Ender Mujica Díaz

emujicad@gmail.com

Summary

Document developed to describe the method for creating an image of Debian 8 operating systems for use with VIO.

Debian IMAGE  
CREATION for CLOUD

Debian 8

Tabla de contenido

[REQUIREMENTS 3](#_Toc490401928)

[1. Installation Files 3](#_Toc490401929)

[2. Virtual Machine 3](#_Toc490401930)

[OPERATING SYSTEM 4](#_Toc490401931)

[INSTALLING THE OPERATING SYSTEM 4](#_Toc490401932)

[preseed File 7](#_Toc490401933)

[1. Languaje, country and location 7](#_Toc490401934)

[2. Keyboard 7](#_Toc490401935)

[3. Network 7](#_Toc490401936)

[4. Mirror setting 7](#_Toc490401937)

[5. Account setup 7](#_Toc490401938)

[6. Clock and time zone setup 7](#_Toc490401939)

[7. Partitioning 8](#_Toc490401940)

[8. Apt setup – repositories and security update 8](#_Toc490401941)

[9. Package selection 9](#_Toc490401942)

[10. Boot loader installation 10](#_Toc490401943)

[11. Finishing up the installation 10](#_Toc490401944)

[12. Postinstall 10](#_Toc490401945)

[cloud FILE 10](#_Toc490401946)

[1. cloud.cfg 12](#_Toc490401947)

[STARTUP SCRIPT 14](#_Toc490401948)

[1. rc.local 14](#_Toc490401949)

The installation of the operating system of "debian 9" is based on the methods of installation unattended by webserver that it provides (Preseeding). Details of the debian installer can be located at:

<https://wiki.debian.org/DebianInstaller/Preseed>

# REQUIREMENTS

## Installation Files

To perform the unattended installation, the first thing that is required is a web server (eg apache2) and create a directory structure under the control of the web server that houses the necessary configuration and script files.

/var/www/html/recipes/linux/unattended

/var/www/html/recipes/linux/unattended/debian8

/var/www/html/recipes/linux/unattended/supportfiles/debian8

Into the /var/www/html/recipes/linux/unattended/ directory there is a file called **seed-cloud.cfg** that contains a template with all the instructions so that the image installation will be done in unattended way.

Within the /var/www/html/recipes/linux/unattended/supportfiles/ directory there two files: **cloud.cfg** and **rc.local.openstack**. These will be used by cloud-unit application and rc.local to execute a series of commands that configure additional details of the OS. In case of not running the installation in a cloud environment, grow the file system "/" to the maximum size of the disk that has been assigned to the server. The content of these files is explained later.

/var/www/html/recipes/linux/unattended/supportfiles/debian8/cloud.cfg

/var/www/html/recipes/linux/unattended/supportfiles/debian8/rc.local.openstack

The template file on the web server can be accessed and performed through a browser as follows::

http://WEBSERVER/recipes/linux/unattended/debian8/seed-cloud.cfg

Where WEBESERVER can be an FQDN or and IP address.

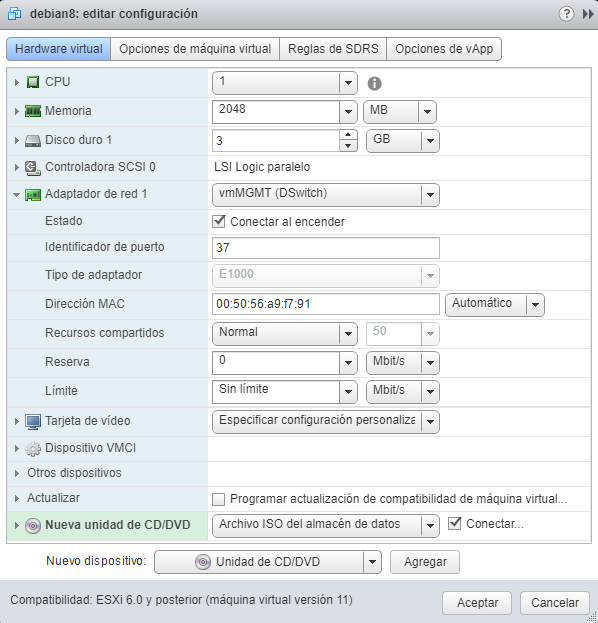
The structure can be changed at your convenience, just keep in mind that you must modify the seed-cloud.cfg file. This structure and files can also be placed in GitHub if you do not have a web server of your own.

