European Organisation for Astronomical Research in the Southern Hemisphere

Programme: E-ELT

Project/WP: E-ELT Telescope Control

E-ELT Linux Installation Guide

Document Number: ESO-287339

Document Version: 5.6

Document Type: Procedure (PRO)

Released On: 2022-05-04

Document Classification: ESO Internal [Confidential for Non-ESO Staff]

Owner:	Guirao, Carlos
Validated by PA/QA:	Kurlandczyk, Hervé
Validated by WPM:	Kornweibel, Nick
Approved by PM:	Kornweibel, Nick
	Name

Authors

Name	Affiliation
C. Guirao	ESO
F. Pellegrin	ESO

Change Record from previous Version

Affected Section(s)	Changes / Reason / Remarks
4.6	Added ELT_ROLE usage and explanation
4.2	Added 4.2.1 to require BIOS settings for low latency
4.9	Describing an upgrade to an intermediate release
4.10	Describing network choices and REALTIME variant
4.11	Completing chapter 4.9
4.12	New chapter 4.11
4.13	Add warning against a “yum update”
4.14	Recommend a “reboot” after each update
5.1	Upgraded to CentOS-8x. Renaming eelt to elt
5.2	Recommendations from Paola Sivera
5.3	“yum makecache fast” is obsolete in CentOS-8.
5.4	New definition of Major.Minor.Patch numbers
5.5	Excluding obsolete CentOS-8 repositories in yum command
5.6	Appendix A. Changing disk partitioning

Contents

1 Introduction

1.1 Scope

This document describes the installation process of the ELT Linux Development Environment (DevEnv).

The main aim of this installation guide is to provide a simple but robust installation process that guarantees that, when completed, all systems are configured the same way: with the same packages, tools and services. The process finishes with a system ready for software development for the ELT project.

The document is addressed to system managers at ESO and at external contractors in charge of the installation and maintenance of systems supporting the ELT DevEnv. However the process is simple enough so that also developers with same basic knowledge in system administration can get the ELT DevEnv installed by themselves in their own systems: desktops, laptops or virtual machines.

This document applies to all system installations belonging to the ELT programme that will be developed by ESO or by external contractors.

1.2 Document structure

The document first describes some particularities of the installation process, like releases and the ESO repository. It continues later describing the HW requirements, the documentation available and precise instructions on how to prepare and complete the installation. Finally the document covers also the process to perform upgrades and indicates how to obtain ESO support in case of problems.

1.3 Definitions and Conventions

1.3.1 Abbreviations and Acronyms

The following abbreviations and acronyms are used in this document:

ESO	European Southern Observatory
CentOS	Community Enterprise Operating System
DevEnv	ELT Linux Development Environment
ELT	Extremely Large Telescope
OS	Operating System
RPM	RedHat Package Manager
TBC	To Be Confirmed

TBD	To Be Defined

1.3.2 Stylistics Conventions

Courier font is used to indicate text displayed by, or to be entered into the system. Bold font for entered text, selected text or clicks in window buttons. Italicized text in angled brackets indicates placeholders or descriptions of fields.

2 Related Documents

2.1 Applicable Documents

The following documents, of the exact version shown, form part of this document to the extent specified herein. In the event of conflict between the documents referenced herein and the content of this document, the content of this document shall be considered as superseding.

2.1.1 ESO Documents

None

2.1.2 Standards

None

2.2 Reference Documents

The following documents, of the exact version shown herein, are listed as background references only. They are not to be construed as a binding complement to the present document.

RD1 ELT Programming Language Coding Standard
ESO-254539 Version 2

3 ELT Development Environment

The ELT Linux Development Environment (DevEnv) comprises a collection of hardware, software procedures and tools for the developing, testing and debugging of software components for the ELT. It has to support largescale and long-term maintenance of software.

To guarantee a correct and proper integration of software components for the ELT, all participants in the development effort (software developers at ESO and external contractors) must conform to the same rules. Using the same version of the DevEnv is one of these rules.

As technology improves and requirements changes, the DevEnv functionality will tend to change. To accommodate these changes newer releases of the DevEnv will become available. Traceability of these changes is a must, therefore DevEnv releases will be subjected to strict configuration control.

The document describes a flexible, simple and robust installation process of the latest release of the DevEnv:

1. It is flexible because the OS can be installed on a large variety of HW: on bare-metal chassis or on virtual machines; on light system or on very powerful servers. There is only a minimum set of HW requirements which is granted almost by all systems.
2. It is reduced to a short sequence of commands to be executed immediately after the installation of the OS. It can be performed by System Administrators who may have to maintain it later; or by end users who may want to have their own copy of the DevEnv installed on light systems, like laptops or virtual machines.
3. It is robust because it is written on a single script, based on PUPPET, which guarantees the complete and correct installation of the DevEnv. This script takes care of the download and installation of the remaining software packages; configuration of services and the creation of user accounts. Another script can be used at any time to verify and certify that the system is complaint with the current release of the DevEnv.
4. Updates are trivial.

The DevEnv installation relies on a RPM repository maintained at ESO that guarantees that all software packages required during the installation process are available.

4 ELT Linux Installation Guide

The ELT Linux DevEnv is based on CentOS 8 (Community Enterprise Operating System). CentOS is forked from RedHat Linux, a Linux Distro fine-tuned for servers.

CentOS 8 is now shipping for 64 bit platforms, and currently there is no 32 bit ISO image. This is primarily due to the fact that most computers in production are 64 bit.

Therefore CentOS can be installed on most relatively-modern bare-metal computers with x86_64 architecture (Intel, AMD) like laptops, desktops or servers. Alternatively CentOS can also be installed on virtual machines (VM) under Intel-based hypervisors like commercial VMware or public ones like VirtualBox.

The installation of DevEnv is based on PUPPET, an open-source configuration management tool. The RPM **puppet-elt** contains the puppet scripts to complete and verify the DevEnv installation, consisting of:

- Installation of supporting RPMs
- Configuration of services
- Creation of user accounts for software development

4.1 The ESO/ELT CentOS-8 repository

The current release of the ELT Linux DevEnv is based on the original CentOS-8 image with a minimal installation of software packages or RPMs (RPM Package Manager).

The real DevEnv installation consists in the download and installation of software packages relevant to the development of the ELT project. Most of these packages (RPMs) were originally retrieved from public CentOS and EPEL (Extra Packages for Enterprise Linux) repositories. In addition to these RPMs, the DevEnv also installs other public-domain tools and ESO packages not available at public repositories. The process should guarantee the same result in DevEnv installations done at ESO or by external contractors.

Traceability: For testing and support purposes it is necessary to provide a mechanism that can retrieve and install any of the previous releases of the DevEnv. In reality it means to have access to all RPMs referred by previous installations of the DevEnv. For this reason, the DevEnv cannot rely on public repositories like CentOS and EPEL; they do not support traceability as old RPMs might be removed with new ones.

The DevEnv installation process has to rely on its own repositories, compiled and maintained at ESO and available to external users at:

- ftp://ftp.eso.org/pub/elt/ELTREPO/8
- ftp://ftp.eso.org/pub/elt/ELTREPO/epel/8
- ftp://ftp.eso.org/pub/elt/ELTREPO/CentOS-8

At our ESO/ELT repositories we combined the packages downloaded from CentOS and EPEL repositories with other external public-domain packages and with those packages developed at ESO. With new releases of the DevEnv new RPMs are accumulated to our ESO/ELT CentOS-8 repository; old RPMs are never removed.

