



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.

THE BIOS/EFI SEARCH AND EXECUTE THE MBR

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 BIOSBOOT(8). IT IS INSTALLED BY THE INSTALLBOOT(8) UTILITY.

/boot

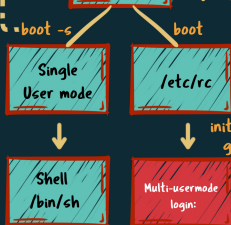
SECOND STAGE BOOT LOADER /BOOT, THE BOOT(8) PROGRAM LOCATES AND LOADS THE KERNEL.

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

```

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

### FILES



### 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 flags
- Set keyboard encoding
- Apply wscnscctl.conf(5) settings
- Set initial temporary PF ruleset
- Populate net.inet.tcp.pludp.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 slaacd daemon (ipvb)
- Start network using the script /etc/netstart
- Check /usr and /var are mounted
- Start the daemons dnpleased, unwind (validating DNS resolver) and resolvd(handle nameserver configuration)
- Load PF rules and bring up pfync 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): syslogd, ldattach, pflogd, nsd, unbound, ntpd, icssd, isakmpd, iked, sasync, ldapd, and npppd
- Load IPsec rules if it is enabled
- Start (if enabled) RPC daemons: portmap, yppadm, ypserv, ypbind, mountd, nfsd, lockd, statd, and amd
- Check and mount remaining file systems and enable additional swap
- Build kvm(3) and /dev databases
- Save core dumps in /var/crash
- Store ACPI tables in /var/db/acpi to be used by sendbug(1)
- Check quotas
- 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: ldomb, schd, switchd, snmpd, ldap, ripd, ospfd, ospfd, bgpd, ifstated, relayd, dhcpcd, dhcrelay, mrouted, dvmpd, radiusd, eigrpd, routebd, rad, hostapd, lpd, smtpd, slowcgi, httpd, ftpd, ftpproxy, ftpproxyb, tftpd, tftpproxy, identd, inetd, rarpd, bootparamd, rbootd, mopd, vmd, spamd, spamlogd and sndiod
- 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\_updatel1) & 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, senced, hotplugd, watchdogd, cron, wsmoused and sendm
- Re-link the kernel, placing the objects in a random order, replace current with/relinked kernel and inform root about it
- And finally show datel1)

# Full OpenBSD 6.9 Startup described By OpenBSDJumpstart.org

