

This document describes the installation of a CentOS 7.1 server. The purpose of this guide is to provide a minimal setup that can be used as basis for our other tutorials here at howtoforge like the perfect server guides or the SAMBA , LAMP and LEMP server tutorials.

## Requirements

To get started with the CentOS 7.1 installation, we will need the installer ISO file. This can either be the CentOS minimal ISO or the DVD ISO file. If you plan to install just this one server then choose the minimal ISO as it is a smaller, the installer will download the required packages during installation later. I will install several servers with CentOS 7.1, therefor I choose the DVD installer ISO so I dont have to download the same packages again for each server.

- Download the CentOS 7.1 DVD from a mirror next to you <http://www.centos.org/download/>. I will use this mirror: [http://ftp.hosteurope.de/mirror/centos.org/7.1.1503/isos/x86\\_64/CentOS-7-x86\\_64-DVD-1503-01.iso](http://ftp.hosteurope.de/mirror/centos.org/7.1.1503/isos/x86_64/CentOS-7-x86_64-DVD-1503-01.iso)

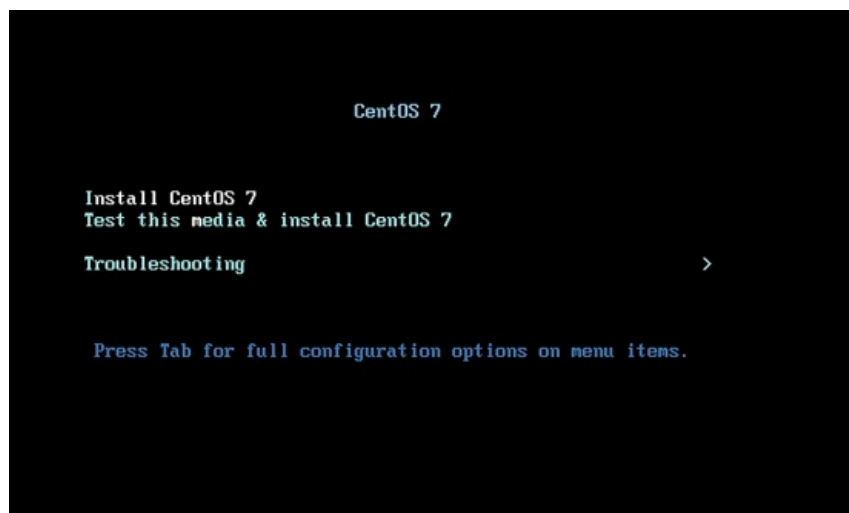
I will do the installation on a vmware virtual machine. The installation steps on a physical server are the same. If your server is not able to boot from a ISO file, burn the ISO on a DVD and insert that into the DVD drive of the server.

## Preliminary Note

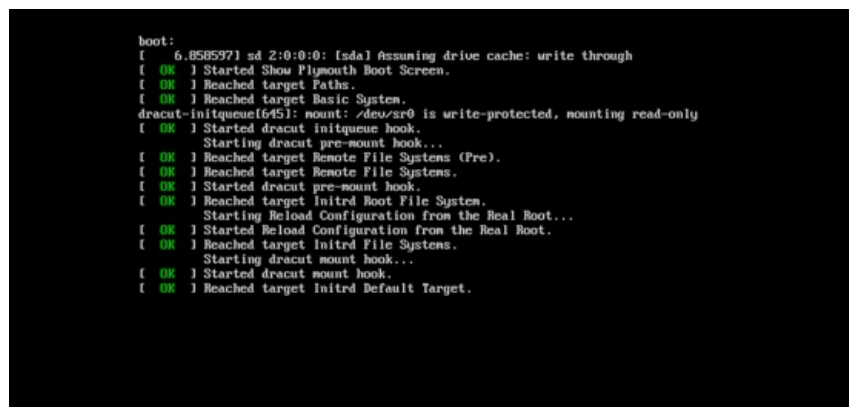
This tutorial is based on CentOS 7.1 server, I use `192.168.0.100` as my IP address in this tutorial and `server1.example.com` as the hostname. These settings might differ for you, so you have to replace them where appropriate.

## Install the Base System

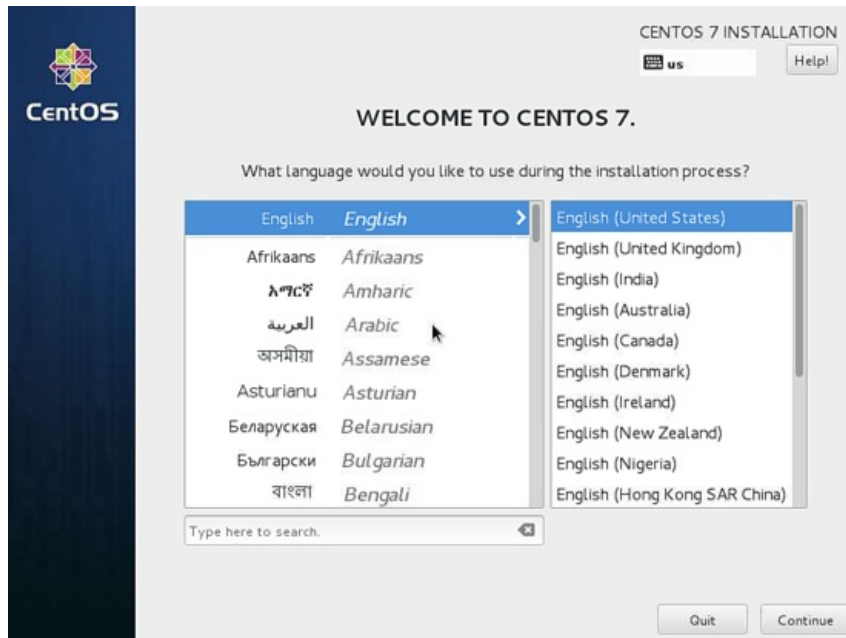
Boot from your CentOS 7 DVD. Select *Install CentOS 7.1*



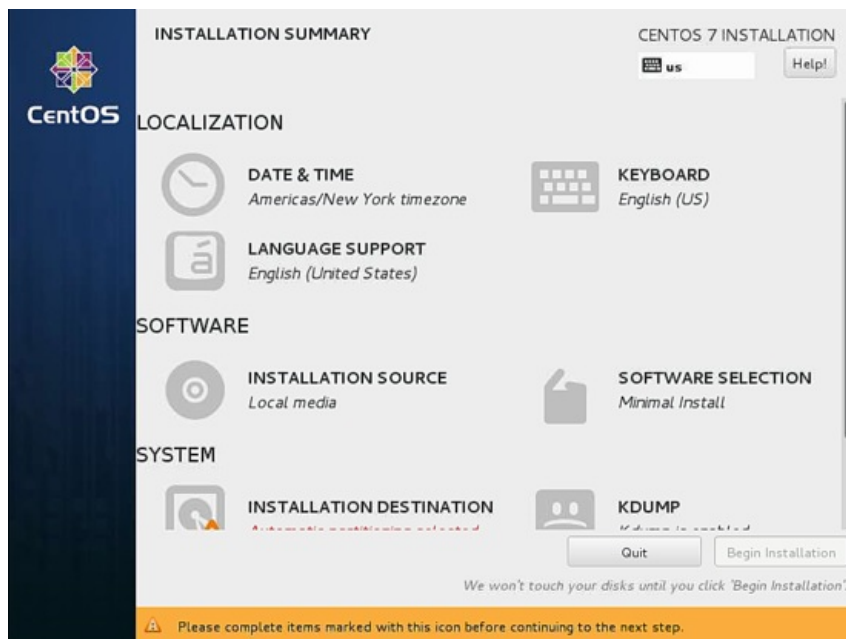
Next press *ENTER*



Next, you can customize the CentOS 7.1 installation setup-language. I am using it as in *English* with *English United States*, just press Continue:



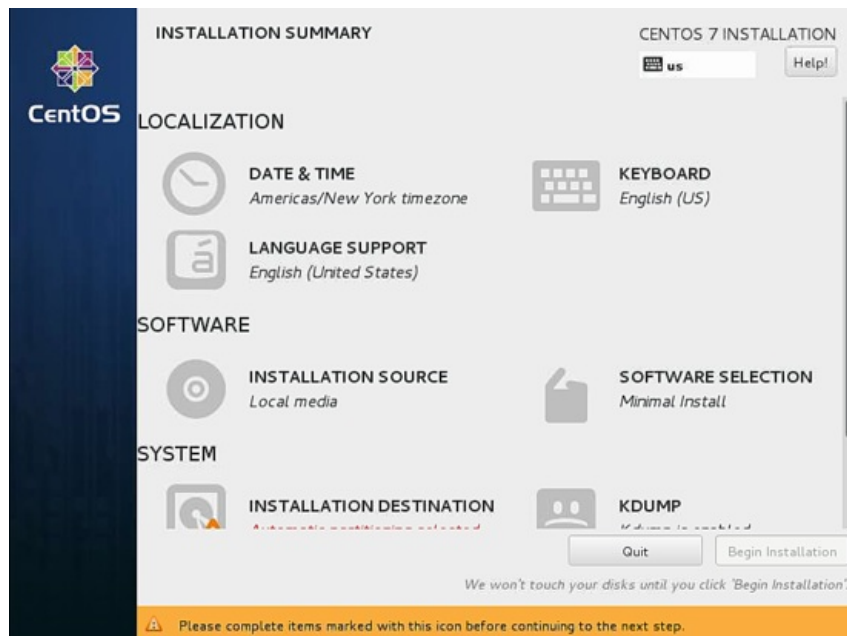
Next we will get the following screen



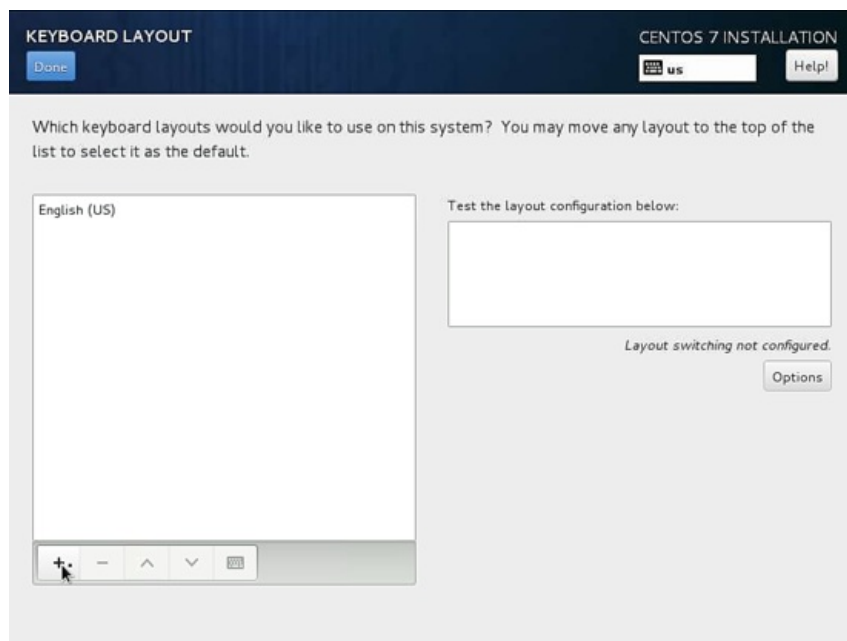
We will start to customize the settings starting with *LOCALIZATION* in *DATE & TIME*. Click on *DATE & TIME*. Now Select your timezone, in my case I am selecting *Region* as *Europe* and *City* as *Berlin*. Press *Done* after finish.



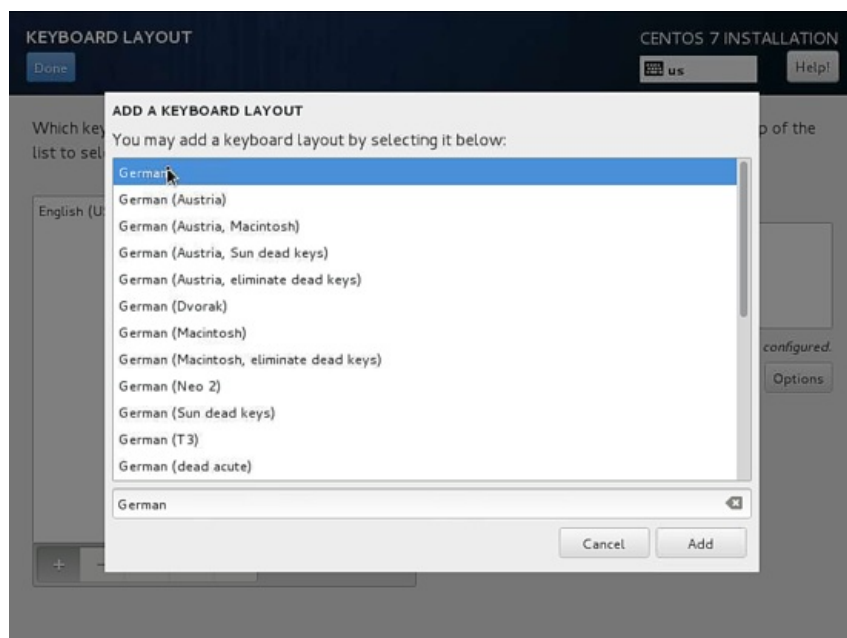
It will make the server *DATE & TIME* as *Europe/Berlin* *timezone*. Next we will customize our *KEYBOARD* press over that.



