

Install FreeBSD with GPT partitions

Wednesday, 15 June 2011

Last Updated Friday, 15 July 2011

For now sysinstall does not support GPT partition. So everything must be installed manually but it is easy. This quick tutorial shows you step by step installation using GPT partitions. Why GPT? Better support for bigger drives (>2TB) and up to 128 partitions instead of 4.

Step 1. Partitioning the drive

We boot from FreeBSD 8.2 USB image and then we go to Fixit menu and then from there we mount USB Live image, that has all necessary tools to install FreeBSD.

First we will initialize the drive to support GPT partitions:

```
gpart create -s GPT ad0
```

Then we will proceed creating partitions. We will create partitions for boot, root, tmp, var, usr.

```
gpart add -s 128 -t freebsd-boot -l boot ad0          # for boot   (1)
gpart add -s 5G -t freebsd-ufs -l root ad0           # for / (root) (2)
gpart add -s 4G -t freebsd-ufs -l tmp ad0            # for tmp    (3)
gpart add -s 4G -t freebsd-swap -l swap ad0         # for swap   (4)
gpart add -s 10G -t freebsd-ufs -l var ad0           # for var    (5)
gpart add -s 200G -t freebsd-ufs -l usr ad0         # for usr and home (6)
```

To see partitions you've just created use:

```
gpart show ad0
```

To see partition with labels use:

```
gpart show -l ad0
```

To delete a partition use:

```
gpart delete -i3 ad0      # where 3 is third partition.
```

If you want to remove GPT table from drive ad0 run (in case you want start partitioning process again):

```
gpart destroy ad0
```

Now we will place the MBR code in our freebsd-boot partition:

```
gpart bootcode -b /mnt2/boot/pmbr -p /mnt2/boot/gptboot -i 1 ad0
```

We need now to newfs (format) our newly created partitions:

```
newfs -U /dev/ad0p2
newfs -U /dev/ad0p3
newfs -U /dev/ad0p5
newfs -U /dev/ad0p6
```

We will mount now all partitions to install FreeBSD on them:

```
mount /dev/ad0p2 /mnt
mkdir /mnt/boot /mnt/tmp /mnt/var /mnt/usr
mount /dev/ad0p3 /mnt/tmp
mount /dev/ad0p5 /mnt/var
mount /dev/ad0p6 /mnt/usr
```

Step 2. Installing FreeBSD

We will now use the install.sh script from our FreeBSD USB Image:

```
export DESTDIR=/mnt
cd /dist/8.2-RELEASE
for dir in base catpages dict doc info lib32 manpages; do (cd $dir; ./install.sh); done
```

If you install FreeBSD i386 you will not need lib32 in the for loop.
Now we install the kernel:

```
cd kernels
./install.sh GENERIC
```

We install the sources:

```
cd ..
cd src
./install.sh all
```

We copy the kernel in /mnt/boot/kernel directory:

```
cd /mnt/boot
mkdir kernel
cp -Rp GENERIC kernel
```

Step 3. Configuring FreeBSD

Last step is to configure FreeBSD by creating /etc/fstab, /etc/resolv.conf and /etc/rc.conf files.

We edit /mnt/etc/resolv.conf file and add our dns there:

```
nameserver 208.67.220.220      # that is your ISP's DNS
```

We create /etc/rc.conf with the following content (asuming em0 is our network card):

```
/etc/rc.confdefaultrouter="10.0.0.1"
hostname="server"
ifconfig_em0="inet 10.0.0.2 netmask 255.255.255.0"
sshd_enable="YES"
```

We must create an /etc/fstab file with the following content:

/etc/fstab#	Device	Mountpoint	FStype	Options	Dump	Pass#
/dev/ad0p4	none	swap	sw	0 0		
/dev/ad0p2	/	ufs	rw	1 1		
/dev/ad0p3	/tmp	ufs	rw	2 2		
/dev/ad0p6	/usr	ufs	rw	2 2		
/dev/ad0p5	/var	ufs	rw	2 2		
/dev/cd0	/cdrom	cd9660	ro,noauto	0 0		

GPT Partitioning Tips

- If you get the error "Device Busy" when you try to delete GPT from drive with `gpart delete ad0`, you can successfully delete GPT if you turn of the safety with `sysctl` variable:

`sysctl kern.geom.debugflags=16`

- If you want to destroy GPT table from the drive with `gpart delete ad0` command you must first delete all partitions (Otherwise you might get "Device Busy" error).