This repository is also the official channel to provide minor and patch releases to the DevEnv (A major release, e.g. a new OS, will require a new repository).

4.2 HW requirements

The minimum requirements for the installation of the ELT Linux DevEnv are:

- 4x CPUs x86_64 (Intel or AMD)
- 8GB RAM
- 100GB disk
- 1x NIC
- HW compatible with CentOS 8

Depending on your own requirements, e.g. disk-space, number of users, heavy usage of GUI components etc. or in your current HW configuration these requirements should be exceeded.

4.2.1 Configuring BIOS for low latency environments (RT)

If you plan to use the host for real-time applications (RT) you may need to set your BIOS options for low latency. Please notice that factory BIOS defaults are optimized to provide a good balance between performance and power efficiency for general-purpose environments. However there are environments where you many need to optimize your hardware for maximum throughput or lowest latency to provide optimal responsiveness where real-time responses are needed.

The available BIOS options may vary, depending upon server model, processor/memory architecture, and BIOS revision. You have to consult your Hardware Owner's Manual for more details.

For Dell PowerEdge 12th Generation Severs please follow recommendations for low latency as indicated in Dell document:

- https://www.eso.org/~eltmgr/configuring-low-latency-environments-on-dell-poweredge-12g-servers.pdf

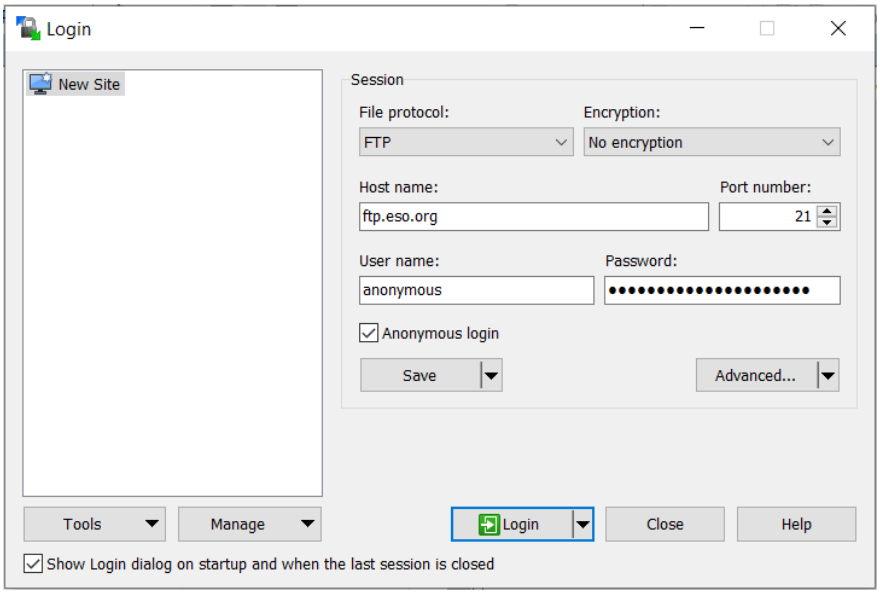
4.3 CentOS image and documentation

- The DVD ISO used for the ELT Linux DevEnv is the CentOS-8.2.2004-x86_64-dvd1.iso and can be downloaded from our ESO FTP area at:
  - ftp://ftp.eso.org/pub/elt/CentOS-8.2.2004-x86_64-dvd1.iso
  - with sha256sum: c87a2d81d67bbaeaf646aea5bedd70990078ec252fc52f5a7d65ff609871e255

- CentOS installation documents. There are many web pages that describe the CentOS 8 installation step by step, snapshots included. Here just a few of them:
  - http://www.tecmint.com/centos-8-installation/
  - https://linoxide.com/distros/how-to-install-centos/
  - https://www.howtoforge.com/tutorial/centos-8-minimal-server/
  - https://www.linuxtechi.com/centos-8-installation-guide-screenshots/
- Network configuration (valid also for CentOS 8):
  - https://lintut.com/how-to-setup-network-after-rhelcentos-7-minimal-installation/

You can download the DVD ISO image with the following applications:

- **wget** on Linux:
  - wget ftp://ftp.eso.org/pub/elt/CentOS-8.2.2004-x86_64-dvd1.iso
- **WinSCP** (https://winscp.net) on Windows 10 with an anonymous login as indicated in the snapshot below:



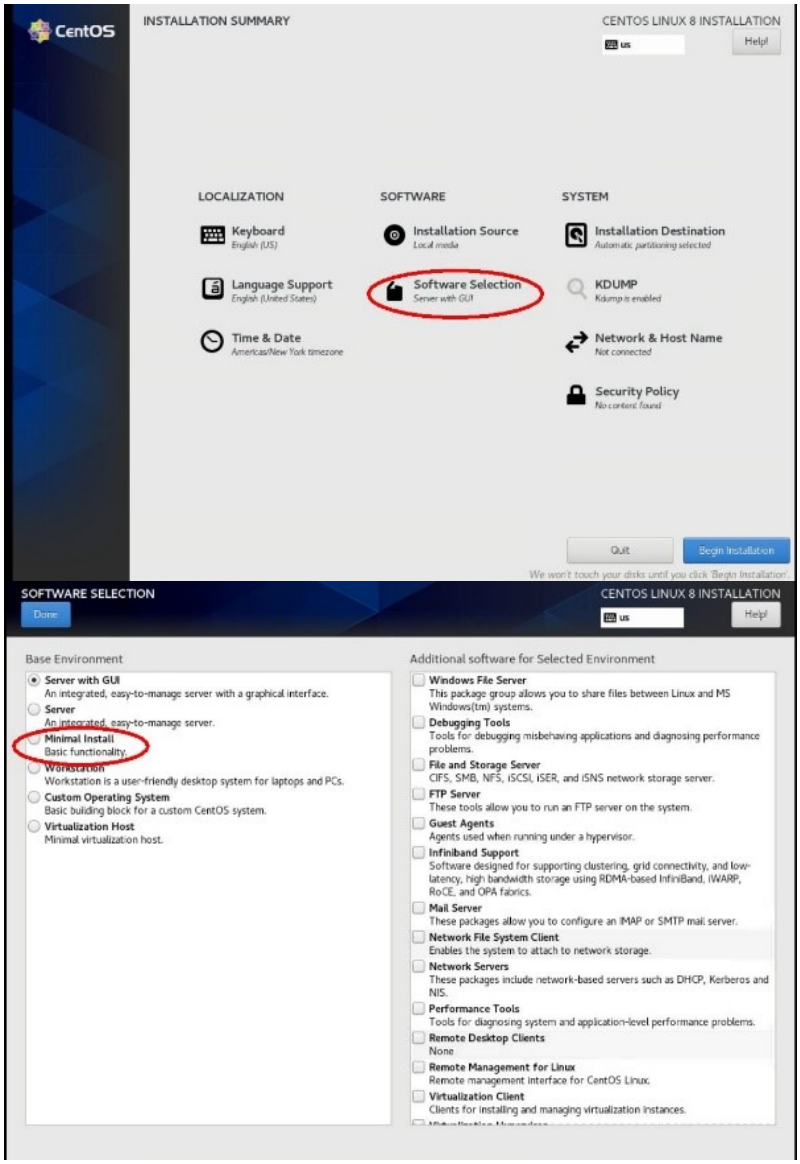
After login, navigate till pub/elt/CentOS-8.2.2004-x86_64-dvd1.iso