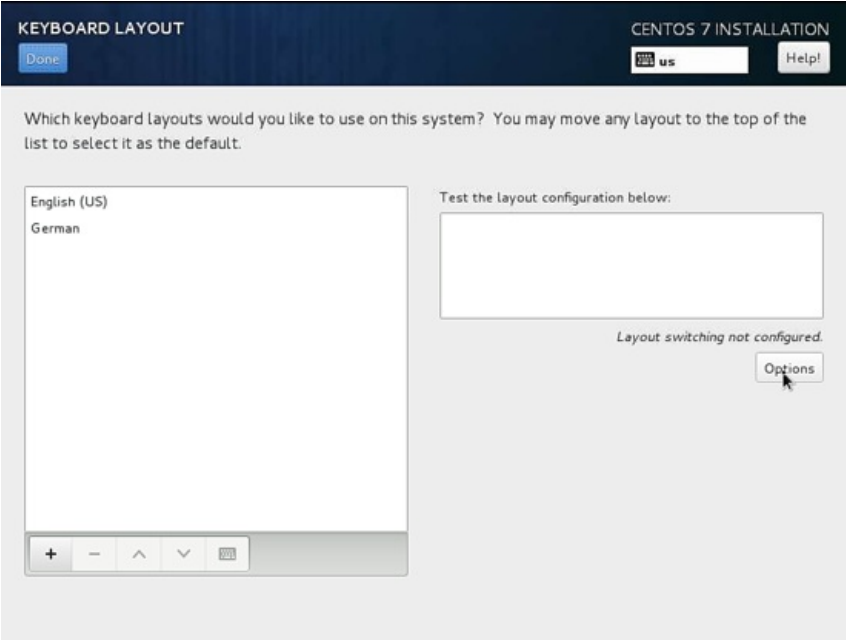
Next it will show the following screen, to add more keyboard layout press + icon



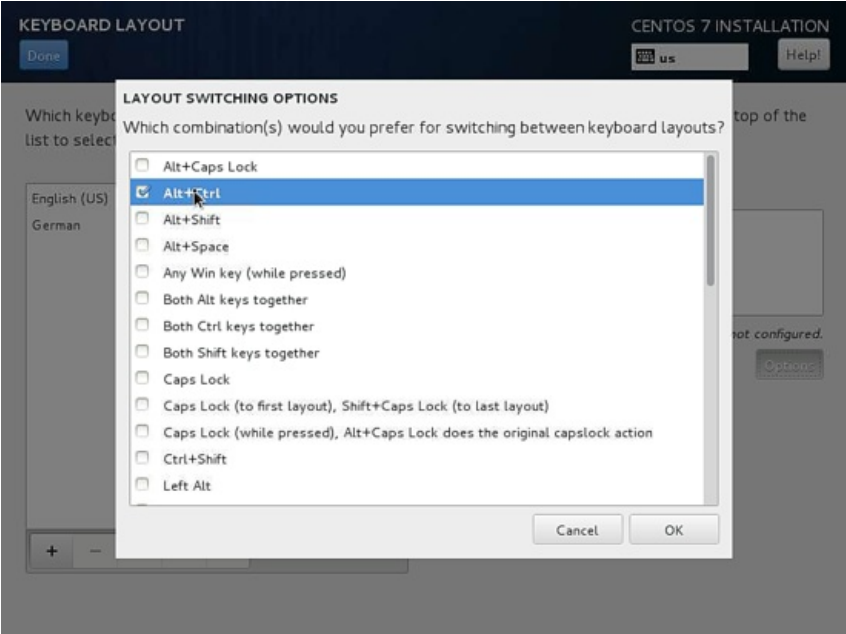
It will show the following window, just add more languages as you need. In my case I am adding *German*, further press *Add*.



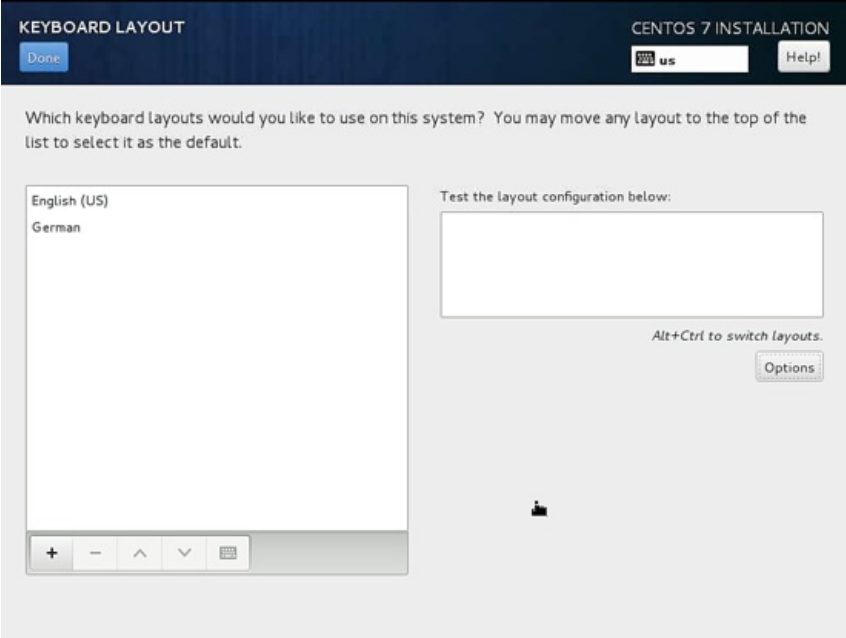
Next we can customize the *LAYOUT SWITCHING OPTIONS* by pressing *Options*:



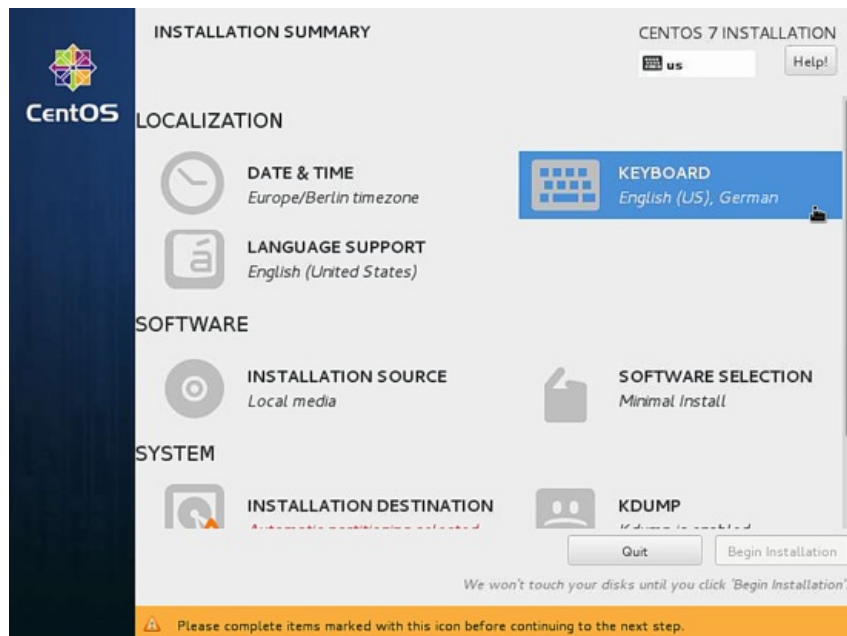
Next you can use any key combination for switching between the keyboards, in my case I am using *Alt+Ctrl*. Further after selection press *Done*



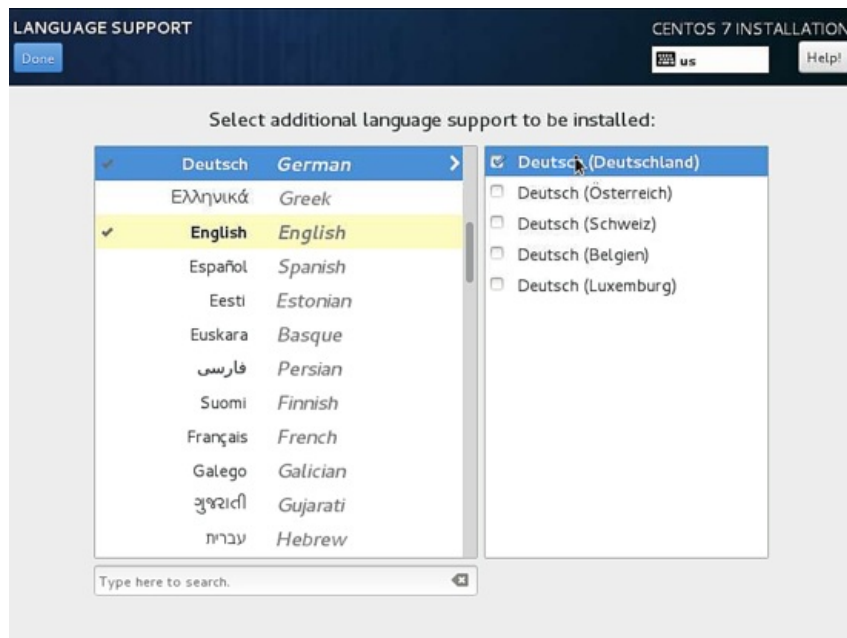
Next press *Done*



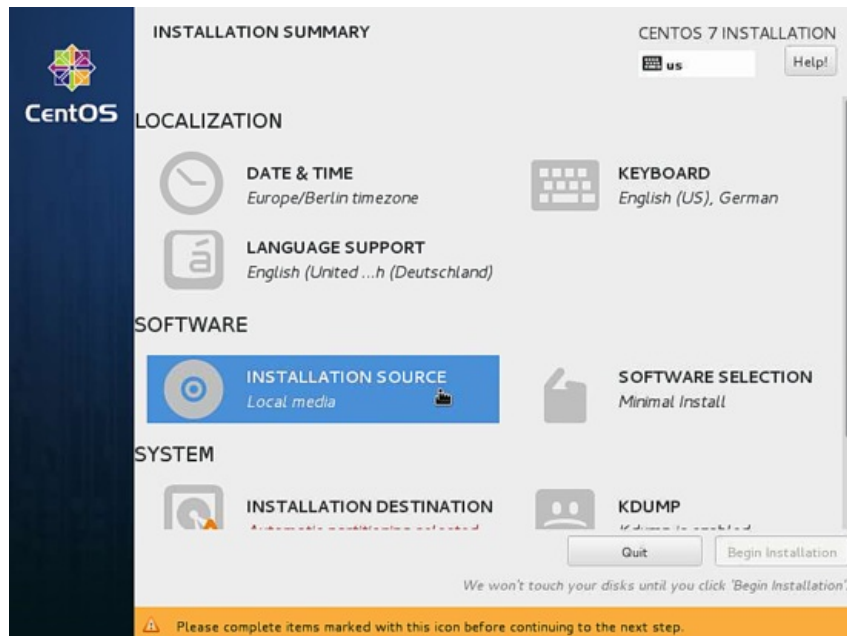
Next we will add *LANGUAGE SUPPORT* by selecting it.



By default CentOS comes with *English*, we can add more language support. Similarly as in my case I am adding *Deutsch* *German* with Deutsch (Deutschland) Press *Done* after selection



Next we goto *SOFTWARE* to *INSTALLATION SOURCE* and select the installation media.



Next you will see that source of installation will be *Auto-detected installation media*, if you have any other source of installation like any network install then you can put the path *On the network* with and without proxy from *Proxy Setup*. Additionally we can add *Additional repositories* as per our choice and needs. After press *Done*.

INSTALLATION SOURCE

CENTOS 7 INSTALLATION

Done

us

Help!

Which installation source would you like to use?

Auto-detected installation media:

Device: sr0

Label: CentOS\_7\_x86\_64

Verify

On the network:

http://

Proxy setup...

This URL refers to a mirror list.

Additional repositories

Enabled

Name

+

-

Name:

http://

This URL refers to a mirror list.

Proxy URL:

User Name:

Password:

You need to configure the network to use a network installation source.

Next we will proceed for *SOFTWARE SELECTION*, by default it comes with *Minimal Install*.

CentOS

INSTALLATION SUMMARY

CENTOS 7 INSTALLATION

us

Help!

LANGUAGE SUPPORT

English (United ...h (Deutschland)

SOFTWARE

INSTALLATION SOURCE

Local media

SOFTWARE SELECTION

Minimal Install

SYSTEM

INSTALLATION DESTINATION

Automatic partitioning selected

KDUMP

Kdump is enabled

NETWORK & HOST NAME

Not connected

Quit

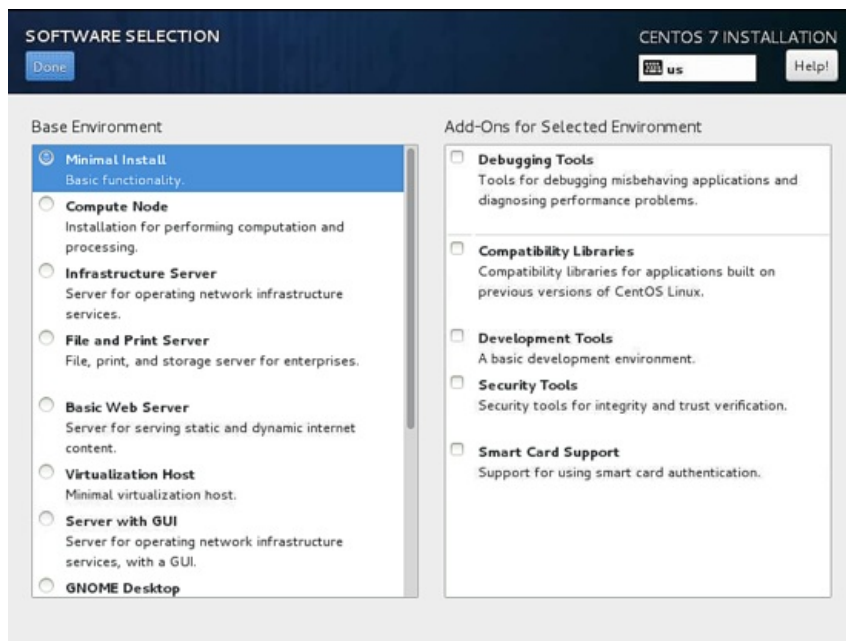
Begin Installation

We won't touch your disks until you click 'Begin Installation'.

