Install Slackware on a VPS based on Proxmox VE or SolusVM

I've been faced with the challenge to install Slackware as a VPS at a hosting provider that had Proxmox VE as virtualization platform.

Proxmox VE is a complete open source virtualization management solution for servers. It is based on KVM virtualization and container-based virtualization and manages virtual machines, storage, virtualized networks, and HA Clustering. Further information is contained in the Proxmox Wiki.

Recently I've had to migrate to a SolusVM based infrastructure, using KVM virtualization. This does not apply to Xen virtualization by SolusVM which officially supports Slackware. Assuming your is configured with virtio type disk, all the instructions below apply the same.

SolusVM is a commercial virtualization infrastructure able to provide KVM virtualization, OpenVZ containers and Xen. Further information is contained in the SolusVM Documentation.

Prerequisites

This guide is valid for **Slackware-14.1 only**. Slackware-14.2 already has the virtio support in the default kernel that is deployed at installation. *(verified at least with >4.4.190 kernel build)*

If you build your own kernel make sure to include virtio* drivers built-in, not modules.

You must ask your hosting provider to add the Slackware install DVD ISO image to the virtual CD/DVD device of the virtual machine.

Also, you must have a local machine (either physical or virtual by other means) with a running copy of the same Slackware version that you will be installing on the Proxmox VE.

Preparing a custom initrd

On your own running Slackware installation (I've done this using Slackware64-14.1) you must prepare an `initrd` for your virtual server.
Make sure you have the proper kernel version, if your installation was updated you may need to temporarily downgrade the kernel to match the version in the installation disk.
Alternately, you may through in a quick installation in VMware Player, build your initrd there, then deploy it to a FTP server.

It is very simple, with `mkinitrd` tool.

```
# mkinitrd -m virtio_pci:virtio_mmio:virtio_balloon:virtio_blk -k 3.10.17 -f
ext4 -r /dev/vda1
```

This creates an initial ramdisk (initrd.gz) with the virtio* modules, for booting a root partition

/dev/vda1 formatted using etx4 filesystem.
Adjust according to your needs and use the same partition name and fs when you install your VM.

The resulting ramdisk will be here: `/boot/initrd.gz` Upload this file to a FTP or HTTP server so that it may be downloadable using `wget`. That server's IP address must be accessible to the hosted VM.
It must be a direct download link, so don't use a filesharing system like Rapidshare, etc.
Remember, the Slackware installation disk has a minimum set of utilities.
Also note that you may not have the resolver library working, so I recommend using plain FTP server and call it by IP instead of hostname.

Installation

Booting from the ISO image presents no problems.

After booting, as usual, start by partitioning the (virtual) disk. You will discover that instead of the usual *sda* or *hda*, you are presented with devices such as:

```
/dev/vda
```

Not to worry, it's the same as `/dev/sda`, but it is the source of several issues that need to be solved later.

Partition your disk, run `setup` and perform the installation.

Configuration

Upon configuring `lilo`, choose **Advanced**.
Simple `lilo` autoconfiguration defaults to `/dev/sda` and won't generate the proper lilo.conf.
Using *Advanced* you will manually tell `lilo` to install itself on `dev/vda`.
Configure the rest of the parameters as needed.

Installing `lilo` will fail due to unrecognized block device `0xfc00`. Accept the error and proceed with the rest of the configuration process.

DO NOT REBOOT after the configuration process ends.

Exit the `setup` program and configure your networking.

```
# ip a add IP_ADDRESS/NETMASK dev eth0
# ip link set eth0 up
# ip ro add default via GATEWAY dev eth0
```

Go to the `/boot` directory of the installation and download the initrd.gz file

```
# cd /mnt/boot
# wget URL_TO_YOUR_SERVER/initrd.gz
```

Go to the `/etc` directory of your installation and edit lilo.conf

```
# cd /mnt/etc
# vi lilo.conf
```

Right under boot= add the following (without the comments):

```
disk=/dev/vda               # your disk device
bios=0x80                   # first disk according to BIOS
max-partitions=15           # as usual for SCSI disks, this refers to no. of
disks
                            # on a controller, see man lilo.conf
```

Scroll down to where the bootable partition is configured and add below root = :

```
initrd = /boot/initrd.gz
```

Save and run lilo:

```
lilo -C lilo.conf
```

You will get some warnings, but it will work.

Now you can reboot your system.

Important: you must rebuild the initrd.gz file every time you upgrade or change the kernel.

Sources

- Originally written by Andrei B.

howtos, virtualization, kvm, proxmox, solus, solusvm

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:slackware_admin:slackware-proxmox-ve

Last update: **2020/01/13 19:42 (UTC)**