4.4 CentOS installation

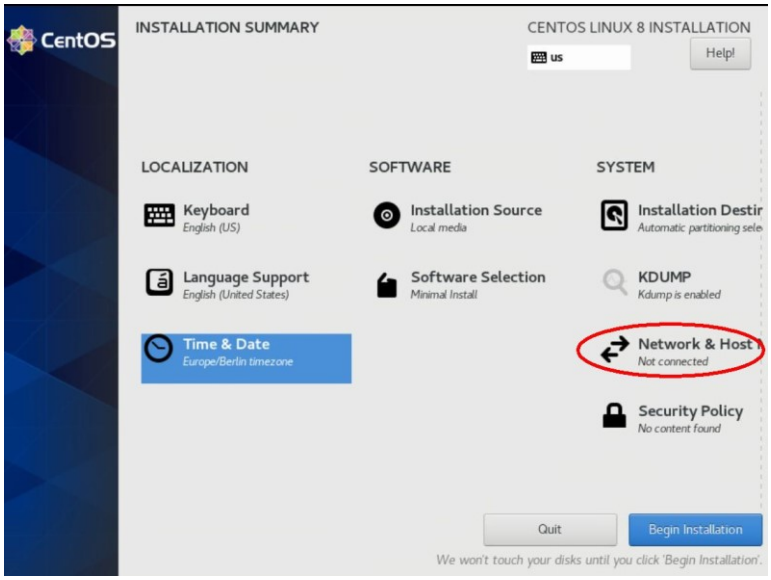
OS installation details are not part of this document. Please use any of the many documents already available in the web (see any of the list in the OS documentation section above).

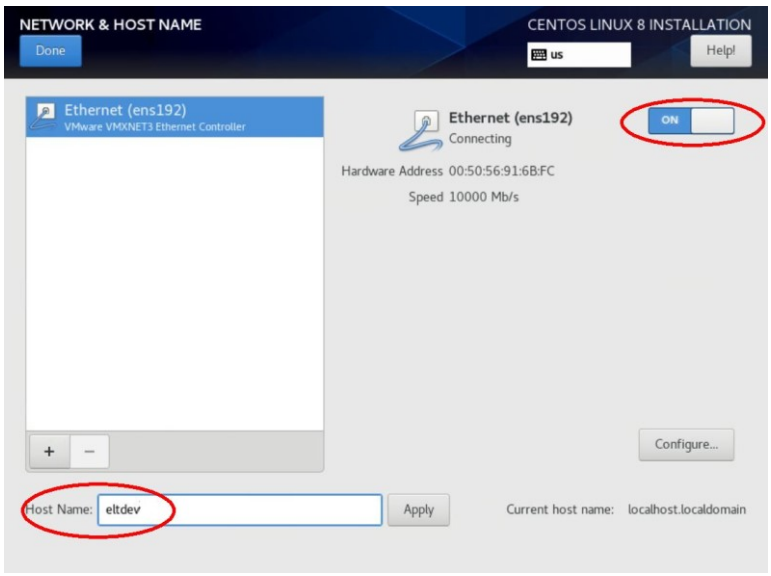
The ELT DevEnv requires the following setting (in **bold text** where input is required):

- LOCALIZATION
  - **DATE & TIME**: Set date and local time
  - KEYBOARD: *English (US)*
  - LANGUAGE SUPPORT: *English (United States)*
- SECURITY
  - SECURITY POLICY: *No profile selected*
- SOFTWARE
  - INSTALLATION SOURCE: *Local media*
  - **SOFTWARE SELECTION: Minimal Install**



- SYSTEM
  - **INSTALLATION DESTINATION**: First disk, with Automatic partitioning selected (*or any other disk partition that better fits your needs, reserving at least 50GB for the "/" partition*). Please, see APPENDIX A, is you plan to run elatsticsearch in this host.
  - KDUM: *Kdum is enabled*
  - **NETWORK & HOSTNAME**: Configure your NIC (DHCP or static) and select a hostname, e.g. eeltdev





- **Begin Installation** when above selections are completed
- USER SETTINGS
  - **ROOT PASSWORD**: *Set root password*
  - USER CREATION: *No user will be created*
- **Finish installation**
- **Reboot (*)**

(*) NOTE: You might need to stop the system and dismount the DVD before rebooting. In same cases, the DVD remains mounted and next reboot will bring you back to the CentOS installation.

After rebooting the system will presents the following text prompt:

```
CentOS Linux 8 (Core)
Kernel 3.10.0-327.el7.x86_64 on an x86_64
```

```
eltdev login:
```

Login as **root** with the password selected during the OS installation:

```
eltdev login: root
Password: <your_password>
Last login: <date & time> on tty1
[root@eltdev ~]#
```

WARNING: DO NOT execute any "yum update" in this Minimal CentOS host as you might get newer RPMs than those required and provided in our DevEnv distribution. The installation of DevEnv is unable to downgrade them. If this happens you have two alternatives: remove the newer RPMs by hand (it is risky and not always possible), or start the installation from the beginning.

To complete the ELT Linux installation is necessary for the machine to have access to internet. If the network configuration during the OS installation was skipped or incomplete you can still configure it with the text user interface tool

```
[root@eltdev ~]# nmtui
```

If you are not familiar with the command **nmtui** request the assistance of your sysadmin, or follow instructions available in the web, e.g. https://lintut.com/how-to-setup-network-after-rhelcentos-7-minimal-installation/

Alternatively you can edit or check the network files by hand:

- **/etc/hostname** # Contains the fully qualified name of the host, e.g. eltdevenv.hq.eso.org
- **/etc/sysconfig/network-scripts/ifcfg-ens192** # contains the configuration of the first NIC (ensxxx in VMware or enpxsx in VirtualBox:

As example only, this could be the typical configuration with DHCP:

```
TYPE="Ethernet"
PROXY_METHOD="none"
BROWSER_ONLY="no"
BOOTPROTO="dhcp"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
IPV6_ADDR_GEN_MODE="stable-privacy"
NAME="enp0s3"
UUID="e8e51bdb-09a3-4183-bdf0-aad31b967c6e"
```

```
DEVICE="enp0s3"
ONBOOT="yes"
IPV6_PRIVACY="no"
```

And this the typical configuration with an STATIC IP:

```
HWADDR=00:50:56:91:45:6d
NAME=ens192
GATEWAY=134.171.2.254
DNS1=134.171.7.11
DOMAIN=hq.eso.org
DEVICE=ens192
ONBOOT=yes
USERCTL=no
BOOTPROTO=static
NETMASK=255.255.255.0
IPADDR=134.171.2.118
PEERDNS=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
```

And after any modification of network parameters execute a reboot or the command:

```
[root@eltdev ~]# systemctl restart network
```

If you are installing the ELT DevEnv as a VM on a local hypervisor as VirtualBox or VMware Worsktation, please notice that the network card of your VM can be created basically with two options:

- **NAT.** The guest VM shares the IP of the host. With the internet access of the host but the guest is isolated from the world. The guest is typically configured with DHCP. This is the default
- **Bridged.** The guest VM has direct access to the network card of the host. With your own internet access and the guest can be accessed from the world. You can chose DHCP or a static IP.