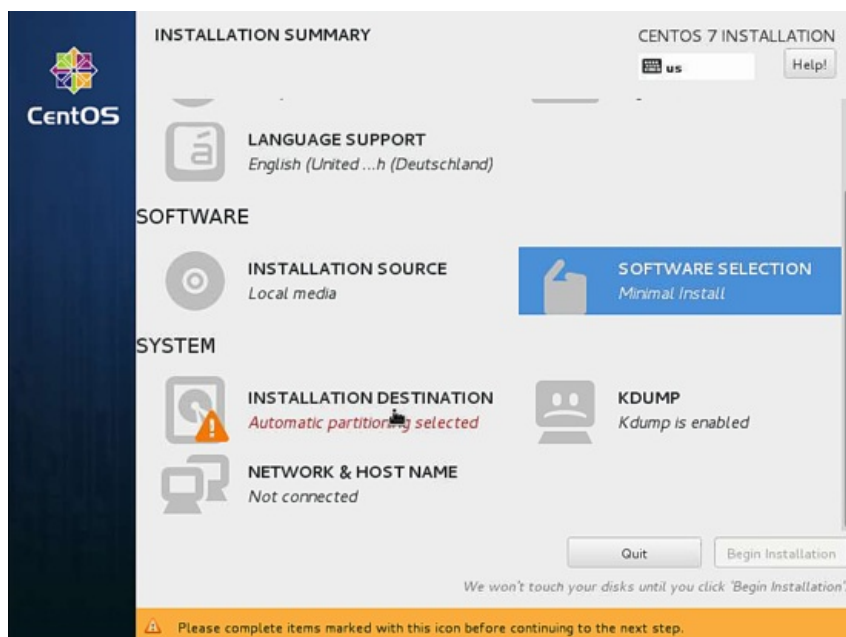
Please complete items marked with this icon before continuing to the next step.

Next we can add more if required, but in my case I need only *Minimal Install(Basic Functionality)* so I will save the settings by pressing *Done*.

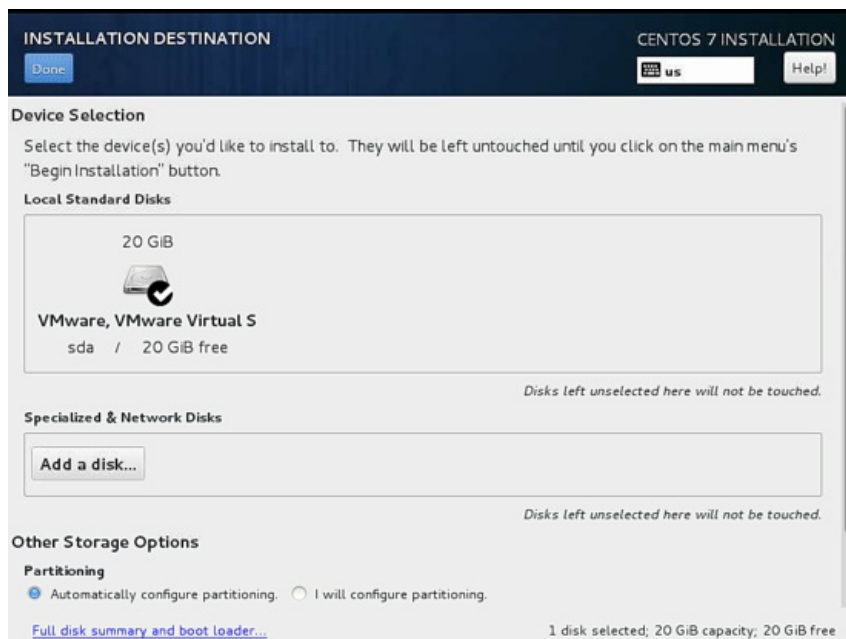




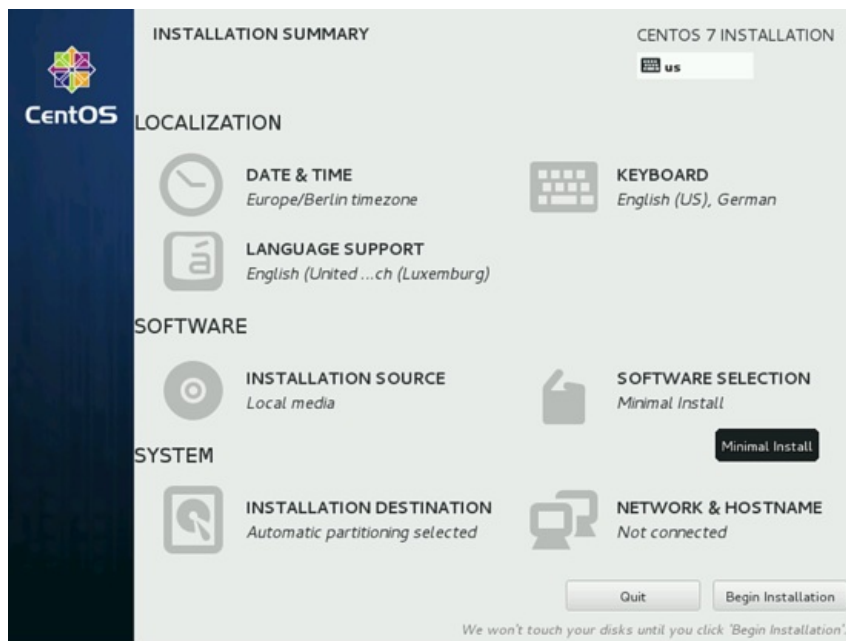
Next we will move towards *SYSTEM* and then to *INSTALLATION DESTINATION*.



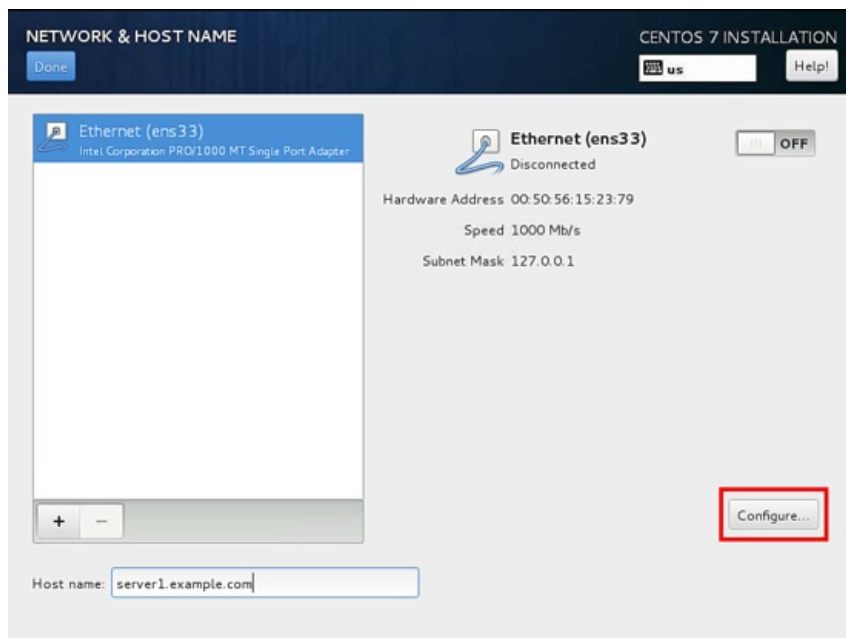
Next if you need a manual partitioning scheme then you can select *I will configure partitioning*, in my case I will prefer *Automatically configure partitioning* after that press *Done*.



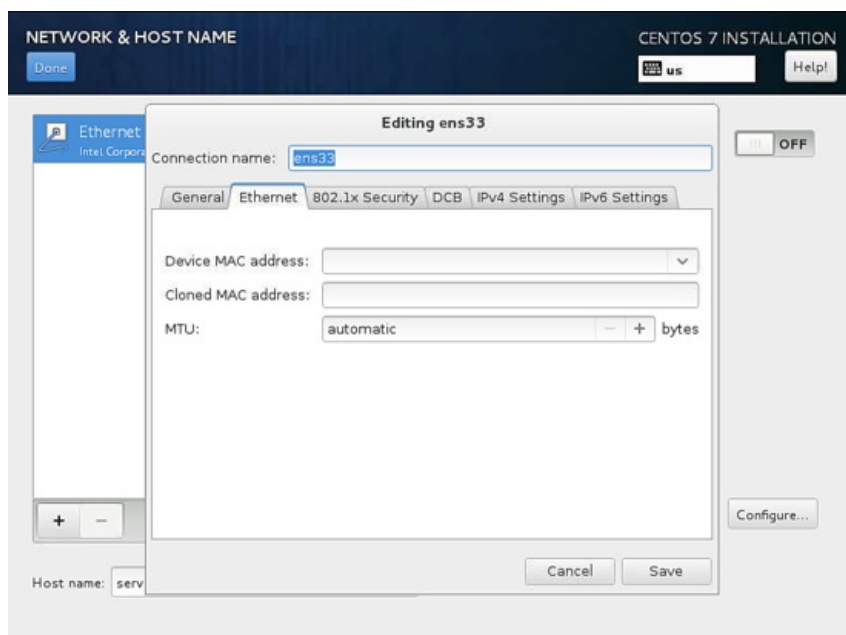
Next we will customize our *NETWORK & HOSTNAME* by selecting that.



Further we will add the hostname as `server1.example.com` and then we will customize our ethernet part by pressing *Configure*.

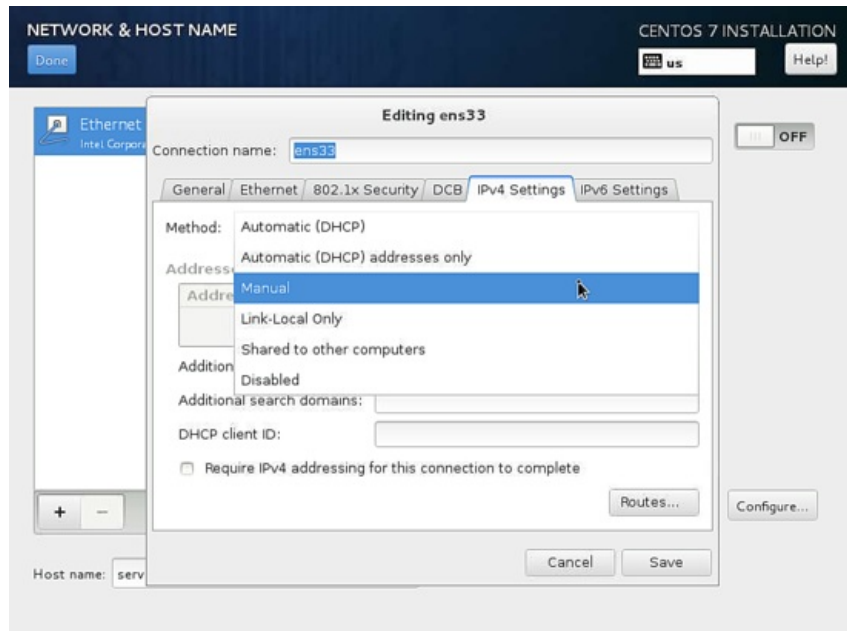


Next screen will be like that, now we want to add *IPv4 Settings* by selecting the same. If you have IPv6 then you add it in *IPv6 Settings*.

