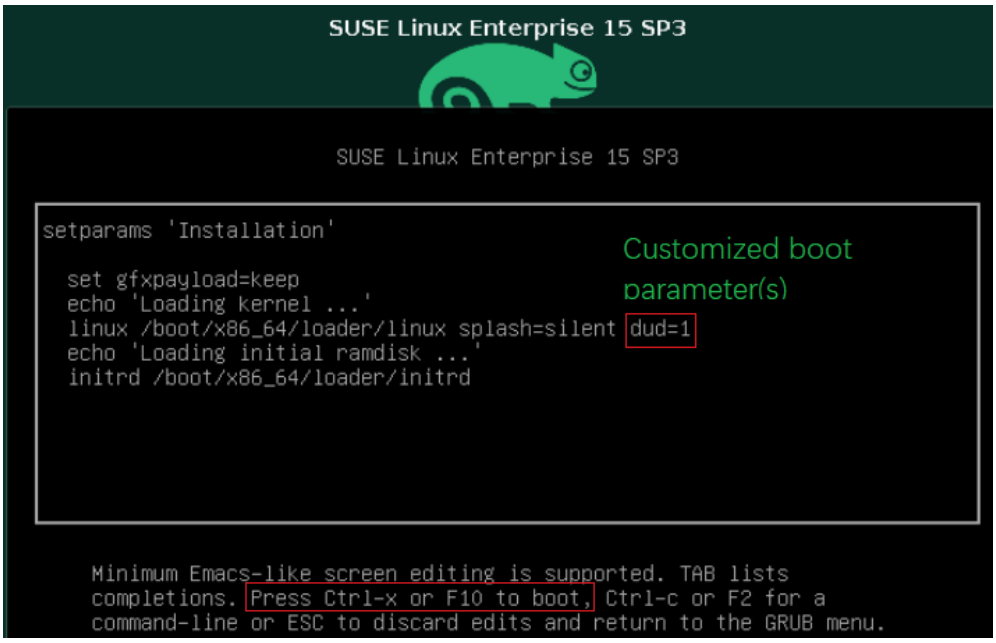


Figure 14. Adding parameters

Common parameters

This section introduces some common but ambiguous parameters and debug settings of GRUB2.

The following is the list of common parameters about different distributions.

Driver update parameters:

- **RHEL**:
  `inst.dd=hd:LABEL=DISK_LABEL:/DRIVER.rpm`
  `inst.dd=http://IP/PATH/DRIVER.rpm`
- **SLES**:
  `dud=disk:/DRIVER.dud`
  `dud=http://IP/PATH/DRIVER.dud insecure=1`
- **Ubuntu**:
  `debian-installer/driver-update=`

DHCP parameters:

- **RHEL**:
  `ip=dhcp|auto6`
- **SLES**:
  `ifcfg=*=[dhcp|dhcp6]`
- **Ubuntu**:
  `ip=dhcp`

Private parameters:

- **RHEL**:
  `inst.addrepo=REPO_NAME,[http,https,ftp,nfs]:// /`
- **SLES**:
  1. `self_update=[0|1]`
  2. `addon=https://example.com/addon1,ftp://user:password@example.com/addon2`
- **Ubuntu**:
  `toram`
  `todisk=DEVICE`

Public parameters (all distributions):

1. Debug boot issue: `console=tty0 console=[ttyS0|ttyS1],[BAUD_RATE]n8` (where BAUD_RATE depends on UEFI setting)
2. Debug display issue: `nomodeset`

The following are the GRUB2 debug settings steps:

1. Enable the serial debug in the files “/etc/grub.d/00_header”and “/etc/default/grub”.

```
# cat /etc/grub.d/00_header  |grep -E "pager|debug"
set pager=0
set debug=all
#
```

Figure 15. Enable the serial debug in 00_header

```
# cat /etc/default/grub |grep -Ew "GRUB_TERMINAL|GRUB_SERIAL_COMMAND"
GRUB_TERMINAL="serial"
GRUB_SERIAL_COMMAND="serial --speed=115200 --unit=0 --word=8 --parity=no --stop=1"
#
```

Figure 16. Enable the serial debug in grub

2. Rebuild the grub.cfg file using one of the following commands:
    - RHEL 7 & RHEL 8 with UEFI:
      ```
      # grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
      ```
    - RHEL 7 & RHEL 8 with Legacy BIOS
      ```
      # grub2-mkconfig -o /boot/grub2/grub.cfg
      ```
    - RHEL 9:
      ```
      # grub2-mkconfig -o /boot/grub2/grub.cfg
      ```
    - SLES:
      ```
      # grub2-mkconfig -o /boot/grub2/grub.cfg
      ```
    - Ubuntu:
      ```
      # update-grub
      ```

Author

Song Shang is a Linux Engineer in Lenovo Infrastructure Solutions Group, based in Beijing, China.

Related product families

Product families related to this document are the following:

- Red Hat Enterprise Linux
- SUSE Linux Enterprise Server

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service. Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
8001 Development Drive
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary. Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk. Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This document, LP1703, was created or updated on March 4, 2023.

Send us your comments in one of the following ways:

- Use the online Contact us review form found at:
  https://lenovopress.lenovo.com/LP1703
- Send your comments in an e-mail to:
  comments@lenovopress.com

This document is available online at https://lenovopress.lenovo.com/LP1703.

Trademarks

Lenovo and the Lenovo logo are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. A current list of Lenovo trademarks is available on the Web at https://www.lenovo.com/us/en/legal/copytrade/.

The following terms are trademarks of Lenovo in the United States, other countries, or both:
Lenovo®

The following terms are trademarks of other companies:

Linux® is the trademark of Linus Torvalds in the U.S. and other countries.

Other company, product, or service names may be trademarks or service marks of others.