## Virtual Machine

Create an image of a virtual machine with the following characteristics:

* 1 VCPU
* 2GB of RAM
* 3GB of space in VMDK format
* LSI Logic Parallel SCSI Controller.
* A network interface with e1000 driver.
* Compatible with ESX 6.0 or higher

In this case the image used was created on a VMware vSphere server using vCenter.

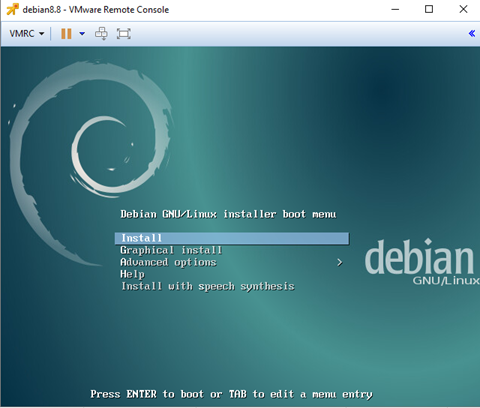


# OPERATING SYSTEM

An operating system image in iso format must be available. For this case it is recommended to use the net install file and that it has been configured to start with the virtual machine so that it is read at the time of the start.

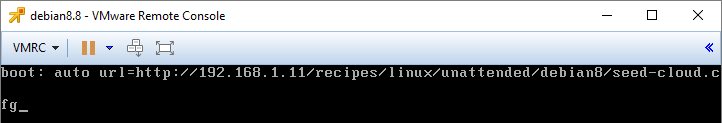
# INSTALLING THE OPERATING SYSTEM

Start the operating system from the ISO of the distribution and open a VMware virtual machine remote console. The following screen appears with the boot menu.



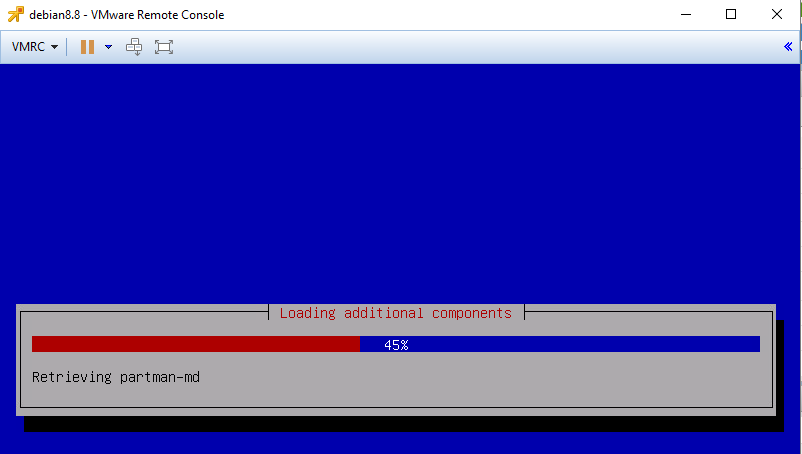
Press the ESC key to go to the basic prompt: "boot:". At the basic prompt, type the following line and press ENTER:

