



BIOS | EFI

EFI- acronym for Extensible Firmware Interface BIOS-stands for Basic Input/Output System.

POST

POST (Power-On Self Test) - a few very basic routines intended to insure the hardware is not obviously faulty.

MBR

MBR (Master Boot Record) is installed on the disk by fdisk(8).

PBR

PBR (Partition Boot Record). The first 512 bytes of the boot disk's OpenBSD partition contain the first stage boot loader bootboot(8). It is installed by the installboot(8) utility.

/boot

Second stage boot loader /boot. The boot(8) program locates and loads the kernel.

Kernel

The kernel (ELF executable file) is loaded in memory.

init

The init program is the last stage of the boot process. Single-user mode is also entered if the boot scripts fail.

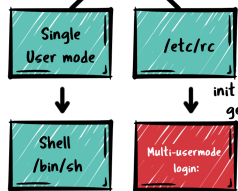
Messages from console

The BIOS/EFI search and execute the MBR

```
Using drive 0, partition 3
Loading.....
probing: pc0 com0 com1 mem[638K 1918M a20=on]
[Using 386464 bytes of bsd ELF symbol table]
```

Files

```
/usr/mdec/mbr
/usr/mdec/bootboot
/boot
/etc/boot.conf
/bsd - OpenBSD kernel
/bsd.sp - OpenBSD kernel for single processor machines
/bsd.mp - OpenBSD kernel for multiprocessor machines
/bsd.rd - OpenBSD kernel for installation/recovery
```



Mainly:

- T sends SIGINFO (status request)
- Set the domainname for YP (Yellow Pages) if the file /etc/defaultdomain is present
- Add swap block-devices
- Run filesystem check unless a /fastboot file exists
- Unmount all filesystems except root
- Mount all filesystems except those of type NFS and VND
- Re-mount the root filesystem read/writeable
- Backup /bsd to /bsd.booted
- Remove /fastboot
- Set flags on ttys
- Set keyboard encoding
- Apply wscnscctl.conf(5) settings
- Set initial temporary PF ruleset
- Populate net.inet.tcp.pluidp.baddynamic with the contents of /etc/services so as to avoid randomly allocating source ports that correspond to well-known services
- Apply sysctl.conf(5) settings and update resource limits based on login.conf settings
- Start (if enabled) the slacsd daemon (lpv6)
- Start network using the script /etc/netstart
- Check /usr and /var are mounted
- Start the unwind daemon (validating DNS resolver)
- Load PF rules and bring up pfsync interface
- Generate random seed, push the old seed into the kernel, create a future seed and create a seed file for the boot-loader
- Re-link libraries, placing the objects in a random order
- Clean up left-over files
- Save a copy of the boot messages (/var/run/dmesg.boot)
- Generate keys for isakmpd,iked and schd if they don't exist yet
- starting early daemons (if enabled): syslog ldattach pflogd nsd unbound ntpd issicid isakmpd iked casync ldapd npppd
- Load IPsec rules if it is enabled
- Start (if enabled) RPC daemons: portmap updap upsync vrbind mountd nfsd lockd statd amd
- Check and mount remaining file systems and enable additional swap
- Save core dumps in /var/crash
- Store ACPI tables in /var/db/acpi to be used by sendbug(1)
- Check quotas
- Build kvm(3) and /dev databases
- Set proper permission for the tty device files
- Check for the password temp/lock file
- Clearing /tmp
- Create Unix sockets directories for X if needed and make sure they have correct permissions
- Run /etc/rc.securelevel if present
- Set kern.securelevel (default is 1)
- Patch /etc/motd
- Turning on accounting if the feature is enabled
- Creating runtime link editor directory cache
- Preserving editor files (vi)
- Start (if enabled) network daemons: ldomd schd switchd snmpd lpd ripd ospfd ospf6d bgpd ifstated relayd dhcpcd dncrelay mouted dvmpd radiusd eigrpd routebd rad hostapd lpd smtpd slowcgi httpd ftpd ftpproxy ftpproxy tftpd tftpproxy identd inetd carpd bootparamd rbootd mopd vmd spamd spamlogd sniod
- If the file /etc/rc.firsttime exists, run it just once, and make sure it is deleted
- * After a fresh install, this file contains fw_update(1) & syspatch(8)
- Run rc.d(8) scripts from packages
- Run /etc/rc.local if present
- Disable carp interlock
- Apply mixerctl.conf(5) settings
- Start (if enabled) local daemons: apmd sensorsd hotplugd watchdogd cron wsmoused xenodm
- Re-link the kernel, placing the objects in a random order, replace current with relinked kernel and inform root about it
- And finally show date(1)

Full OpenBSD Startup described

By OpenBSDJumpstart.org