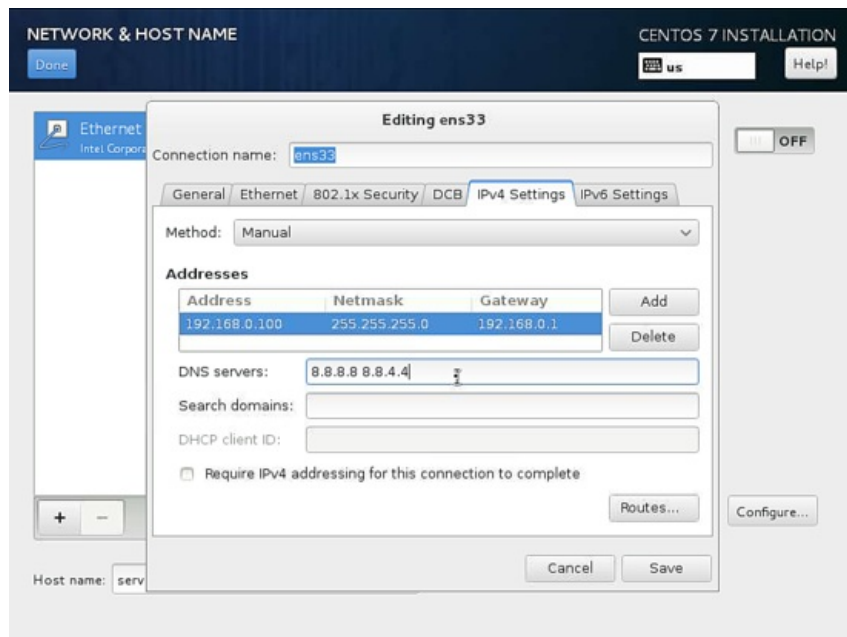




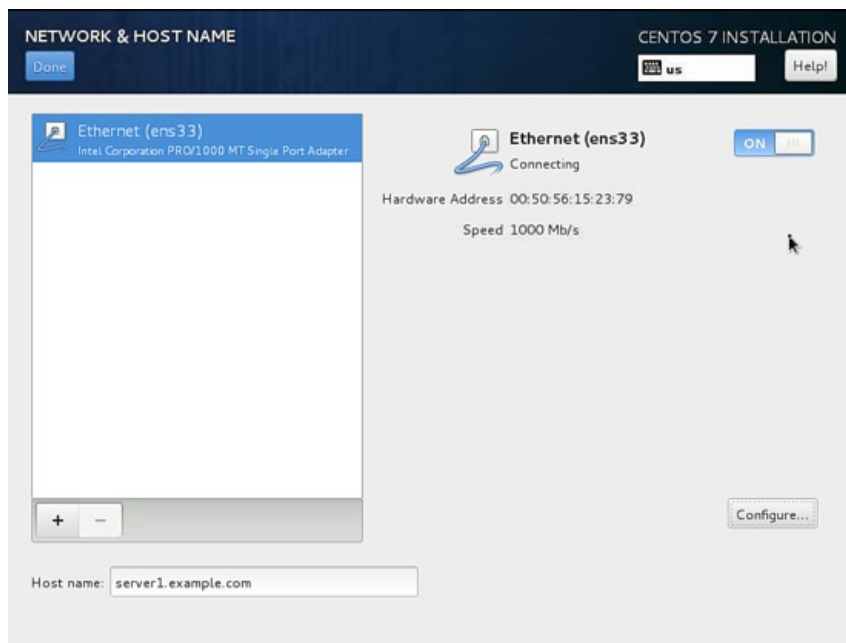
Next you can manage the network as per your LAN/WAN connection in my case I am using static IP so I will select *Manual*



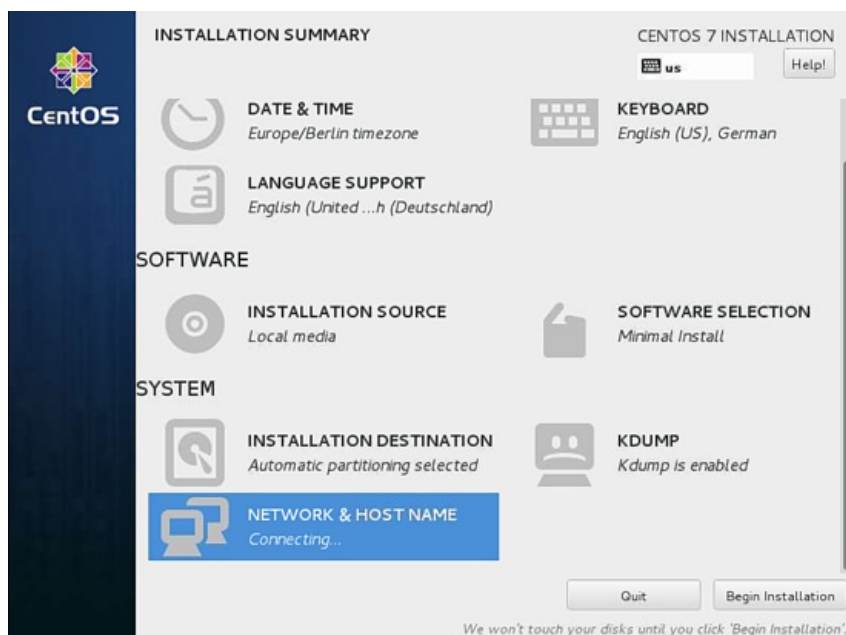
Next add the entries as per your static IP environment, in my case I am using *Address* as *192.168.0.100*, *Netmask* as *255.255.255.0*, *Gateway* as *192.168.0.1* and *DNS servers* as *8.8.8.8 8.8.4.4* These values may vary according to your network environment. After that press *Save*.



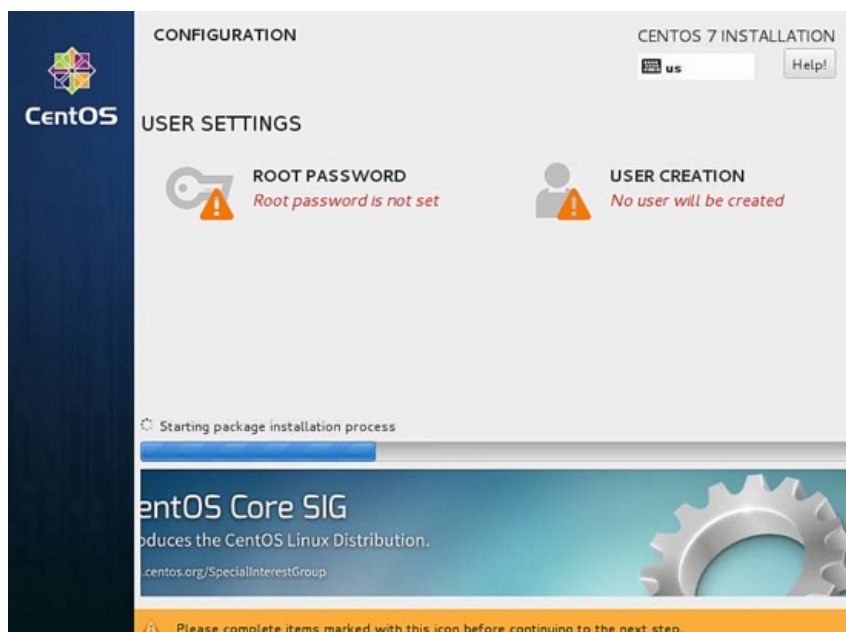
Next we need to make the connection *ON* as shown in screenshot below. It will show the details as per settled by you previously. Further press *Done*.



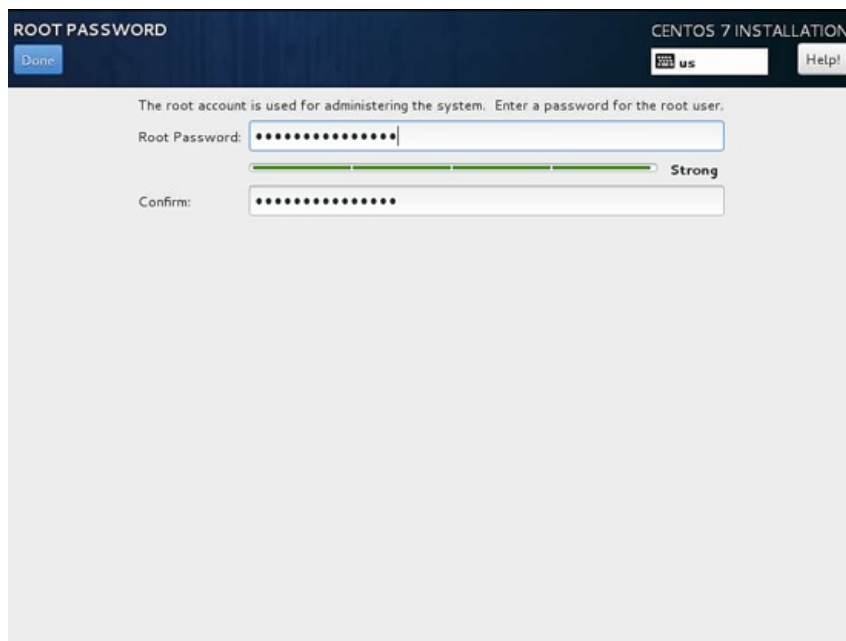
Now we are ready for the installation process, you can crosscheck all the settings again and then press *Begin Installation*.



It will start the process of installation. Now we need to provide *USER SETTINGS* with *ROOT PASSWORD* and *USER CREATION*. I will first go for root password.



Next Enter any password of your choice and press *Done*



ROOT PASSWORD

CENTOS 7 INSTALLATION

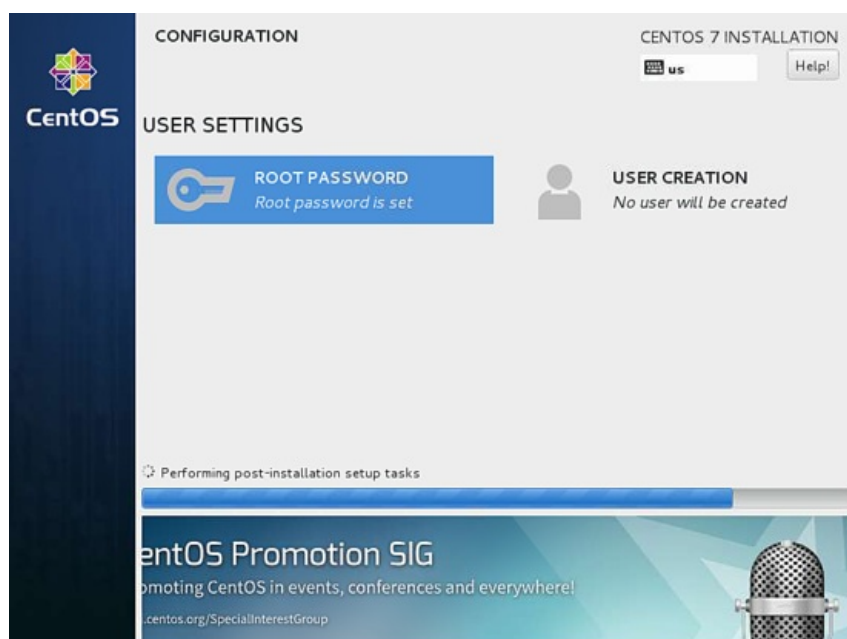
Done us Help!

The root account is used for administering the system. Enter a password for the root user.

Root Password: [password field] Strong

Confirm: [password field]

Next we will go for *USER CREATION*.



CONFIGURATION

CENTOS 7 INSTALLATION

Done us Help!

CentOS

USER SETTINGS


ROOT PASSWORD  
Root password is set

USER CREATION  
No user will be created

Performing post-installation setup tasks

CentOS Promotion SIG  
Promoting CentOS in events, conferences and everywhere!  
[centos.org/SpecialInterestGroup](http://centos.org/SpecialInterestGroup)

Next I will create user, as in my case I used *Full name* as *Administrator*, *Username* as *administrator*, check the option *Require the password to use this account* and then press *Done*. Off-course you can use any value as per your choice.



CREATE USER

CENTOS 7 INSTALLATION

Done us Help!

Full name: Administrator

User name: administrator

Tip: Keep your user name shorter than 32 characters and do not use spaces.

☐ Make this user administrator

☒ Require a password to use this account

Password: [password field] Strong

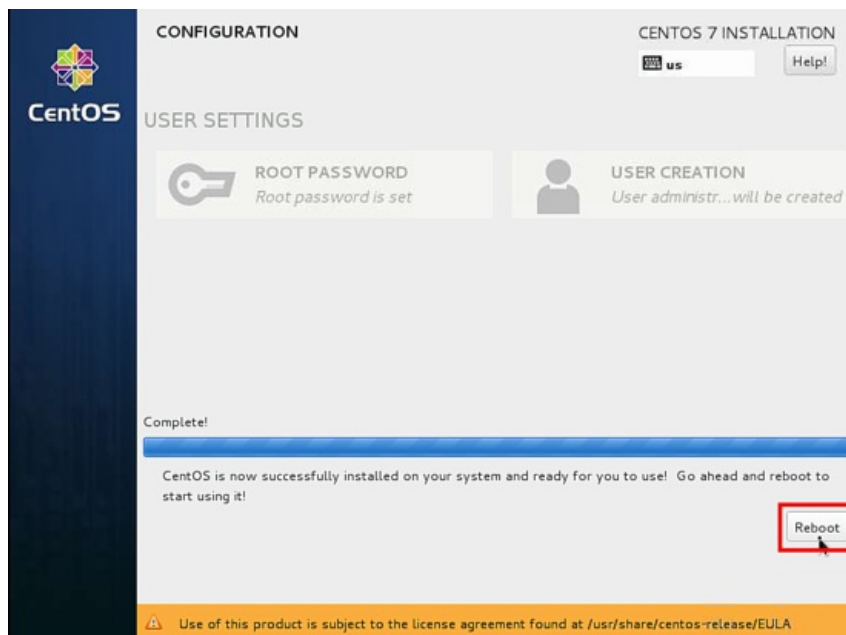
Confirm password: [password field]

Advanced...

