 

**Como criar uma distribuição Linux personalizada.**

1. Vamos precisar de uma maquina com o CentOS 7 instalado.

Se não e a sua distrubição padão, a qual você já usa em sua maquina, podemos fazer isso em uma VM.

Vamos precisa de um espaço máximo de 20 GB de hd.

1. Prepare your build environment

If the installation of your CentOS 7.1 virtual machine is finished you can continue with preparing your build environment.

2.1 Create needed folder structure as root

mkdir -p ~/kickstart\_build/isolinux/{images,ks,LiveOS,Packages,postinstall}

The folders will contain the following:

images: contents of the images directory located on the CentOS DVD

ks: all your kickstart files which we will create later on

LiveOS: contents of the LieveOS directory on the CentOS DVD

Packages: all RPM packages from CentOS 7 DVD plus additional packages. In my case I will also install Puppet agent. Therefore I need some packages from Puppet Labs

postinstall: everything you want so to do after installation, for example executing custom scripts or in my case Puppet modules.

2.2 Copy needed content

Now you need to copy all needed content from the CentOS DVD to your local folders. Please save the CentOS 7 ISO file in /tmp and mount it somewhere.

mkdir -p /mnt/iso

mount -o loop /tmp/CentOS-7-x86\_64-DVD-1503-01.iso /mnt/iso

cp /mnt/iso/.discinfo ~/kickstart\_build/isolinux/

cp /mnt/iso/isolinux/\* ~/kickstart\_build/isolinux/

rsync -av /mnt/iso/images/ ~/kickstart\_build/isolinux/images/

cp /mnt/iso/LiveOS/\* ~/kickstart\_build/isolinux/LiveOS/

ll /mnt/iso/repodata/ | grep -i comps

-rw-r--r--. 1 root root 157580 1. Apr 01:43 0e6e90965f55146ba5025ea450f822d1bb0267d0299ef64dd4365825e6bad995-c7-x86\_64-comps.xml.gz

cp /mnt/iso/repodata/0e6e90965f55146ba5025ea450f822d1bb0267d0299ef64dd4365825e6bad995-c7-x86\_64-comps.xml.gz ~/kickstart\_build/

cd ~/kickstart\_build/

gunzip 0e6e90965f55146ba5025ea450f822d1bb0267d0299ef64dd4365825e6bad995-c7-x86\_64-comps.xml

mv 0e6e90965f55146ba5025ea450f822d1bb0267d0299ef64dd4365825e6bad995-c7-x86\_64-comps.xml comps.xml

2.3 Get additional packages if needed

As I already mentioned I will also install Puppet Opensource client on my machines to install and configure my machines as needed. So I need to implement additional packages on the custom boot ISO to be able to install and run Puppet manifests.

mkdir /tmp/packages

cd /tmp/packages

wget http://mirror.centos.org/centos/7/os/x86\_64/Packages/libselinux-ruby-2.2.2-6.el7.x86\_64.rpm

wget -e robots=off --mirror --no-parent --no-host-directories --cut-dirs=4 http://yum.puppetlabs.com/el/7/products/x86\_64/

wget -e robots=off --mirror --no-parent --no-host-directories --cut-dirs=4 http://yum.puppetlabs.com/el/7/dependencies/x86\_64/

rm -Rf index\* repodata

2.4 Copy all your packages and create repodata

Now it`s time to bring the CentOS packages and your additional needed packages together. First copy all packages from CentOS 7 ISO to your local folder. Afterwards you can move all additional packages from /tmp/packages to your kickstart packages folder:

rsync -av /mnt/iso/Packages/ ~/kickstart\_build/isolinux/Packages/

rsync -av /tmp/packages/ ~/kickstart\_build/isolinux/Packages/

Now we need to create the repodata folder in ~/kickstart\_build/isolinux/Packages/:

yum install -y createrepo

cd ~/kickstart\_build/isolinux

createrepo -g ~/kickstart\_build/comps.xml .

3) Prepare Kickstart file

3.1 Create a kickstart

You need to create the kickstart file in ~/kickstart\_build/isolinux/ks and name it for example ks.cfg. The content can look like this:

#version=RHEL7

# System authorization information

auth --enableshadow --passalgo=sha512

# Use CDROM installation media

cdrom

# Use text install

install

text

# Run the Setup Agent on first boot

firstboot --disable

#ignoredisk --only-use=sda

# Keyboard layouts

keyboard --vckeymap=de-nodeadkeys --xlayouts='de (nodeadkeys)'

# System language

lang de\_DE.UTF-8

# Network information

network --bootproto=static --device=ens3 --noipv6 --activate --ip=192.168.100.2 --netmask=255.255.255.0 --gateway=192.168.100.1 --nameserver=192.168.100.1 --hostname=infrastructure.reimer.local

network --bootproto=static --device=ens9 --noipv6 --activate --ip=10.10.100.2 --netmask=255.255.255.0

# Root password

rootpw --iscrypted "some-crypted-password"

# System timezone

timezone Europe/Berlin --isUtc

# System bootloader configuration

bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda

# Partition clearing information

clearpart --all --initlabel

# Disk partitioning information

part /boot --fstype="xfs" --size=512

part pv.219 --fstype="lvmpv" --size 1 --grow

volgroup vg\_system --pesize=4096 pv.219

logvol / --fstype="xfs" --size=1 --grow --label="rootlv" --name=rootlv --vgname=vg\_system

logvol swap --fstype="swap" --size=2048 --name=swaplv --vgname=vg\_system

reboot

%packages

@core

@Base

kexec-tools

git

mc

screen

puppet

tree

%end

You can validate your kickstart file like this:

ksvalidator ~/kickstart\_build/isolinux/ks/ks.cfg

To create a crypted root password which you can use within your kickstart file do the following:

python -c 'import crypt; print(crypt.crypt("My Password", "$6$My Salt"))'

This generates a SHA512 crypted password.

3.2 Create kickstart postinstallation section

If you want to perform some postinstallation tasks within your kickstart installation you can add an appropriate section in the kickstart file. In my case I want to perform my Puppet configuration during the kickstart installation.

HINT: Be careful. In this case the first step will be the copy of the Puppet manifests from the ISO from which you boot your machine you want to kickstart. This action takes place in the NON-CHROOTED environment. The second step will be the Puppet run itself. This takes place in the CHROOTED environment.

Add the following at the end of your kickstart file:

# Copy needed Puppet files to /root/postinstall

%post --nochroot

#!/bin/sh

set -x -v

exec 1>/mnt/sysimage/root/kickstart-stage1.log 2>&1

echo "==> copying files from media to install drive..."

cp -r /run/install/repo/postinstall /mnt/sysimage/root

%end

%post

#!/bin/sh

set -x -v

exec 1>/root/kickstart-stage2.log 2>&1

ls -l /root/postinstall

puppet apply -l /root/puppetrun.log /root/postinstall/puppet/manifests/site.pp --modulepath=/root/postinstall/puppet/modules/ $\*

%end

4) Time for action: create your custom CentOS 7 ISO file and test it

yum install -y genisoimage

cd ~/kickstart\_build/

mkisofs -o centos-7-custom.iso -b isolinux.bin -c boot.cat -no-emul-boot -V 'CentOS 7 x86\_64' -boot-load-size 4 -boot-info-table -R -J -v -T isolinux/

Now start a new virtual machine from your custom CentOS 7 ISO file and insert the following option at kernel boot:

linux inst.ks=cdrom:/dev/cdrom:/ks/ks.cfg

Congratulations 🙂 Your kickstart installation should run.