auto url=http://yourwebserver/recipes/linux/unattended/debian8/seed-cloud.cfg

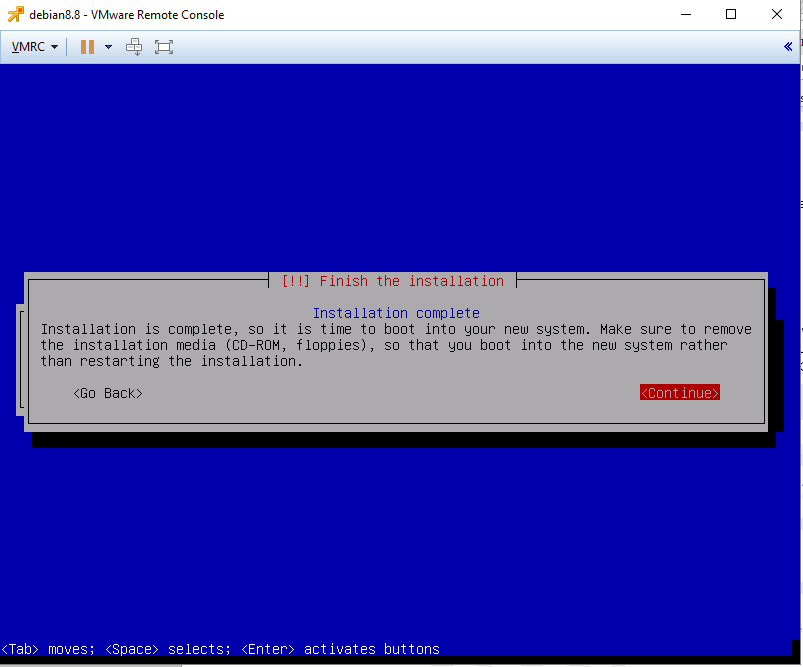


This method assumes that in your network there is a DHCP server, as provided by a cloud environment, and that you assign a temporary IP address and later configure the machine interface to invoke an IP address.

Once it starts, the process that will install the operating system on the virtual machine will begin.



Wait until it's over



# preseed File

The file is based on the preseed documentation and should be located at:

https://www.debian.org/releases/stable/amd64/apbs04.html.en

It is divided into sections to improve your understanding

## Languaje, country and location

d-i debian-installer/language string en

d-i debian-installer/country string US

d-i debian-installer/locale string en\_US.UTF-8

These lines define the American English language

## Keyboard

d-i keyboard-configuration/layoutcode string us

d-i keyboard-configuration/xkb-keymap select us

Keyboard is defined US-English

## Network

Hostname and domain is defined

d-i netcfg/get\_hostname string localhost

d-i netcfg/get\_domain string localdomain

Disable the annoying WEP key dialog screen.

d-i netcfg/wireless\_wep string

## Mirror setting

d-i mirror/protocol string http

d-i mirror/country string manual

d-i mirror/http/hostname string ftp.us.debian.org

d-i mirror/http/directory string /debian/

d-i mirror/http/proxy string

The USA mirror is set, however you can choose one different

## Account setup

The root account is disabled and a normal account is created

d-i passwd/make-user boolean true

d-i passwd/root-login boolean false

Create an aditional (normal) user, called debian

d-i passwd/user-fullname string Debian User

d-i passwd/username string debian

The debian user password is in clear text. Do not worry about that, this account will be disable by cloud-init

d-i passwd/user-password password P@ssW0rd

d-i passwd/user-password-again password P@ssW0rd

## Clock and time zone setup

The clock is not on UTC

d-i clock-setup/utc boolean false

The default zone is Etc/UTC, however can be changed it at convenience)

d-i time/zone string Etc/UTC

d-i clock-setup/ntp boolean true

d-i clock-setup/ntp-server string pool.ntp.org

## Partitioning

All the partition are deleted, with out confirmation dialog.

d-i partman-auto/method string regular

d-i partman-auto/purge\_lvm\_from\_device boolean true

d-i partman-lvm/device\_remove\_lvm boolean true

d-i partman-md/device\_remove\_md boolean true

d-i partman-lvm/confirm boolean true

The partitioning for cloud do not use LVM nor swap, so disable the warning and question about not having swap partition.

d-i partman-basicfilesystems/no\_swap boolean false

Recipe por partitioning included only two (/boot and /):

* /boot 512 MB
* / The entire disk

d-i partman-auto/expert\_recipe string \

conf-minidisk :: \

512 512 512 ext4 \

$primary{ } $bootable { } \

method{ format } format{ } \

use\_filesystem{ } filesystem{ ext4 } \

mountpoint{ /boot } \

. \

1024 2048 -1 ext4 \

method{ format } format{ } \

use\_filesystem{ } filesystem{ ext4 } \

mountpoint{ / } \

.

Create the partitions without any confirmation dialog

d-i partman-partitioning/confirm\_write\_new\_label boolean true

d-i partman/choose\_partition select finish

d-i partman/confirm boolean true

d-i partman/confirm\_nooverwrite boolean true

#

d-i partman-md/confirm boolean true

d-i partman-partitioning/confirm\_write\_new\_label boolean true

d-i partman/choose\_partition select finish

d-i partman/confirm boolean true

d-i partman/confirm\_nooverwrite boolean true

## Apt setup – repositories and security update

Enable the main repositories (non-free, contrib) and set the mirror for security updates

d-i apt-setup/non-free boolean true

d-i apt-setup/contrib boolean true

d-i apt-setup/use\_mirror boolean true

d-i apt-setup/services-select multiselect security, updates

d-i apt-setup/security\_host string security.debian.org

## Package selection

It is a program who provides a simple interface for users who want to configure their system to perform a specific task or tasks combination. The packages selected to unattended installation are Standard, Standard system utilities, File server, SSH server.