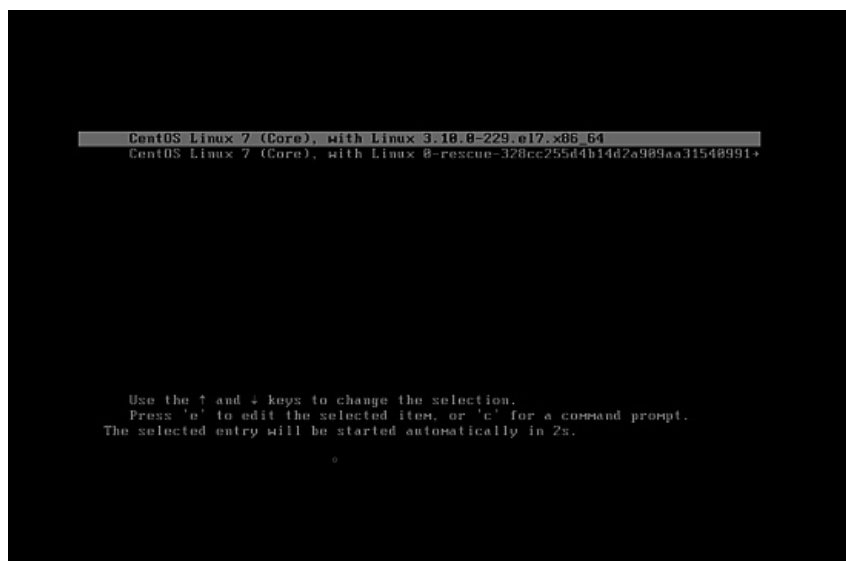
Further have patience and wait for the completion of the setup.



After completion of the installation, it will ask for reboot, just press *Reboot*.



Next It will boot and your login screen will be onscreen.



```
CentOS Linux 7 (Core)  
Kernel 3.10.0-229.el7.x86_64 on an x86_64  
  
server1 login: _
```

Now we are ready to do login with the users just created above or we can use *root* credentials.

## Network Configuration

First I will install two commandline editors to be able to edit configuration files on the shell:

```
yum install nano vim
```

If you want to change or see the network configuration file, just edit the file

```
nano /etc/sysconfig/network-scripts/ifcfg-ens33
```

It will be like this when you configured a static IP address:

```
TYPE="Ethernet"  
BOOTPROTO="none"  
DEFROUTE="yes"  
IPV4_FAILURE_FATAL="no"  
IPV6INIT="yes"  
IPV6_AUTOCONF="yes"  
IPV6_DEFROUTE="yes"  
IPV6_FAILURE_FATAL="no"  
NAME="ens33"  
UUID="acbc0204-057f-4eaa-9c6e-343d207ac403"  
ONBOOT="yes"  
DNS1="8.8.8.8"  
DNS2="8.8.4.4"  
DOMAIN="example.com"  
HWADDR=00:0C:29:A5:A5:31  
IPADDR=192.168.0.100  
PREFIX=24  
GATEWAY=192.168.1.1  
IPV6_PEERDNS=yes  
IPV6_PEERROUTES=yes
```

Or like this when you use DHCP:

```
TYPE="Ethernet"  
BOOTPROTO="dhcp"  
DEFROUTE="yes"  
IPV4_FAILURE_FATAL="no"  
IPV6INIT="yes"  
IPV6_AUTOCONF="yes"  
IPV6_DEFROUTE="yes"  
IPV6_FAILURE_FATAL="no"  
NAME="ens33"  
UUID="b0842d73-974a-4305-90af-568db9b39cce"  
DEVICE="ens33"  
ONBOOT="yes"  
HWADDR="00:50:56:15:23:79"  
PEERDNS="yes"  
PEERROUTES="yes"  
IPV6_PEERDNS="yes"  
IPV6_PEERROUTES="yes"  
IPV6_PRIVACY="no"
```

Change the values if required.

[Note: The above DEVICE name may vary so please check the equivalent file in the directory /etc/sysconfig/network-scripts.](#)

CentOS 7.1 minimal don't come pre-installed with the *ifconfig* command we will install it as follows:

```
yum install net-tools
```

## Adjust /etc/hosts

Now we will login with root credentials and adjust the file /etc/hosts as follows:

```
nano /etc/hosts
```

Make the values like this:

```
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
192.168.0.100  server1.example.com      server1

::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
```

Congratulations! Now we have basic minimal CentOS 7.1 server setup which provides a solid basis for our other tutorials.

## Links

- CentOS : <http://www.centos.org/>



This tutorial shows how to install ISPConfig 3 on a CentOS 7.1 (64Bit) server. ISPConfig 3 is a web hosting control panel that allows you to configure the following services through a web browser: Apache web server, Postfix mail server, MySQL, BIND nameserver, PureFTPd, SpamAssassin, ClamAV, Mailman, and many more. Since version 3.0.4, ISPConfig comes with full support for the nginx web server in addition to Apache; this tutorial covers the setup of a server that uses Apache, not nginx.

## 1 Requirements

To install such a system you will need the following:

- A Centos 7.1 minimal server system. This can be a server installed from scratch as described in our [Centos 7.1 minimal server tutorial](#) or a virtual-server or root-server from a hosting company that has a minimal Centos 7.1 setup installed.
- A fast Internet connection.

## 2 Preliminary Note

In this tutorial I use the hostname *server1.example.com* with the IP address *192.168.1.100* and the gateway *192.168.1.254*. These settings might differ for you, so you have to replace them where appropriate.

## 3 Set the keyboard layout

In case that the keyboard layout of the server does not match your keybord, you can switch to the right keyboard (in my case "de" for a german keyboard layout, with the localectl command:

```
localectl set-keymap de
```

To get a list of all available keymaps, run:

```
localectl list-keymaps
```

I want to install ISPConfig at the end of this tutorial, ISPConfig ships with the Bastille firewall script that Ilike to use as firewall, therefor I disable the default CentOS firewall now. Of course, you are free to leave the CentOS firewall on and configure it to your needs (but then you shouldn't use any other firewall later on as it will most probably interfere with the CentOS firewall).

Run...

```
yum -y install net-tools
systemctl stop firewalld.service
systemctl disable firewalld.service
```

to stop and disable the CentOS firewall.

Then you should check that the firewall has really been disabled. To do so, run the command:

```
iptables -L
```

The output should look like this:

```
[root@server1 ~]# iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
```

Or use the firewall-cmd command:

```
firewall-cmd --state
```

```
[root@server1 ~]# firewall-cmd --state
not running
[root@server1 ~]#
```

Now I will install the network configuration editor and the shell based editor "nano" that I will use in the next steps to edit the config files:

```
yum -y install nano wget NetworkManager-tui
```

If you did not configure your network card during the installation, you can do that now. Run...

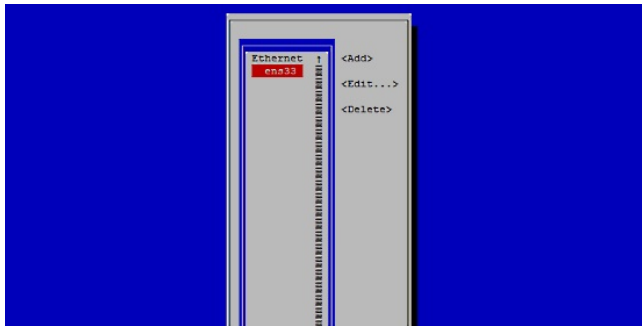
```
nmtui
```

... and go to *Edit a connection*:

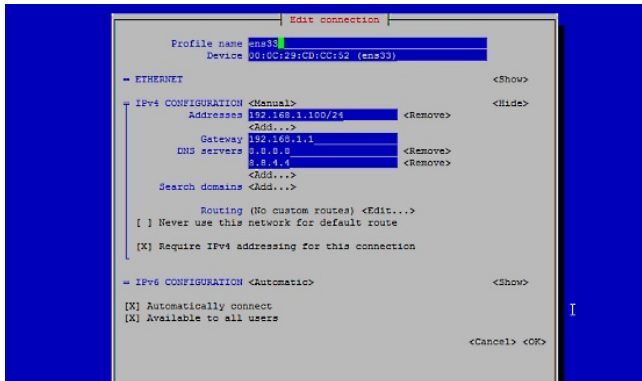


I

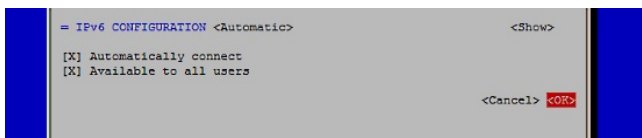
Select your network interface:



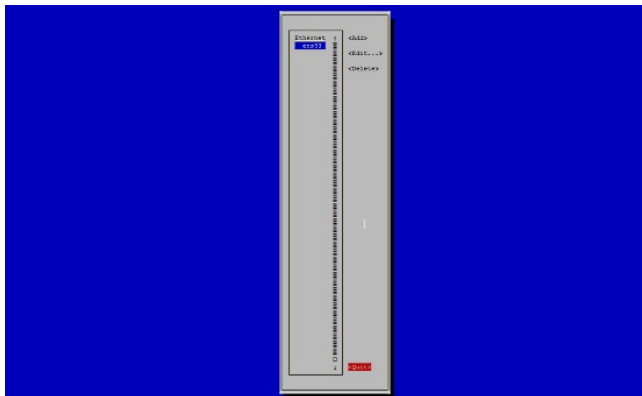
Then fill in your network details - disable DHCP and fill in a static IP address, a netmask, your gateway, and one or two nameservers, then hit *ok*:



Next select *OK* to confirm the changes that you made in the network settings



and *Quit* to close the nmtui network configuration tool.



You should run

```
ifconfig
```

now to check if the installer got your IP address right:

```
[root@server1 ~]# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.100 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::20c:29ff:fed:cc52 prefixlen 64 scopeid 0x20

    ether 00:0c:29:cd:cc:52 txqueuelen 1000 (Ethernet)
    RX packets 55621 bytes 79601094 (75.9 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 28115 bytes 2608239 (2.4 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10
    loop txqueuelen 0 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

If your network card does not show up there, then it not be enabled on boot, In this case, open the file `/etc/sysconfig/network-scripts/ifcfg-eth0`

```
nano /etc/sysconfig/network-scripts/ifcfg-ens33
```

and set ONBOOT to yes:

```
[...]
ONBOOT=yes
[...]
```

and reboot the server.

Check your `/etc/resolv.conf` if it lists all nameservers that you've previously configured:

```
cat /etc/resolv.conf
```

If nameservers are missing, run

```
nmtui
```

and add the missing nameservers again.

Now, on to the configuration...

## 4 Adjust /etc/hosts

Next we edit `/etc/hosts`. Make it look like this:

```
nano /etc/hosts
```

```
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
192.168.1.100 server1.example.com      server1
```

```
::1          localhost localhost.localdomain localhost6 localhost6.localdomain6
```

## 5 Disable SELinux

SELinux is a security extension of CentOS that should provide extended security. In my opinion you don't need it to configure a secure system, and it usually causes more problems than advantages (think of it after you have done a week of trouble-shooting because some service wasn't working as expected, and then you find out that everything was ok, only SELinux was causing the problem). Therefore I disable it (this is a must if you want to install ISPConfig later on).

Edit `/etc/selinux/config` and set `SELINUX=disabled`:

```
nano /etc/selinux/config
```

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Afterwards we must reboot the system:

```
reboot
```

## 6 Enable Additional Repositories And Install Some Software

First we import the GPG keys for software packages:

```
rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY*
```

Then we enable the EPEL repository on our CentOS system as lots of the packages that we are going to install in the course of this tutorial are not available in the official CentOS 7 repository:

```
yum -y install epel-release
```

```
yum -y install yum-priorities
```

Edit `/etc/yum.repos.d/epel.repo`:

```
nano /etc/yum.repos.d/epel.repo
```

... and add the line `priority=10` to the `[epel]` section:

```
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/7/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-7&arch=$basearch
failovermethod=priority
enabled=1
priority=10
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
[...]
```

Then we update our existing packages on the system:

```
yum update
```

Now we install some software packages that are needed later on:

```
yum -y groupinstall 'Development Tools'
```

## 7 Quota

(If you have chosen a different partitioning scheme than I did, you must adjust this chapter so that quota applies to the partitions where you need it.)

To install quota, we run this command:

```
yum -y install quota
```

Now we check if quota is already enabled for the filesystem where the website (/var/www) and maildir data (var/vmail) is stored. In this example setup, I have one big root partition, so I search for ' / ':

```
mount | grep ' / '
```

```
[root@server1 ~]# mount | grep ' / '
/dev/mapper/centos-root on / type xfs (rw,relatime,attr2,inode64,noquota)
[root@server1 ~]#
```

If you have a separate /var partition, then use:

```
mount | grep ' /var '
```

instead. If the line contains the word "**noquota**", then proceed with the following steps to enable quota.

### Enabling quota on the / (root) partition

Normally you would enable quota in the /etc/fstab file, but if the filesystem is the root filesystem "/", then quota has to be enabled by a boot parameter of the Linux Kernel.

Edit the grub configuration file:

```
nano /etc/default/grub
```

search for the line that starts with `GRUB_CMDLINE_LINUX` and add `rootflags=uquota,gquota` to the commandline parameters so that the resulting line looks like this:

```
GRUB_CMDLINE_LINUX="rd.lvm.lv=centos/swap vconsole.font=latarcyrheb-sun16 rd.lvm.lv=centos/root crashkernel=auto vconsole.keymap=us rhgb quiet rootflags=uquota,gquota"
```

and apply the changes by running the following command.

```
cp /boot/grub2/grub.cfg /boot/grub2/grub.cfg_bak
grub2-mkconfig -o /boot/grub2/grub.cfg
```

and reboot the server.