4.5 Installation of puppet packages

With the minimal installation of CentOS completed and with the system connected to internet it is possible now to download the RPMs required to install PUPPET and the installation scripts corresponding to the latest release of the DevEnv. ------

To install PUPPET packages and scripts execute:

```
[root@eltdev ~]# yum -y install ftp://ftp.eso.org/pub/elt/wget.rpm --disablerepo="*"
[root@eltdev ~]# cd /tmp
[root@eltdev ~]# wget ftp://ftp.eso.org/pub/elt/puppet-elt-latest/\*.rpm
[root@eltdev ~]# rpm -ihv *.rpm
```

Note: if the command fails due to some RPMs already installed, remove them from /tmp and repeat the command.

After RPMs get installed, you can remove them from /tmp:

```
[root@eltdev ~]# rm -f *.rpm
```

4.6 Installing the ELT DevEnv with puppet

The RPM **puppet-elt** delivers two puppet scripts and their supportive files:

- **/root/elt/puppet-force-align**
  - --help|-h prints this text
  - --verbose|-v Puppet script executed with –verbose option. Log is also sent to terminal
  - --debug|-d Puppet script executed with –debug option.
- **/root/elt/puppet-check**

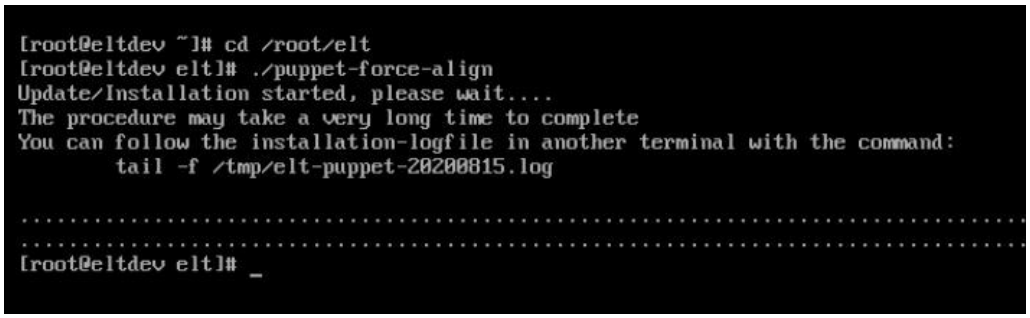
If you need a particular variant of ELT DevEnv there are some environment variables defining them. The name of the variant has to be exported before proceeding with the effective installation.

If nothing is indicated, the puppet installation assumes these default values:

```
ELT_ROLE=ELTDEV
ELT_REALTIME=NO
ELT_DM=NO
```

To proceed with the installation as root, execute the **puppet-force-align** to execute the installation and puppet-check to verify it:

```
[root@eltdev ~]# cd /root/elt
[root@eltdev ~]# ./puppet-force-align
```



The script may takes 40 minutes or more to complete; it depends largely in the internet speed and the connection to the ESO/ELT repository. The screen-shot above shows a complete installation with no errors. A “progress indicator” in the form of “dots” every 5 seconds indicates the script is still progressing.

The procedure will always generate a log-file that can be used in case of problems. The log-file contains detailed information about each step performed and is located in the /tmp directory. The log-file is named `elt-puppet-YYYYMMDD.log` where YYYYMMDD is the string describing the year, month and day of execution. If puppet-force-align is executed again in the same day the results will be appended to the existing log-file.

Please notice that the first time the script is executed it will generate many “Errors” and “Warnings” lines in the logfile. These are completely normal as the script first reports the error before correcting it.

The values of ELT development variables are saved by the installation script in the shell profile configuration file: **/etc/profile.d/eltdev.sh**. They will be loaded in the environment with the next login in the system as a reminder of the current installation. You can still change the configuration, with certain limitations, by re-exporting a new value and re-executing the puppet script **/root/elt/puppe-force-align**. See the different options below:

4.6.1 The ELT_ROLE environment variable

It defines the scope of the installation. The currently known variants for the ELT_ROLE in the DevEnv are:

- ELTDEV: Software development workstation. Includes tools and packages necessary for a standard developer. This is the default choice if nothing is indicated.
- MINIMAL: a minimal installation; for use in testing hosts, e.g. Jenkins. It is subset of ELTDEV.

```
[root@eltdev ~]# export ELT_ROLE=MINIMAL
[root@eltdev ~]# /root/elt/puppet-force-align
```

Note: You can upgrade a MINIMAL installation to ELTDEV, but not viceversa.

4.6.2 The ELT_REALTIME environment variable

It defines if the installation should or shouldn't include the real-time extensions. It has two unique options:

- NO: no real-time workstation. This is the default value.
- YES: install real-time extension. This variant contains a real-time kernel, packages specific to real-time development and other real-time specific system configurations. If you do not know which variant you need, take the default ELTDEV.

```
[root@eltdev ~]# export ELT_REALTIME=YES
[root@eltdev ~]# /root/elt/puppet-force-align
```

With the ELT_REALTIME=YES reboot the machine to be able to run with the new kernel. A reboot is also recommended the first time puppet-force-align is executed after the OS installation:

```
[root@eltdev ~]# reboot
```

Note: You can upgrade a real-time workstation but not viceversa. At the time of writing only ELT projects related to M1 LCS do require the REALTIME variant.

4.6.3 The ELT_DM environment variable

DM stands for "Display Manager" and it sets or doesn't set the execution of a daemon for a display manager in the console with these values:

- NO: no Display Manager will be launched after the installation. This is the default value.
- YES: Display Manager will be launched after the installation.

If you want the installation to finish with the launch of a display manager, execute:

```
[root@eltdev ~]# export ELT_DM=yes
[root@eltdev ~]# /root/elt/puppet-force-align
```

Note: you can change for "YES" to "NO" and viceversa.

4.7 Verifying the ELT DevEnv installation

The installation script **puppet-force-align** finishes with the invocation of the verfication script **puppet-check**, however you can execute puppet-check any time later to verify the system remains complaint with the ELT DevEnv:

```
[root@eltdev ~]# cd /root/elt
[root@eltdev ~]# ./puppet-check
```

This is the screen-shot of the execution of puppet-check when everything is correct:

```
[root@eltdev elt]# ./puppet-check
Info: Loading facts
Info: Loading facts
Notice: Scope(Node[default]): Installing/checking basic ELTDEV station
Notice: Compiled catalog for eltdev.hq.eso.org in environment production in 2.02 seconds
.Info: Applying configuration version '1597505054'
.......Notice: Applied catalog in 37.69 seconds
.[root@eltdev elt]#
```

4.8 ELT DevEnv releases

The ELT Linux DevEnv is defined by the environment variable $ELT_RELEASE. It follows standard numeration **major.minor.patch**-iteration, where an increase in:

- the **major** number indicates a considerable change largely affecting the complete DevEnv, e.g. with a new OS. (2: CentOS-7, 3: CentOS-8).
- the **minor** number indicates a change affecting important components of the DevEnv, like toolkits or their releases.
- the **patch** number will be used to fix a severe bug of a minor release.
- The **iteration** number is meaningless. Only for integration purposes.