tasksel tasksel/first multiselect Standard, Standard system utilities, File server, SSH server

Also theare are individual packages selected

d-i pkgsel/include string \

openssh-server \

build-essential \

vim \

ntp \

ntpdate \

automake \

sudo \

expect \

chkconfig \

dstat \

fam \

hdparm \

iotop \

nfswatch \

quota \

quotatool \

sysstat \

lynx \

arpwatch \

iptraf-ng \

tcpdump \

iptables \

libpcre3-dev \

libtool \

libltdl-dev \

iftop \

saidar \

sendmail-cf \

sendmail-base \

ldap-utils \

dnstop \

ethtool \

nmap \

snmp \

snmpd \

nmon \

gpm \

linuxlogo \

mc \

nfs-common \

less \

console-data \

console-common \

rsync \

source-highlight \

aptitude \

host \

open-vm-tools \

ansible \

debconf-utils \

cloud-init \

cloud-initramfs-growroot \

cloud-utils

Full upgrade is selected after deboostrap

d-i pkgsel/upgrade select full-upgrade

Do not select the popularity contests

popularity-contest popularity-contest/participate boolean false

## Boot loader installation

This is fairly safe to set, it makes grub install automatically to the MBR if no other operating system is detected on the machine.

d-i grub-installer/only\_debian boolean true

This one makes grub-installer install to the MBR if it also finds some other OS, which is less safe as it might not be able to boot that other OS.

d-i grub-installer/with\_other\_os boolean true

Install the MBR to the first device (assuming it is not a USB stick):

d-i grub-installer/bootdev string default

## Finishing up the installation

Configure the poweroff after install. This is how to make the installer shutdown when finished, but not reboot into the installed system.

d-i debian-installer/exit/poweroff boolean true

## Postinstall

It start with d-i preseed/late\_command string and do not use comments.

This section:

* Change Locale en\_US.UTF-8 into the locale file at installation time (/target/etc/default/locale)
* Enables Vim to show parts of the text in another font or color (using syntax).
* Get the rc.local file from the webserver.
* Get the cloud.cfg file from the webserver
* Set other things into /etc/profile.d to change some Application enviroment to improve console behavior (vim, dmesg, cp, mv, rm, ls and less)
* Include debian user into the sudoers profile (It only works if the cloud-init does not work in an cloud environment).
* Comment the “deb cdrom” entry into the /etc/apt/sources.list in cas of you do not use a netinstall iso file.

d-i preseed/late\_command string \

echo "LC\_ALL=\"en\_US.UTF-8\"" >> /target/etc/default/locale; \

sed -r -i 's/\"syntax\ on/syntax\ on/g' /target/etc/vim/vimrc; \

wget -O /target/etc/rc.local.openstack http://192.168.1.11/recipes/linux/unattended/supportfiles/debian8/rc.local.openstack; \

cat /target/etc/rc.local > /target/etc/rc.local.original; \

cat /target/etc/rc.local.openstack > /target/etc/rc.local; \

wget -O /target/etc/cloud/cloud.cfg.new http://192.168.1.11/recipes/linux/unattended/supportfiles/debian8/cloud.cfg; \

cat /target/etc/cloud/cloud.cfg > /target/etc/cloud/cloud.cfg.original; \

cat /target/etc/cloud/cloud.cfg.new > /target/etc/cloud/cloud.cfg; \

echo "export EDITOR=\"vim\"" > /target/etc/profile.d/extra-profile.sh; \

echo "alias dmesg='dmesg -T'" >> /target/etc/profile.d/extra-profile.sh; \

echo "alias cp='cp -i'" >> /target/etc/profile.d/extra-profile.sh; \

echo "alias mv='mv -i'" >> /target/etc/profile.d/extra-profile.sh; \

echo "alias rm='rm -i'" >> /target/etc/profile.d/extra-profile.sh; \

echo "export LS\_OPTIONS='--color=auto'" >> /target/etc/profile.d/extra-profile.sh; \

echo "export LESSOPEN=\"| /usr/share/source-highlight/src-hilite-lesspipe.sh %s\"" > /target/etc/profile.d/less-hl.sh; \

echo "export LESS=' -R '" >> /target/etc/profile.d/less-hl.sh; \

echo "eval \"\`dircolors\`\"" >> /target/etc/profile.d/extra-profile.sh; \

echo "alias ls='ls \$LS\_OPTIONS -F'" >> /target/etc/profile.d/extra-profile.sh; \