```
reboot
```

Now check if quota is enabled:

```
mount | grep ' / '
```

```
[root@server1 ~]# mount | grep ' / '
/dev/mapper/centos-root on / type xfs (rw,relatime,attr2,inode64,usrquota,grpquota)
[root@server1 ~]#
```

When quota is active, we can see "**usrquota,grpquota**" in the mount option list.

### Enabling quota on a separate /var partition

If you have a separate /var partition, then edit /etc/fstab and add `,uquota,gquota` to the / partition (/dev/mapper/centos-var):

```
nano /etc/fstab
```

```
#
# /etc/fstab
# Created by anaconda on Sun Sep 21 16:33:45 2014
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/centos-root / xfs defaults 1 1
/dev/mapper/centos-var /var xfs defaults,uquota,gquota 1 2
UUID=9ac06939-7e43-4efd-957a-486775edd7b4 /boot xfs defaults 1 3
/dev/mapper/centos-swap swap swap defaults 0 0
```

Then run

```
mount -o remount /var
```

```
quotacheck -avugm
quotaon -avug
```

to enable quota. When you get a error that there is no oartition with quota enabled, then reboot the server before you proceed.

## 8 Install Apache, MySQL, phpMyAdmin

We can install the needed packages with one single command:

```
yum -y install ntp httpd mod_ssl mariadb-server php php-mysql php-mbstring phpmyadmin
```

## 9 Install Dovecot

Dovecot can be installed as follows:

```
yum -y install dovecot dovecot-mysql dovecot-pigeonhole
```

Create a empty dovecot-sql.conf file and symlink:

```
touch /etc/dovecot/dovecot-sql.conf
ln -s /etc/dovecot/dovecot-sql.conf /etc/dovecot-sql.conf
```

Now create the system startup links and start Dovecot:

```
systemctl enable dovecot
systemctl start dovecot
```

## 10 Install Postfix

Postfix can be installed as follows:

```
yum -y install postfix
```

Then turn off Sendmail and start Postfix and Mariadb (MySQL):

```
systemctl enable mariadb.service
systemctl start mariadb.service
```

```
systemctl stop sendmail.service
systemctl disable sendmail.service
systemctl enable postfix.service
systemctl restart postfix.service
```

We disable sendmail to ensure that it does not get started in case it is installed on your server. So the error message "Failed to issue method call: Unit sendmail.service not loaded." can be ignored.

## 11 Install Getmail

Getmail can be installed as follows:

```
yum -y install getmail
```

## 12 Set MySQL Passwords And Configure phpMyAdmin

Set passwords for the MySQL root account:

```
mysql_secure_installation
```

```
[root@server1 tmp]# mysql_secure_installation
```

*NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!*

*In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
you haven't set the root password yet, the password will be blank,  
so you should just press enter here.*

*Enter current password for root (enter for none):  
OK, successfully used password, moving on...*

*Setting the root password ensures that nobody can log into the MariaDB  
root user without the proper authorisation.*

*Set root password? [Y/n] <-- ENTER  
New password: <-- yourrootsqlpassword  
Re-enter new password: <-- yourrootsqlpassword  
Password updated successfully!  
Reloading privilege tables..  
... Success!*

*By default, a MariaDB installation has an anonymous user, allowing anyone  
to log into MariaDB without having to have a user account created for  
them. This is intended only for testing, and to make the installation  
go a bit smoother. You should remove them before moving into a  
production environment.*

*Remove anonymous users? [Y/n] <-- ENTER  
... Success!*

*Normally, root should only be allowed to connect from 'localhost'. This  
ensures that someone cannot guess at the root password from the network.*

*Disallow root login remotely? [Y/n] <-- ENTER  
... Success!*

*By default, MariaDB comes with a database named 'test' that anyone can  
access. This is also intended only for testing, and should be removed  
before moving into a production environment.*

*Remove test database and access to it? [Y/n] <-- ENTER  
- Dropping test database..  
... Success!  
- Removing privileges on test database..  
... Success!*

*Reloading the privilege tables will ensure that all changes made so far  
will take effect immediately.*

*Reload privilege tables now? [Y/n] <-- ENTER*



```
... Success!
```

```
Cleaning up...
```

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!

```
[root@server1 tmp]#
```

Now we configure phpMyAdmin. We change the Apache configuration so that phpMyAdmin allows connections not just from localhost (by commenting out the two "Require ip" lines and adding the new line "Require all granted" in the `<Directory /usr/share/phpMyAdmin/>` stanza):

```
nano /etc/httpd/conf.d/phpMyAdmin.conf

# phpMyAdmin - Web based MySQL browser written in php
#
# Allows only localhost by default
#
# But allowing phpMyAdmin to anyone other than localhost should be considered
# dangerous unless properly secured by SSL

Alias /phpMyAdmin /usr/share/phpMyAdmin
Alias /phpmyadmin /usr/share/phpMyAdmin

<Directory /usr/share/phpMyAdmin/>
  <IfModule mod_authz_core.c>
    # Apache 2.4
    <RequireAny>
    #   Require ip 127.0.0.1
    #   Require ip ::1
    #   Require all granted
    </RequireAny>
  </IfModule>
  <IfModule !mod_authz_core.c>
    # Apache 2.2
    Order Deny,Allow
    Deny from All
    Allow from 127.0.0.1
    Allow from ::1
  </IfModule>
</Directory>
```

Next we change the authentication in phpMyAdmin from `cookie` to `http`:

```
nano /etc/phpMyAdmin/config.inc.php

[...]
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'http';
[...]
```

Then we create the system startup links for Apache and start it:

```
systemctl enable httpd.service
systemctl restart httpd.service
```

Now you can direct your browser to `http://server1.example.com/phpmyadmin/` or `http://192.168.0.100/phpmyadmin/` and log in with the user name `root` and your new root MySQL password.

## 13 Install Amavisd-new, SpamAssassin And ClamAV

To install amavisd-new, spamassassin and clamav, run the following command:

```
yum -y install amavisd-new spamassassin clamav clamav-update unzip bzip2 perl-DBD-mysql
```

Edit the freshclam configuration file `/etc/freshclam.conf`

```
nano /etc/freshclam.conf
```

and comment out the line "Example"

```
[...]
# Example
[...]
```

Then we start freshclam, amavisd, and clamd.amavisd:

```
sa-update
freshclam
systemctl enable amavisd.service
```

## 14 Installing Apache2 With mod\_php, mod\_fcgi/PHP5, PHP-FPM And suPHP

ISPConfig 3 allows you to use mod\_php, mod\_fcgi/PHP5, cgi/PHP5, and suPHP on a per website basis.

We can install Apache2 with mod\_php5, mod\_fcgid, and PHP5 as follows:

```
yum -y install php php-devel php-gd php-imap php-ldap php-mysql php-odbc php-pear php-xml php-xmlrpc php-pecl-apc php-mbstring php-mcrypt php-mssql php-snmp php-soap php-tidy
curl curl-devel perl-libwww-perl ImageMagick libxml2 libxml2-devel mod_fcgid php-cli httpd-devel php-fpm
```

Next we open `/etc/php.ini...`

```
nano /etc/php.ini
```



... and change the error reporting (so that notices aren't shown any longer), set the timezone and uncomment `cgi.fix_pathinfo=1`:

```
[...]
;error_reporting = E_ALL & ~E_DEPRECATED
error_reporting = E_ALL & ~E_NOTICE & ~E_DEPRECATED
[...]
; cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support for CGI. PHP's
; previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME, and to not grok
; what PATH_INFO is. For more information on Papp.tldTH_INFO, see the cgi specs. Setting
; this to 1 will cause PHP CGI to fix its paths to conform to the spec. A setting
; of zero causes PHP to behave as before. Default is 1. You should fix your scripts
; to use SCRIPT_FILENAME rather than PATH_TRANSLATED.
; http://www.php.net/manual/en/ini.core.php#ini.cgi.fix_pathinfo
cgi.fix_pathinfo=1
[...]
date.timezone = 'Europe/Berlin'
[...]
```

Next we install suPHP (there is a `mod_suphp` package available in the repositories, but unfortunately it isn't compatible with ISPConfig, therefore we have to build suPHP ourselves):

```
cd /usr/local/src
wget http://suphp.org/download/suphp-0.7.2.tar.gz
tar xzvf suphp-0.7.2.tar.gz
```

CentOS 7.1 uses apache-2.4, so we need a patch suphp before we can compile it against Apache. The patch gets applied like this:

```
wget -O suphp.patch https://lists.marsching.com/pipermail/suphp/attachments/20130520/74f3ac02/attachment.patch
patch -Np1 -d suphp-0.7.2 < suphp.patch
cd suphp-0.7.2
autoreconf -if
```

```
[root@server1 suphp-0.7.2]# autoreconf -if
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, `config'.
libtoolize: copying file `config/ltmain.sh'
libtoolize: Consider adding 'AC_CONFIG_MACRO_DIR([m4])' to configure.ac and
libtoolize: rerunning libtoolize, to keep the correct libtool macros in-tree.
libtoolize: Consider adding '-I m4' to ACLOCAL_AMFLAGS in Makefile.am.
configure.ac:9: warning: AM_INIT_AUTOMAKE: two- and three-arguments forms are deprecated. For more info, see:
configure.ac:9: http://www.gnu.org/software/automake/manual/automake.html#Modernize-AM_005fINIT_005fAUTOMAKE-invocation
configure.ac:24: installing `config/config.guess'
configure.ac:24: installing `config/config.sub'
configure.ac:9: installing `config/install-sh'
configure.ac:9: installing `config/missing'
src/Makefile.am: installing `config/depcomp'
[root@server1 suphp-0.7.2]#
```

It will apply the patch, now we can compile the new source as follows:

```
./configure --prefix=/usr/ --sysconfdir=/etc/ --with-apr=/usr/bin/apr-1-config --with-apache-user=apache --with-setid-mode=owner --with-logfile=/var/log/httpd/suphp_log
make
make install
```

Then we add the suPHP module to our Apache configuration...

```
nano /etc/httpd/conf.d/suphp.conf
```

```
LoadModule suphp_module modules/mod_suphp.so
```

... and create the file `/etc/suphp.conf` as follows:

```
nano /etc/suphp.conf
```

```
[global]
;Path to logfile
logfile=/var/log/httpd/suphp.log
;Loglevel
loglevel=info
;User Apache is running as
webserver_user=apache
;Path all scripts have to be in
docroot=/
;Path to chroot() to before executing script
chroot=/mychroot
; Security options
allow_file_group_writeable=true
allow_file_others_writeable=false
allow_directory_group_writeable=true
allow_directory_others_writeable=false
;Check wheter script is within DOCUMENT_ROOT
check_vhost_docroot=true
;Send minor error messages to browser
errors_to_browser=false
;PATH environment variable
env_path=/bin:/usr/bin
;Umask to set, specify in octal notation
umask=0077
; Minimum UID
min_uid=100
; Minimum GID
min_gid=100

[handlers]
;Handler for php-scripts
x-httpd-suphp="php:/usr/bin/php-cgi"
;Handler for CGI-scripts
x-suphp-cgi="execute:!self"
```

Edit the file `/etc/httpd/conf.d/php.conf` to enable php parsing only for phpmyadmin, roundcube and other system packages in `/usr/share` but not for websites in `/var/www` as ISPConfig will activate PHP for each website individually.

```
nano /etc/httpd/conf.d/php.conf
```

change the lines:

```
<FilesMatch \.php$>
SetHandler application/x-httpd-php
```

```
</FilesMatch>
```

to:

```
<Directory /usr/share>
<FilesMatch \.php$>
SetHandler application/x-httpd-php
</FilesMatch>
</Directory>
```

So that the PHP handler is enclosed by the Directory directive.

Enable httpd and PHP-FPM to get started at boot time and start the PHP-FPM service.

```
systemctl start php-fpm.service
systemctl enable php-fpm.service
systemctl enable httpd.service
```

Finally we restart Apache:

```
systemctl restart httpd.service
```

## 15 Installation of mod\_python

The apache module mod\_python is not available as RPM package, therefor we will compile it from source. The first step is to install the python development files and download the current mod\_python version as tar.gz file

```
yum -y install python-devel
```

```
cd /usr/local/src/
wget http://dist.modpython.org/dist/mod_python-3.5.0.tgz
tar xzf mod_python-3.5.0.tgz
cd mod_python-3.5.0
```

and then configure and compile the module

```
./configure
make
make install
```

and enable the module in apache

```
echo 'LoadModule python_module modules/mod_python.so' > /etc/httpd/conf.modules.d/10-python.conf
systemctl restart httpd.service
```

## 16 Install PureFTPd

PureFTPd can be installed with the following command:

```
yum -y install pure-ftpd
```

Then create the system startup links and start PureFTPd:

```
systemctl enable pure-ftpd.service
systemctl start pure-ftpd.service
```

Now we configure PureFTPd to allow FTP and TLS sessions. FTP is a very insecure protocol because all passwords and all data are transferred in clear text. By using TLS, the whole communication can be encrypted, thus making FTP much more secure.

OpenSSL is needed by TLS; to install OpenSSL, we simply run:

```
yum install openssl
```

Open /etc/pure-ftpd/pure-ftpd.conf...