Typically DevEnv will be released with the following frequency:

- **Major** releases will be released with a frequency of at least 6 months. A major release will be announced with at least two months in advance.
- **Minor** releases are expected to be released with a frequency of months
- **Patch** releases will be made available to fix a severe bug in a Minor release.

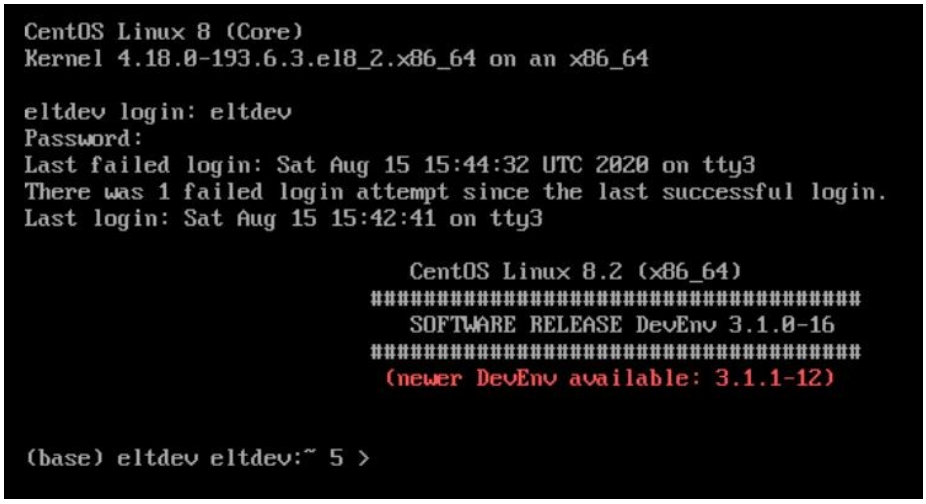
To know which release of the DevEnv is installed on any system execute:

```
[root@eltdev ~]# echo $ELT_RELEASE
```

The release is also indicated in the login welcome message at **/etc/motd.**

```
CentOS Linux 8 (Core)
Kernel 4.18.0-193.el8.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

eltdev login: eltdev
Password:

                    CentOS Linux 8.2 (x86_64)
                ########################################
                  SOFTWARE RELEASE DevEnv 3.0.0-16
                ########################################


(base) eltdev eltdev:~ 1 > _
```

The “Message Of The Day” might also indicates if there is a newer release of the DevEnv available for download from our repository. Example:



4.9 Updating the ELT DevEnv

To check if there are newer releases available, execute as root:

```
[root@eltdev ~]# yum makecache
[root@eltdev ~]# yum --showduplicates list puppet-elt
```

The first command is needed to dowload the latest metadata from repositories. The option `--showduplicates` will list all available releases in the repository; and in green font the current release. You can install an upgrade to any newer release by explicitely indicating that release:

```
[root@eltdev ~]# yum -y update puppet-elt-3.1.1
```

or to the latest release:

```
[root@eltdev ~]# yum -y update puppet-elt
```

After the installation of a newer or latest **puppet-elt** RPM, complete the upgrade with the execution of the remaining puppet installation and verification:

```
[root@eltdev ~]# ./puppet-force-align
[root@eltdev ~]# ./puppet-check
```

4.10 Installing an older release

Instead proceed from scratch with the installation of CentOS as described in section 4.4.

Choose one of the DevEnv releases available from the ESO anonymous FTP server:

- **ftp://ftp.eso.org/pub/elt**

Please notice that **puppet-elt-latest** is soft link pointing to the latest release of the DevEnv.

Install the puppet packages as described in section 4.5 but changing string **"latest"** with the numeration of the release of your choice, e.g. to install release **3.1.1**, execute:

```
[root@eltdev ~]# yum -y install \
ftp://ftp.eso.org/pub/elt/wget-1.19.5-8.el8_1.1.x86_64.rpm --disablerepo="*"

[root@eltdev ~]# cd /tmp
[root@eltdev ~]# wget ftp://ftp.eso.org/pub/elt/puppet-elt-3.1.1/\*.rpm
[root@eltdev ~]# rpm -ihv *.rpm
[root@eltdev ~]# rm -f *.rpm
```

Complete the installation and verification with sections 4.6 and 4.7.

4.11 Downgrading to an older release

There is a procedure supporting the downgrade of your DevEnv release system to an older previously running version. Example, if your system was upgraded from release 3.1.1 to new 3.1.3 following the instruction in chapter 4.9, you can, after the upgrade, downgrade back to your previous release 3.1.1 with the following sequence:

```
[root@eltdev ~]# cd /root/elt
[root@eltdev ~]# ./puppet-downgrade puppet-elt-3.1.1
```

The process is guaranteed under certain limitations:

- Rollback or downgrade of the following packages is unsupported as the packages themselves, and dependencies, either assume an update-only or install-only process: `dbus, kernel, glibc or selinux-policy`
- The downgrade to a given release can only be done on hosts updated previously from that release. You cannot downgrade to a given older version if the host never was updated from that given version.
- There are some inherited risks to the downgrade process: x it may leave some orphan files not cleared by the undo process.
- The process will be verified to work between two consecutive releases, i.e. from the current release to the previous one. Any attempt to downgrade to an older release is not guaranteed and can only be done at your own risk.
- Only packages installed with YUM are considered in the downgrade process, and the process is unable to distinguish between packages installed or updated by the update DevEnv process from other packages installed with YUM by hand. The downgrade process might remove/downgrade them all.

5 Development User accounts

The **puppet-force-align** script creates two user accounts:

- Username: **eltdev** password: **2Garch1ng**
- Username: **eltmgr** password: **pass4u**

eltdev is the standard development account. Other similar accounts can be created for other developers.

eltmgr is an administrative account which tasks, in this release, are not yet defined.

REMARK: Please note that these account-names substitutes the account-names *eeltdev* and *eeltmgr* accounts defined in previous DevEnv releases 2.1.x and 2.2.x with CentOS-7.

6 Support

For ESO internal use only: Support to the ELT Linux DevEnv installation will be provided via creating a new ticket in our JIRA ticketing system **EELTMGR** or with an email to eeltmgr@eso.org

For external users and developers: please send any problem regarding the ELT Linux DevEnv installation to the JIRA ticketing system associated to your project.

Before reporting a problem check first if a new DevEnv is available and install it to see if that fixes or solves your problem. To check if a new release is available execute:

```
[root@eltdev ~]# yum check-update puppet-elt
```

If a new release is available and you decide to install I then execute:

```
[root@eltdev ~]# yum -y update puppet-elt
[root@eltdev ~]# cd /root/elt
[root@eltdev ~]# ./puppet-force-align
[root@eltdev ~]# ./puppet-check
```

On reporting a problem or requesting support please do not forget to mention your DevEnv release. To obtain the release execute:

```
[root@eltdev ~]# rpm -q puppet-elt
```

Attach to your problem description the output of the verification procedure **puppet-check**:

```
[root@eltdev ~]# cd /root/elt
[root@eltdev ~]# ./puppet-check
```

And if available also attach the log file generated by the **puppet-force-align** execution as described in section 4.6.