chmod 755 /target/etc/profile.d/\*.sh; \

echo "debian ALL=(ALL) NOPASSWD:ALL" > /target/etc/sudoers.d/img-debian; \

in-target sed -i -e 's/^deb cdrom:/#deb c**drom:/' /etc/apt/sources.list;**

# cloud FILE

## cloud.cfg

# The top level settings are used as module and system configuration.

# A set of users which may be applied and/or used by various modules

# when a 'default' entry is found it will reference the 'default\_user'

# from the distro configuration specified below

users:

- default

# If this is set, 'root' will not be able to ssh in and they

# will get a message to login instead as the above $user (ubuntu)

disable\_root: true

resize\_rootfs: True

# This will cause the set+update hostname module to not operate (if true)

preserve\_hostname: true

# The modules that run in the 'init' stage

cloud\_init\_modules:

- migrator

- bootcmd

- write-files

- resizefs

- set\_hostname

- update\_hostname

- update\_etc\_hosts

- ca-certs

- rsyslog

- users-groups

- ssh

# The modules that run in the 'config' stage

cloud\_config\_modules:

# Emit the cloud config ready event this can be used by upstart jobs for 'start on cloud-config'.

- emit\_upstart

- mounts

- ssh-import-id

- locale

- set-passwords

- grub-dpkg

- apt-pipelining

- apt-configure

- package-update-upgrade-install

- landscape

- timezone

- puppet

- chef

- salt-minion

- mcollective

- disable-ec2-metadata

- runcmd

- byobu

# The modules that run in the 'final' stage cloud\_final\_modules:

- rightscale\_userdata

- scripts-per-once

- scripts-per-boot

- scripts-per-instance

- scripts-user

- ssh-authkey-fingerprints

- keys-to-console

- phone-home

- final-message

- power-state-change

# System and/or distro specific settings(not accessible to handlers/transforms)

system\_info:

# This will affect which distro class gets used

distro: debian

# Default user name + that default users groups (if added/used)

default\_user:

name: debian

lock\_passwd: True

gecos: Debian

sudo: ["ALL=(ALL) NOPASSWD:ALL"]

shell: /bin/bash

groups: [adm, audio, cdrom, dialout, floppy, video, plugdev, dip]

# Other config here will be given to the distro class and/or path classes

paths:

cloud\_dir: /var/lib/cloud/

templates\_dir: /etc/cloud/templates/

upstart\_dir: /etc/init/

package\_mirrors:

- arches: [default]

failsafe:

primary: http://debian.mirrors.ovh.net/debian/

growpart:

mode: auto

devices: ['/']

ignore\_growroot\_disabled: false

# STARTUP SCRIPT

## rc.local

By default this script try to grow the root partition if cloud-init can not.

/usr/bin/growpart -v /dev/sda 2 -u auto > /dev/null 2>&1

/usr/bin/growpart -v /dev/sda 5 -u auto > /dev/null 2>&1

resize2fs /dev/sda5 > /dev/null 2>&1

If application linux\_logo ins installed. The script modify create /etc/issue.net and set the linux\_logo and set the message of the day file (motd) and insert the linux logo, a custom message and the IP Address.

if [ -f /usr/bin/linux\_logo ]; then

OS\_DISTRIBUTIONID=`lsb\_release -i | awk {'print $3'}`

OS\_VERSION=`lsb\_release -r | awk {'print $2'}`

IP\_ADDRESS=`ifconfig eth0 |grep inet |awk -F: {'print $2'}| awk {'print $1'} |head -1`

echo "" > /etc/issue

/usr/bin/linux\_logo -L debian -t "$R" >> /etc/issue

/usr/bin/linux\_logo -L debian -a -t "$R" > /etc/issue.net

echo >> /etc/issue

cat /etc/issue > /etc/motd

echo "" >> /etc/motd

echo "$OS\_DISTRIBUTIONID $OS\_VERSION Server Edition for OpenStack" >> /etc/motd

echo "IP Address: $IP\_ADDRESS" >> /etc/motd

echo "" >> /etc/motd

fi

If there is a devices for swap and cloud-init can does not mount, this script try to do it.

blkid |grep "swap"|cut -d: -f1|xargs swapon > /dev/null 2>&1

exit 0

All the scripts are provided with this documentation.