```
nano /etc/pure-ftpd/pure-ftpd.conf
```

If you want to allow FTP and TLS sessions, set *TLS* to *1*:

```
[...]
# This option can accept three values :
# 0 : disable SSL/TLS encryption layer (default).
# 1 : accept both traditional and encrypted sessions.
# 2 : refuse connections that don't use SSL/TLS security mechanisms,
#     including anonymous sessions.
# Do not uncomment this blindly. Be sure that :
# 1) Your server has been compiled with SSL/TLS support (--with-tls),
# 2) A valid certificate is in place,
# 3) Only compatible clients will log in.
```

```
TLS                      1
[...]
```

In order to use TLS, we must create an SSL certificate. I create it in /etc/ssl/private/, therefore I create that directory first:

```
mkdir -p /etc/ssl/private/
```

Afterwards, we can generate the SSL certificate as follows:

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem -out /etc/ssl/private/pure-ftpd.pem
```

Country Name (2 letter code) [XX]: <-- Enter your Country Name (e.g., "DE").  
State or Province Name (full name) []: <-- Enter your State or Province Name.  
Locality Name (eg, city) [Default City]: <-- Enter your City.

Organization Name (eg, company) [Default Company Ltd]: <-- Enter your Organization Name (e.g., the name of your company).  
Organizational Unit Name (eg, section) []: <-- Enter your Organizational Unit Name (e.g. "IT Department").  
Common Name (eg, your name or your server's hostname) []: <-- Enter the Fully Qualified Domain Name of the system (e.g. "server1.example.com").  
Email Address []: <-- Enter your Email Address.

Change the permissions of the SSL certificate:

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

Finally restart PureFTPd:

```
systemctl restart pure-ftpd.service
```

That's it. You can now try to connect using your FTP client; however, you should configure your FTP client to use TLS.

## 17 Install BIND

We can install BIND as follows:

```
yum -y install bind bind-utils
```

Make a backup of the existing `/etc/named.conf` file and create a new one as follows:

```
cp /etc/named.conf /etc/named.conf_bak  
cat /dev/null > /etc/named.conf  
nano /etc/named.conf
```

```
//  
// named.conf  
//  
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS  
// server as a caching only nameserver (as a localhost DNS resolver only).  
//  
// See /usr/share/doc/bind*/sample/ for example named configuration files.  
//  
options {  
    listen-on port 53 { any; };  
    listen-on-v6 port 53 { any; };  
    directory "/var/named";  
    dump-file "/var/named/data/cache_dump.db";  
    statistics-file "/var/named/data/named_stats.txt";  
    memstatistics-file "/var/named/data/named_mem_stats.txt";  
    allow-query { any; };  
        allow-recursion {"none";};  
    recursion no;  
};  
logging {  
    channel default_debug {  
        file "data/named.run";  
        severity dynamic;  
    };  
};  
zone "." IN {  
    type hint;  
    file "named.ca";  
};  
include "/etc/named.conf.local";
```

Create the file `/etc/named.conf.local` that is included at the end of `/etc/named.conf` (`/etc/named.conf.local` will later on get populated by ISPConfig if you create DNS zones in ISPConfig):

```
touch /etc/named.conf.local
```

Then we create the startup links and start BIND:

```
systemctl enable named.service  
systemctl start named.service
```

## 18 Install Webalizer, And AWStats

Webalizer and AWStats can be installed as follows:

```
yum -y install webalizer awstats perl-DateTime-Format-HTTP perl-DateTime-Format-Builder
```

## 19 Install Jailkit

Jailkit is used to chroot SSH users and cronjobs. It can be installed as follows (**important: Jailkit must be installed before ISPConfig - it cannot be installed afterwards!**):

```
cd /tmp  
wget http://olivier.sessink.nl/jailkit/jailkit-2.17.tar.gz  
tar xvfz jailkit-2.17.tar.gz  
cd jailkit-2.17  
./configure  
make  
make install  
cd ..  
rm -rf jailkit-2.17*
```

## 20 Install fail2ban

This is optional but recommended, because the ISPConfig monitor tries to show the log.

```
yum -y install iptables-services fail2ban fail2ban-systemd
systemctl mask firewalld.service
systemctl enable iptables.service
systemctl enable ip6tables.service
systemctl stop firewalld.service
systemctl start iptables.service
systemctl start ip6tables.service
```

Next we create the `/etc/fail2ban/jail.local` file and enable monitoring for ssh, email and ftp service.

```
nano /etc/fail2ban/jail.local
```

Add the following content into the `jail.local` file:

```
[sshd]
enabled = true
action = iptables[name=sshd, port=ssh, protocol=tcp]

[pure-ftpd]
enabled = true
action = iptables[name=FTP, port=ftp, protocol=tcp]
maxretry = 3

[dovecot]
enabled = true
action = iptables-multiport[name=dovecot, port="pop3,pop3s,imap,imaps", protocol=tcp]
maxretry = 5

[postfix-sasl]
enabled = true
action = iptables-multiport[name=postfix-sasl, port="smtp,smtps,submission", protocol=tcp]
maxretry = 3
```

Then create the system startup links for fail2ban and start it:

```
systemctl enable fail2ban.service
systemctl start fail2ban.service
```

## 21 Install rkhunter

rkhunter can be installed as follows:

```
yum -y install rkhunter
```

## 22 Install Mailman

If you like to manage mailinglists with Mailman on your server, then install mailman now. Mailman is supported by ISPConfig, so you will be able to create new mailinglists through ISPConfig later.

```
yum -y install mailman
```

Before we can start Mailman, a first mailing list called `mailman` must be created:

```
touch /var/lib/mailman/data/aliases
postmap /var/lib/mailman/data/aliases
/usr/lib/mailman/bin/newlist mailman
```

```
[root@server1 tmp]# /usr/lib/mailman/bin/newlist mailman
Enter the email of the person running the list: <- admin email address, e.g. listadmin@example.com
Initial mailman password: <- admin password for the mailman list
To finish creating your mailing list, you must edit your /etc/aliases (or
equivalent) file by adding the following lines, and possibly running the
'newaliases' program:
```

```
## mailman mailing list
mailman: "|/usr/lib/mailman/mail/mailman post mailman"
mailman-admin: "|/usr/lib/mailman/mail/mailman admin mailman"
mailman-bounces: "|/usr/lib/mailman/mail/mailman bounces mailman"
mailman-confirm: "|/usr/lib/mailman/mail/mailman confirm mailman"
mailman-join: "|/usr/lib/mailman/mail/mailman join mailman"
mailman-leave: "|/usr/lib/mailman/mail/mailman leave mailman"
mailman-owner: "|/usr/lib/mailman/mail/mailman owner mailman"
mailman-request: "|/usr/lib/mailman/mail/mailman request mailman"
mailman-subscribe: "|/usr/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe: "|/usr/lib/mailman/mail/mailman unsubscribe mailman"
```

Hit enter to notify mailman owner... <- ENTER

```
[root@server1 tmp]#
```

Open `/etc/aliases` afterwards...

```
vi /etc/aliases
```

... and add the following lines:

```
[...]
mailman: "|/usr/lib/mailman/mail/mailman post mailman"
mailman-admin: "|/usr/lib/mailman/mail/mailman admin mailman"
mailman-bounces: "|/usr/lib/mailman/mail/mailman bounces mailman"
mailman-confirm: "|/usr/lib/mailman/mail/mailman confirm mailman"
mailman-join: "|/usr/lib/mailman/mail/mailman join mailman"
mailman-leave: "|/usr/lib/mailman/mail/mailman leave mailman"
mailman-owner: "|/usr/lib/mailman/mail/mailman owner mailman"
mailman-request: "|/usr/lib/mailman/mail/mailman request mailman"
mailman-subscribe: "|/usr/lib/mailman/mail/mailman subscribe mailman"
```

```
mailman-unsubscribe: "|/usr/lib/mailman/mail/mailman unsubscribe mailman"
```

Run

```
newaliases
```

afterwards and restart Postfix:

```
systemctl restart postfix.service
```

Now open the Mailman Apache configuration file `/etc/httpd/conf.d/mailman.conf...`

```
nano /etc/httpd/conf.d/mailman.conf
```

... and add the line `ScriptAlias /cgi-bin/mailman/ /usr/lib/mailman/cgi-bin/`. Comment out `Alias /pipemail/ /var/lib/mailman/archives/public/` and add the line `Alias /pipemail /var/lib/mailman/archives/public/`:

```
#
# httpd configuration settings for use with mailman.
#
```

```
ScriptAlias /mailman/ /usr/lib/mailman/cgi-bin/
ScriptAlias /cgi-bin/mailman/ /usr/lib/mailman/cgi-bin/
<Directory /usr/lib/mailman/cgi-bin/>
    AllowOverride None
    Options ExecCGI
    Order allow,deny
    Allow from all
</Directory>
```

```
#Alias /pipemail/ /var/lib/mailman/archives/public/
Alias /pipemail /var/lib/mailman/archives/public/
<Directory /var/lib/mailman/archives/public>
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
    AddDefaultCharset Off
</Directory>
```

```
# Uncomment the following line, to redirect queries to /mailman to the
# listinfo page (recommended).
```

```
# RedirectMatch ^/mailman/[ ]*$ /mailman/listinfo
```

Restart Apache:

```
systemctl restart httpd.service
```

Create the system startup links for Mailman and start it:

```
systemctl enable mailman.service
systemctl start mailman.service
```

After you have installed ISPCongig 3, you can access Mailman as follows:

You can use the alias `/cgi-bin/mailman` for all Apache vhosts (please note that **suExec and CGI must be disabled** for all vhosts from which you want to access Mailman!), which means you can access the Mailman admin interface for a list at `http://<vhost>/cgi-bin/mailman/admin/<listname>`, and the web page for users of a mailing list can be found at `http://<vhost>/cgi-bin/mailman/listinfo/<listname>`.

Under `http://<vhost>/pipemail/<listname>` you can find the mailing list archives.

## 23 Install Roundcube webmail

To install the Roundcube webmail client, run...

```
yum -y install roundcubemail
```

Change the roundcubemail configuration file as follows:

```
nano /etc/httpd/conf.d/roundcubemail.conf
```

```
#
# Round Cube Webmail is a browser-based multilingual IMAP client
#
```

```
Alias /roundcubemail /usr/share/roundcubemail
Alias /webmail /usr/share/roundcubemail
```

```
# Define who can access the Webmail
# You can enlarge permissions once configured
```

```
#<Directory /usr/share/roundcubemail/>
#   <IfModule mod_authz_core.c>
#       # Apache 2.4
#       Require local
#   </IfModule>
#   <IfModule !mod_authz_core.c>
#       # Apache 2.2
#       Order Deny,Allow
#       Deny from all
#       Allow from 127.0.0.1
#       Allow from ::1
#   </IfModule>
#</Directory>
```

```
<Directory /usr/share/roundcubemail/>
    Options none
    AllowOverride Limit
    Require all granted
```



```

</Directory>

# Define who can access the installer
# keep this secured once configured

#<Directory /usr/share/roundcubemail/installer/>
#   <IfModule mod_authz_core.c>
#       # Apache 2.4
#       Require local
#   </IfModule>
#   <IfModule !mod_authz_core.c>
#       # Apache 2.2
#       Order Deny,Allow
#       Deny from all
#       Allow from 127.0.0.1
#       Allow from ::1
#   </IfModule>
#</Directory>

<Directory /usr/share/roundcubemail/installer>
    Options none
    AllowOverride Limit
    Require all granted
</Directory>

# Those directories should not be viewed by Web clients.
<Directory /usr/share/roundcubemail/bin/>
    Order Allow,Deny
    Deny from all
</Directory>
<Directory /usr/share/roundcubemail/plugins/enigma/home/>
    Order Allow,Deny
    Deny from all
</Directory>

```

Restart Apache:

```
systemctl restart httpd.service
```

Now we need a database for roundcube mail, we will initialise it as follows:

```
mysql -u root -p
```

At mariadb prompt use:

```

CREATE DATABASE roundcubedb;
CREATE USER roundcubeuser@localhost IDENTIFIED BY 'roundcubepassword';
GRANT ALL PRIVILEGES ON roundcubedb.* TO roundcubeuser@localhost ;
FLUSH PRIVILEGES;
exit

```

I am using details for roundcube database as a test, please replace the values as per your choice for security reasons.

Now we will install the roundcube on browser at <http://192.168.1.100/roundcubemail/installer>

Database setup

db\_dsnw

Database settings for read/write operations:

MySQL

Database type

localhost

Database server (omit for sqlite)

roundcubedb

Database name (use absolute path and filename for sqlite)

roundcubeuser

Database user name (needs write permissions)(omit for sqlite)

\*\*\*\*\*

Database password (omit for sqlite)

db\_prefix

Optional prefix that will be added to database object names (tables and sequences).

Now fill the entries for the

```
nano /etc/roundcubemail/config.inc.php
```

```

<?php

/* Local configuration for Roundcube Webmail */

// -----
// SQL DATABASE
// -----
// Database connection string (DSN) for read+write operations
// Format (compatible with PEAR MDB2): db_provider://user:password@host/database
// Currently supported db_providers: mysql, pgsql, sqlite, mssql or sqlsrv
// For examples see http://pear.php.net/manual/en/package.database.mdb2.intro-dsn.php
// NOTE: for SQLite use absolute path: 'sqlite:///full/path/to/sqlite.db?mode=0646'
$config['db_dsnw'] = 'mysql://roundcubeuser:roundcubepassword@localhost/roundcubedb';

// -----
// IMAP
// -----
// The mail host chosen to perform the log-in.
// Leave blank to show a textbox at login, give a list of hosts
// to display a pulldown menu or set one host as string.
// To use SSL/TLS connection, enter hostname with prefix ssl:// or tls://
// Supported replacement variables:
// %n - hostname ($_SERVER['SERVER_NAME'])
// %t - hostname without the first part
// %d - domain (http hostname $_SERVER['HTTP_HOST'] without the first part)
// %s - domain name after the '@' from e-mail address provided at login screen
// For example %n = mail.domain.tld, %t = domain.tld
// WARNING: After hostname change update of mail_host column in users table is
//          required to match old user data records with the new host.
$config['default_host'] = 'localhost';

// provide an URL where a user can get support for this Roundcube installation
// PLEASE DO NOT LINK TO THE ROUNDcube.NET WEBSITE HERE!