7 Appendix A. Changing disk partitioning

The default disk partitioning for the system disk provided by CentOS should be sufficient for a standard installation with:

- SYSTEM
  - / 50 GiB
  - */boot/efi* 600 MiB
  - */boot* 1024 MiB
  - *swap* <same as RAM for RAM<8GB, otherwise ½ RAM)

- DATA
  - /home (the rest of the disk)

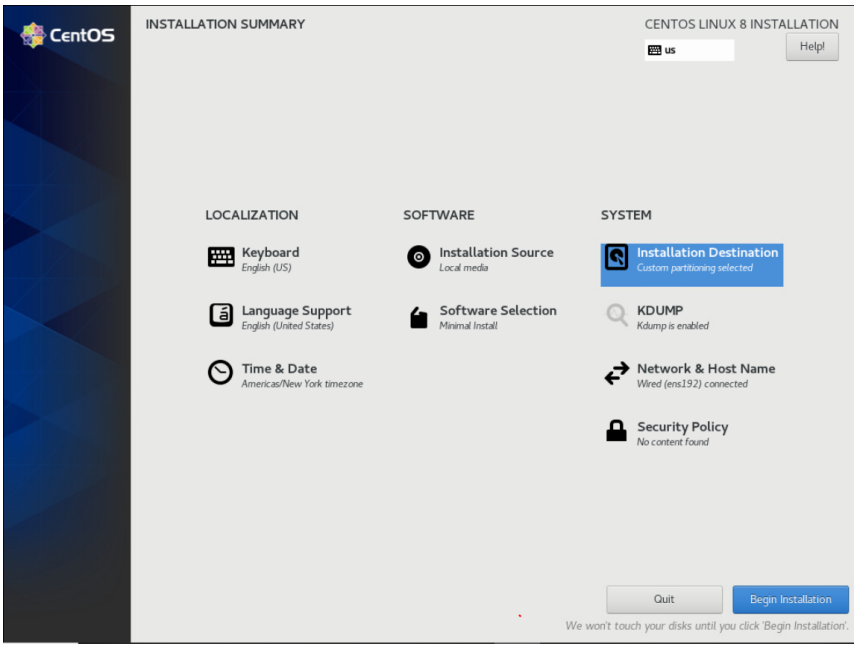
In this default partitioning the directory *"/var"* is located under *"/"* which is set to 50 GiB.

However some applications, for example elasticsearch, a search engine used by CII which may have a high demand of disk space under "*/var/lib/elasticsearch",* may cause the root partition *"/"* to get full, causing the system to get stuck with a full-disk condition.

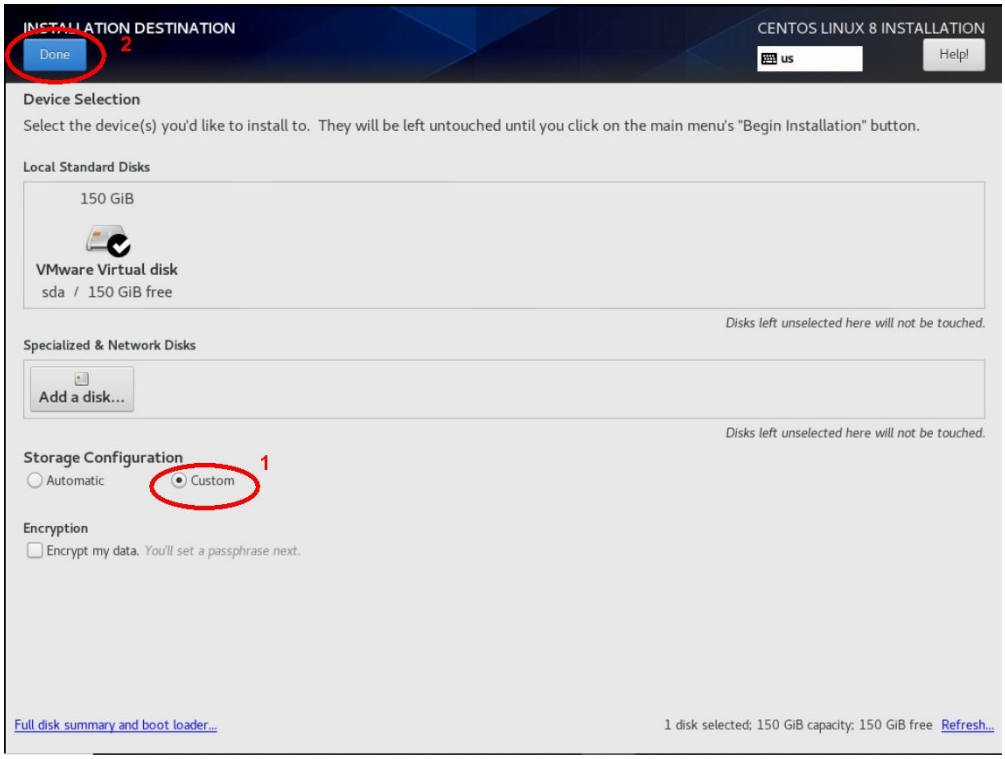
To avoid the full-disk condition in *"/"* , a good idea in such cases is to mount *"/var"* in its own partition.

This section describes how to create an extra *"/var"* partition of 50 GiB on a system disk of 150 GiB, just before the installation of CentOS-8 as described in section 4.4 CentOS installation:

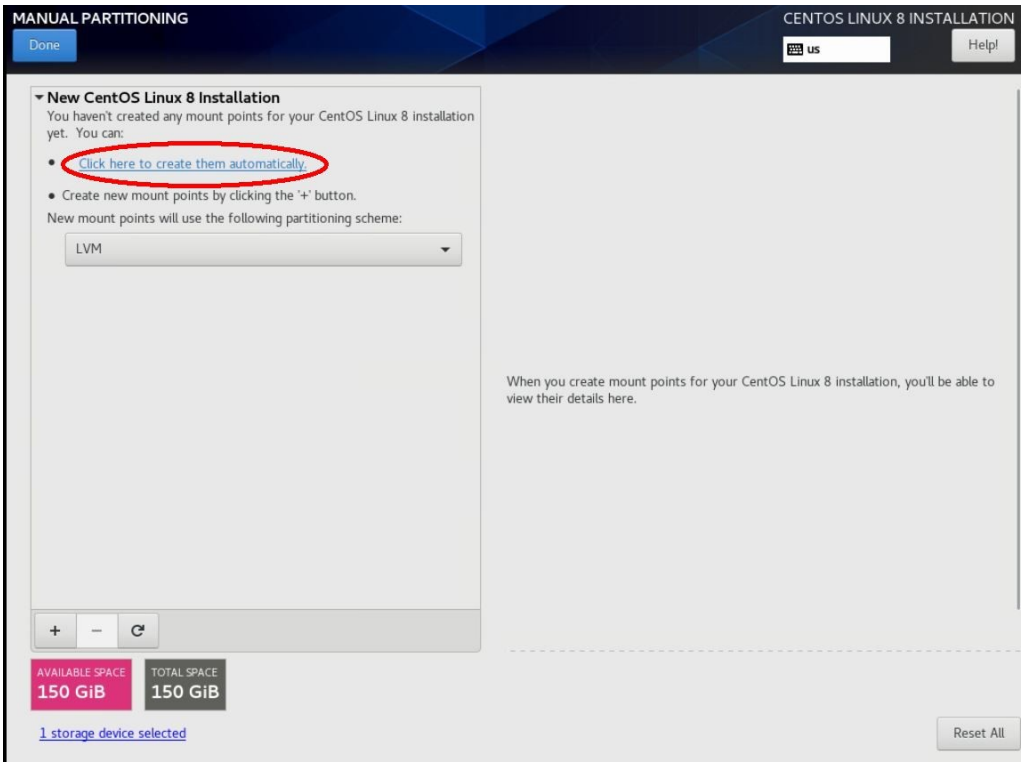
1. Select "Installation Destination"



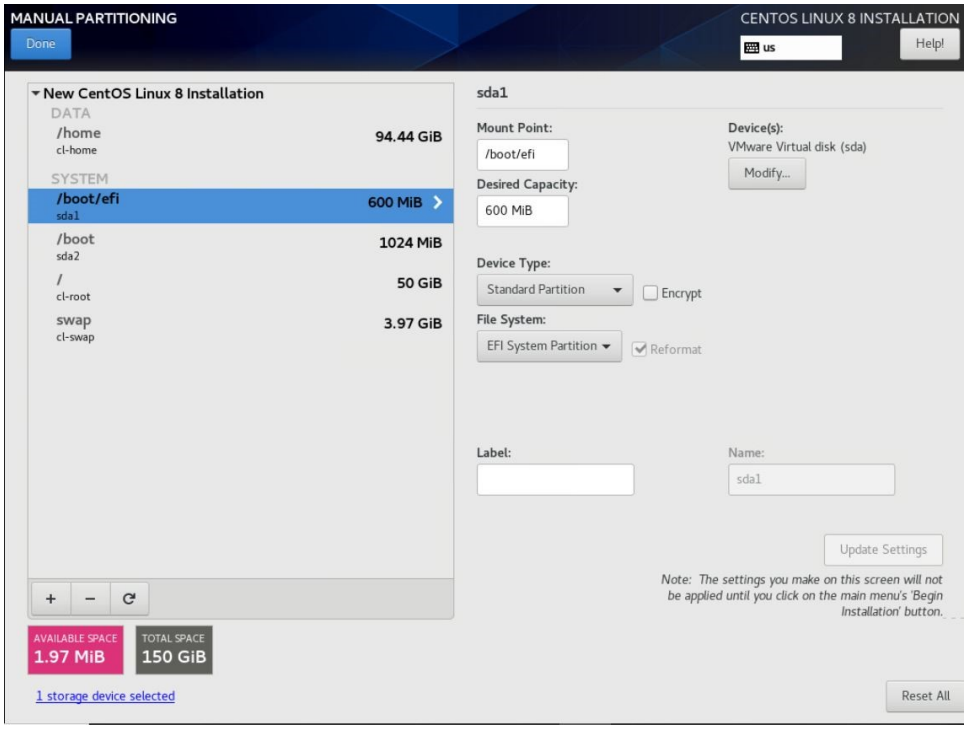
2. In the INSTALLATION DESTINATION panel, select first Storage Configuration: **Custom,** then press **Done** as in following snapshot:



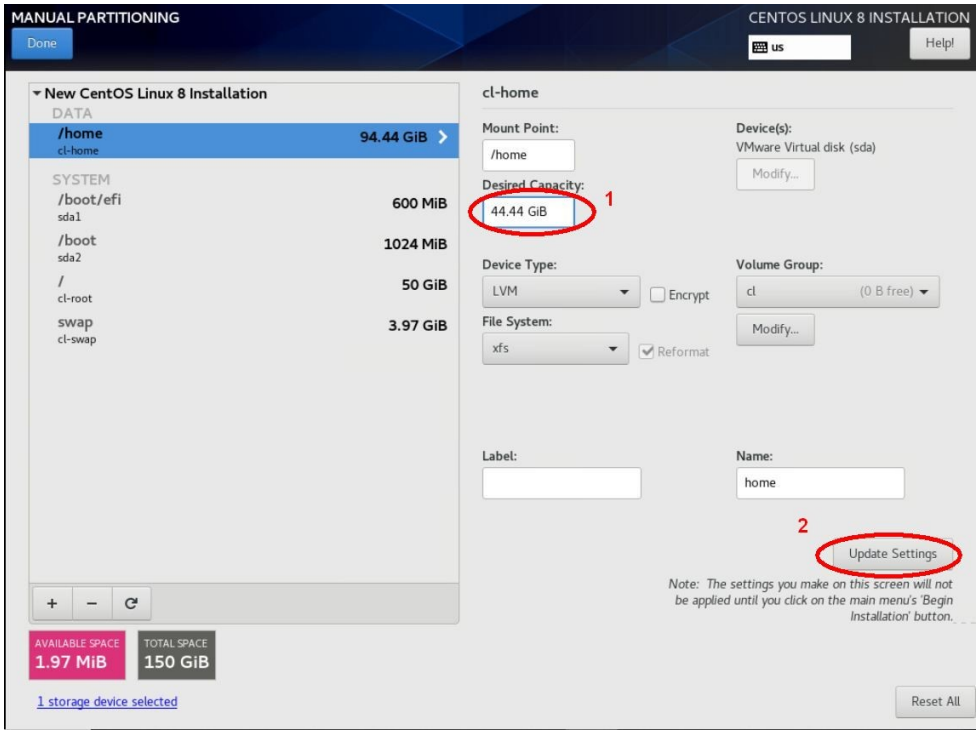
3. In the MANUAL PARTITIONING panel, click in the blue-text “Click here to create them automatically” as indicated in:



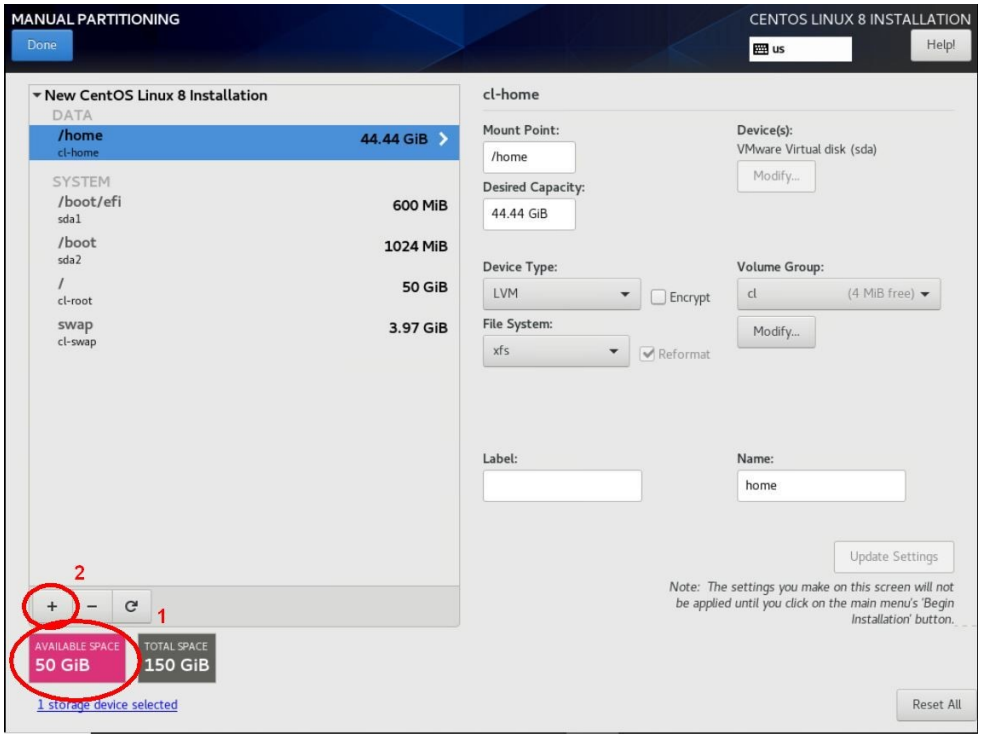
4. The full disk will be already fully partitioned (only 1.97 MiB remains available), so the next step, is to reduce one of the existing partitions to free disk space for *"/var"*