```



```
$config['support_url'] = '';

// this key is used to encrypt the users imap password which is stored
// in the session record (and the client cookie if remember password is enabled).
// please provide a string of exactly 24 chars.
$config['des_key'] = 'FHgaM7ihtMkM1cBwck0cxPdT';

// -----
// PLUGINS
// -----
// List of active plugins (in plugins/ directory)
$config['plugins'] = array();

// Set the spell checking engine. Possible values:
// - 'googie' - the default
// - 'pspell' - requires the PHP Pspell module and aspell installed
// - 'enchant' - requires the PHP Enchant module
// - 'atd' - install your own After the Deadline server or check with the people at http://www.afterthedeathline.com before using their API
// Since Google shut down their public spell checking service, you need to
// connect to a Nox Spell Server when using 'googie' here. Therefore specify the 'spellcheck_uri'
$config['spellcheck_engine'] = 'pspell';
```

Then press on the button "continue" in the web installer. On the following page, press on the button "Initialize database".

Finally, disable the Roundcubemail installer. Change the apacheroundcubemail configuration file:

```
nano /etc/httpd/conf.d/roundcubemail.conf

#
# Round Cube Webmail is a browser-based multilingual IMAP client
#

Alias /roundcubemail /usr/share/roundcubemail
Alias /webmail /usr/share/roundcubemail

# Define who can access the Webmail
# You can enlarge permissions once configured

#<Directory /usr/share/roundcubemail/>
#   <IfModule mod_authz_core.c>
#       # Apache 2.4
#       Require local
#   </IfModule>
#   <IfModule !mod_authz_core.c>
#       # Apache 2.2
#       Order Deny,Allow
#       Deny from all
#       Allow from 127.0.0.1
#       Allow from ::1
#   </IfModule>
#</Directory>

<Directory /usr/share/roundcubemail/>
    Options none
    AllowOverride Limit
    Require all granted
</Directory>

# Define who can access the installer
# keep this secured once configured

<Directory /usr/share/roundcubemail/installer/>
    <IfModule mod_authz_core.c>
        # Apache 2.4
        Require local
    </IfModule>
    <IfModule !mod_authz_core.c>
        # Apache 2.2
        Order Deny,Allow
        Deny from all
        Allow from 127.0.0.1
        Allow from ::1
    </IfModule>
</Directory>

# Those directories should not be viewed by Web clients.
<Directory /usr/share/roundcubemail/bin/>
    Order Allow,Deny
    Deny from all
</Directory>
<Directory /usr/share/roundcubemail/plugins/enigma/home/>
    Order Allow,Deny
    Deny from all
</Directory>
~
```

Restart Apache:

```
systemctl restart httpd.service
```

## 24 Install ISPConfig 3

Download the [current ISPConfig 3 version](#) and install it. The ISPConfig installer will configure all services like Postfix, Dovecot, etc. for you. A manual setup as required for ISPConfig 2 is not necessary anymore.

You now also have the possibility to let the installer create an SSL vhost for the ISPConfig control panel, so that ISPConfig can be accessed using <https://> instead of <http://>. To achieve this, just press *ENTER* when you see this question: *Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]:*.

To install ISPConfig 3 from the latest released version, do this:

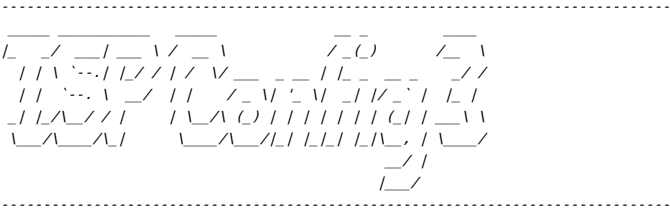
```
cd /tmp
wget http://www.ispconfig.org/downloads/ISPConfig-3-stable.tar.gz
tar xzf ISPConfig-3-stable.tar.gz
cd ispconfig3_install/install/
```

The next step is to run

```
php -q install.php
```

This will start the ISPConfig 3 installer:

```
[root@server1 install]# php -q install.php
```



>> Initial configuration

Operating System: Redhat or compatible, unknown version.

Following will be a few questions for primary configuration so be careful.  
Default values are in [brackets] and can be accepted with <ENTER>.  
Tap in "quit" (without the quotes) to stop the installer.

Select language (en,de) [en]: <- ENTER

Installation mode (standard,expert) [standard]: <- ENTER

Full qualified hostname (FQDN) of the server, eg server1.domain.tld [server1.example.com]: <- ENTER

MySQL server hostname [localhost]: <- ENTER

MySQL root username [root]: <- ENTER

MySQL root password []: <- yourrootsqlpassword

MySQL database to create [dbisconfig]: <- ENTER

MySQL charset [utf8]: <- ENTER

Generating a 2048 bit RSA private key  
.....++  
.....++  
writing new private key to 'smtpd.key'  
-----

You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----

Country Name (2 letter code) [XX]: <- ENTER  
State or Province Name (full name) []: <- ENTER  
Locality Name (eg, city) [Default City]: <- ENTER  
Organization Name (eg, company) [Default Company Ltd]: <- ENTER  
Organizational Unit Name (eg, section) []: <- ENTER  
Common Name (eg, your name or your server's hostname) []: <- ENTER  
Email Address []: <- ENTER  
Configuring Jailkit  
Configuring Dovecot  
Configuring Spamassassin  
Configuring Amavisd  
Configuring Getmail  
Configuring Pureftpd  
Configuring BIND  
Configuring Apache  
Configuring Vlogger  
Configuring Apps vhost  
Configuring Bastille Firewall  
Configuring Fail2ban  
Installing ISPConfig  
ISPConfig Port [8080]: <- ENTER

Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]: <- ENTER

Generating RSA private key, 4096 bit long modulus  
.....++  
.....++  
e is 65537 (0x10001)  
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [XX]: <- ENTER  
State or Province Name (full name) []: <- ENTER  
Locality Name (eg, city) [Default City]: <- ENTER  
Organization Name (eg, company) [Default Company Ltd]: <- ENTER  
Organizational Unit Name (eg, section) []: <- ENTER

Common Name (eg, your name or your server's hostname) []: <-- ENTER  
Email Address []: <-- ENTER

Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []: <-- ENTER  
An optional company name []: <-- ENTER  
writing RSA key  
Configuring DBServer  
Installing ISPConfig crontab  
no crontab for root  
no crontab for getmail  
Restarting services ...  
Stopping mysqld: [ OK ]  
Starting mysqld: [ OK ]  
Shutting down postfix: [ OK ]  
Starting postfix: [ OK ]  
Stopping saslauthd: [FAILED]  
Starting saslauthd: [ OK ]  
Waiting for the process [1424] to terminate  
Shutting down amavisd: Daemon [1424] terminated by SIGTERM [ OK ]  
amavisd stopped  
Starting amavisd: [ OK ]  
  
Stopping clamd.amavisd: [ OK ]  
Starting clamd.amavisd: [ OK ]  
Stopping Dovecot Imap: [ OK ]  
Starting Dovecot Imap: [ OK ]  
Stopping httpd: [ OK ]  
[Thu Mar 14 14:12:32 2013] [warn] NameVirtualHost \*:80 has no VirtualHosts  
Starting httpd: [ OK ]  
Stopping pure-ftpd: [ OK ]  
Starting pure-ftpd: [ OK ]  
Installation completed.  
[root@server1 install]#

The error message "usage: doveadm [-Dv] [-f <formatter>] <command> [<args>]" can be ignored, in case that you get it during ispconfig installation.

To fix the Mailman errors you might get during the ISPConfig installation, open /usr/lib/mailman/Mailman/mm\_cfg.py...

```
vi /usr/lib/mailman/Mailman/mm_cfg.py
```

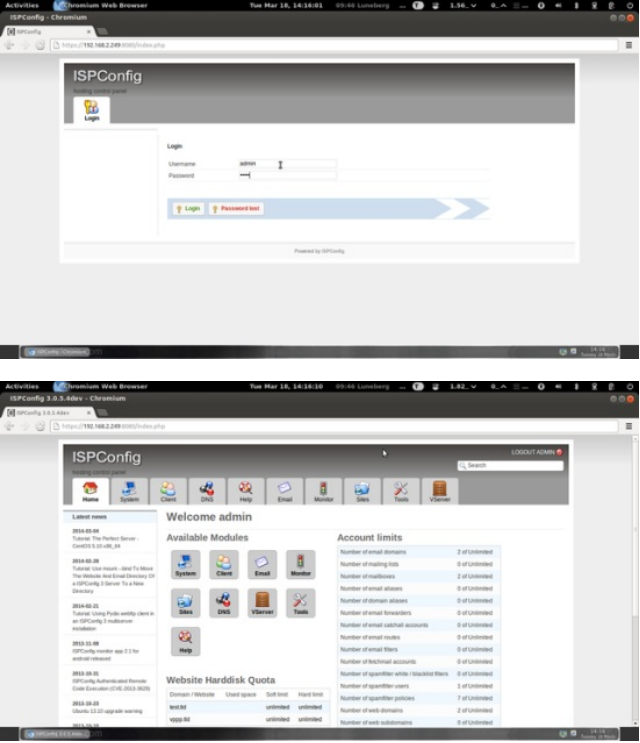
... and set DEFAULT\_SERVER\_LANGUAGE = 'en':

```
[...]
#-----
# The default language for this server.
DEFAULT_SERVER_LANGUAGE = 'en'
[...]
```

Restart Mailman:

```
systemctl restart mailman.service
```

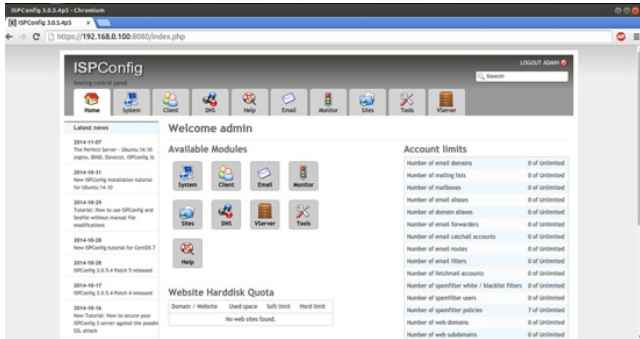
Afterwards you can access ISPConfig 3 under `http(s)://server1.example.com:8080/` or `http(s)://192.168.1.100:8080/` (`http` or `https` depends on what you chose during installation). Log in with the username `admin` and the password `admin` (you should change the default password after your first login):



25 First ISPConfig Login

Afterwards you can access ISPConfig 3 under `http(s)://server1.example.com:8080/` or `http(s)://192.168.0.100:8080/` (`http` or `https` depends on what you chose during installation).

Log in with the username `admin` and the password `admin` (you should change the default password after your first login):



The screenshot shows the ISPConfig 3.10.0 web interface. The top navigation bar includes the logo and version number, and a 'Logout Admin' button. Below it is a menu bar with icons for Home, Systems, Client, DNS, Help, Mail, Monitor, Sites, Tools, and View user. The left sidebar contains a tree view of the application's structure, with 'Server Management' expanded and 'Server Config' selected. The main content area displays the 'Server Configuration' page for 'server1.example.com', showing fields for Name, IP Address, and Hostname, and a 'Page 1' indicator at the bottom.

Home
 System
 Client
 DNS
 Help
 Email
 Monitor
 Users
 Tools
 Viewer

IPConfig  
 Installing control panel

LOGOUT Admin

User Management  
 Of Users  
 Monitor Users  
 Systems  
 Server Services  
 Server Config  
 Server IP addresses  
 Additional client Services  
 Diagnostic Insights  
 Firewall  
 WebServer  
 Java Config  
 Apple II Additions  
 Macintoshes  
 Patches  
 Updates  
 Language Editor  
 Languages  
 New Language  
 Export  
 Import  
 Removable Activities  
 Do OS Update  
 Do OSConfig Update

## Server Config

[Server] [Mail] [Network] [Web] [DNS] [FastCGI] [JDK] [Logger] [Other] [Reboot]

EMD User	named
EMD Group	named
EMD servicefile directory	/etc/named.conf
EMD named.conf path	/etc/named.conf
EMD named.conf.local path	/etc/named.conf.local

```
BIND zonefiles directory: /var/named
BIND named.conf path: /etc/named.conf
BIND named.conf.local path: /etc/named.conf.local
```

On more than 300 pages, it covers the concept behind ISPConfig (admin, resellers, clients), explains how to install and update ISPConfig 3, includes a reference for all forms and form fields in ISPConfig together with examples of valid inputs, and provides tutorials for the most common tasks in ISPConfig 3. It also lines out how to make your server more secure and comes with a troubleshooting section at the end.

- CentOS: <http://www.centos.org/>
- ISPConfig: <http://www.ispconfig.org/>