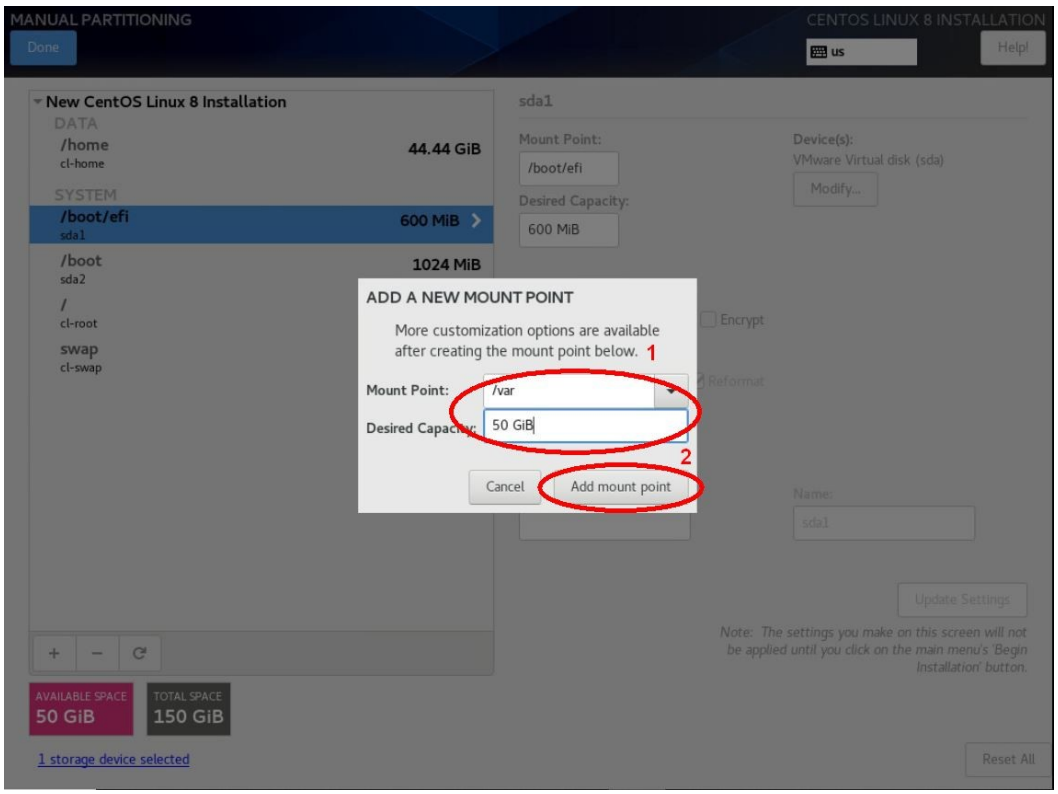
5. Select partitioning *"/home"* and decrease its current size of 99.44 GiB by 50 GiB which will then be assigned to *"/var"*. Follow the sequence indicated in this snapshot:



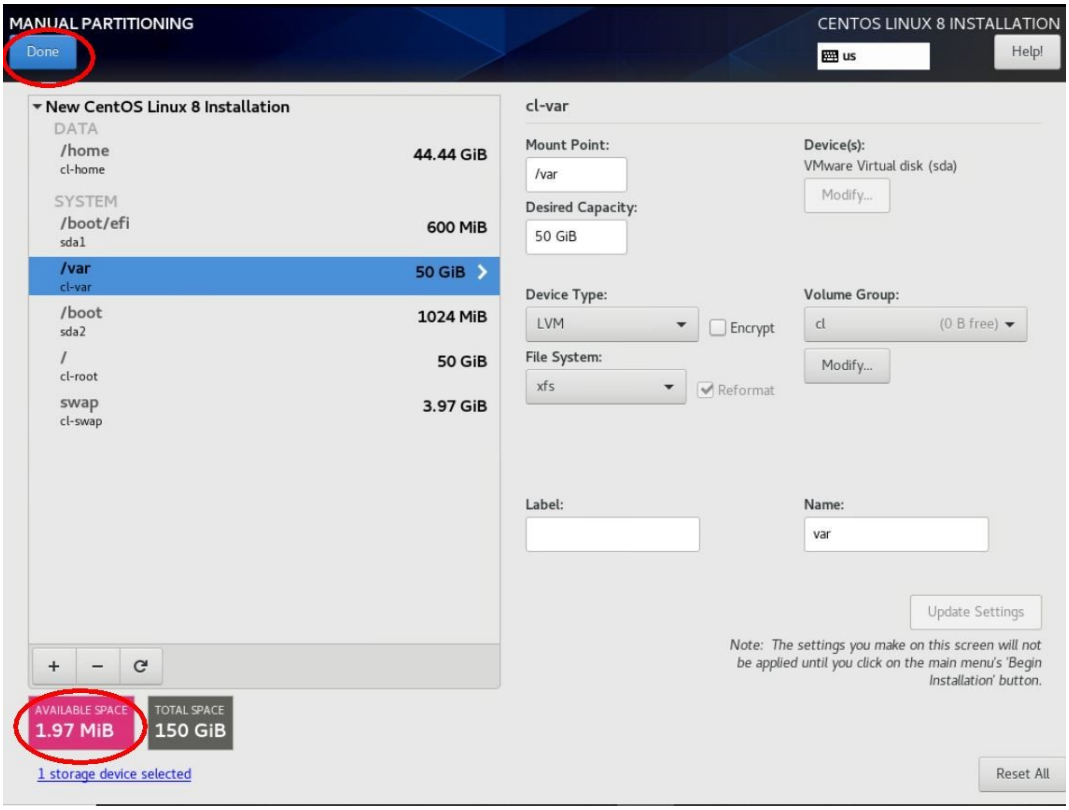
6. The MANUAL PARTITIONING panel is now updated and it shows “AVAILABLE SPACE: 50 GiB”. Next step is to assign this free disk space to a new partition *“/var”*. Click in button **“+”**



7. Enter the text and values as indicated in this snapshot, and click in **“Add mount point”**



8. The MANUAL PARTITIONING window is now updated, indicating the new partition *"/var"* and its size. The AVAILABLE SPACE gets also updated indicating almost no space available. Click on **Done** to return to the main panel.



9. The main installation panel is shown and now you can continue with the rest of the OS installation as indicated in Section 4.4: