

هنا سوف نحاول إيضاح كيف يعمل جهازك في بدء التشغيل حتى الوصول إلى سطر الأوامر (bash) وهو عرض بالتسلسل تم اعتماد نسخة عرب سلاك في هذا العرض هناك اختلاف بسيط بينة وبين التوزيعات الأخرى:

. العتاد:

عندما تقوم بتشغيل جهازك سوف يتم التحقق من أن كل شي في الجهاز يعمل وتتم العملية من خلال برنامج موجود في البيوس (bios) ويسمى (boot strap loader) يقوم بالبحث عن بوت سيكتور (boot sector) والبوت سيكتور هو أول جزء من القرص الصلب يحوي برنامج صغير يستطيع هذا البرنامج تحميل نظام تشغيل والبوت سيكتور موقع برقمين مميزين هما (0XAASS=43603) و (0X1FE=510) وهي الطريقة التي يتعرف بها الجهاز على البوت سيكتور .

برنامج (boot strap loader) لديه قائمة يبحث بها عن البوت سيكتور من ضمن القائمة في جهازي القرص المدمج والقرص المرن والقرص الصلب يتم تحميل البوت سيكتور في الذاكرة من هذه القائمة ثم تسلم المهمة إلى برنامج يقوم بتحميل النظام في لينكس هذا البرنامج غالباً ما يكون ليلو (LILO) .

. ليلو (LILO)

عندما يقوم الجهاز بتحميل البوت سيكتور في جهاز لينكس هو حقيقة يقوم بتحميل جزء من ليلو يسمى (first stage boot loader) وهو البرنامج صغير جداً مهمته الوحيدة هي تحميل الجزء الثاني من ليلو ويسمى (second stage boot loader) الجزء الثاني يعطيك سطر الأوامر ويخير بين أنظمة التشغيل لديك .

عندما يعمل لينكس عند تنفيذ أمر (LILO) في الحقيقة أنت تنفذ خارطة التثبيت ويتم إعداد هذه الخارطة من خلال الملف (/etc/lilo.conf) ويتم كتابة مرشد النظام ومعلومات عن أنظمة التشغيل التي يمكنك تحميلها في القرص الصلب.

إعداد ليلو (/etc/lilo.conf) طبعاً هناك الكتلوج (man lilo.conf) تجدر الإشارة أن الإعداد لي لينكس يكون بتحديد أين تقع النواة والقسم الروت أما الأنظمة الأخرى فيتم تحديد القسم الذي يوجد فيها النظام .

تقوم النواة بالكثير من الأعمال هي من تحمل العتاد للقيام بما يريده البرنامج ، يقوم المعالج بتنفيذ أمر واحد فقط كل مرة لكن لينكس يظهر أنه يقوم بعدد كبير من المهام في وقت واحد تتمكن النواة من القيام بذلك من خلال الانتقال من مهمة إلى أخرى بشكل سريع جداً وبهذه الطريقة يتم الاستفادة بالشكل الأمثل من موارد النظام.

عند تحميل النواة أول مهمة تقوم بها النواة هو البحث عن برنامج (init) لتشغيله ، أفضل طريقة للتعرف على النواة هي القيام ببنائها أو التعرف على خيارات البناء من خلال أمر (make xconfig) في مجلد (/usr/src/linux/) يشترط أن يكون مصدر النواة موجود.

ملف (system.map) يستخدم من قبل النواة لتحديد التعريف الذي ينتج منه رسائل وأيضاً يتم استخدامه من قبل برنامج (top) ، يمكنك توجيه أوامر وتوجيهات للنواة عن طريق ليلو وهناك طريقتين:

-الطريقة الأولى هي من خلال ملف (/etc/lilo.conf) ويتم توجيهها في كل مرة عند الإقلاع

-الطريقة الثانية من خلال سطر الأوامر لـ ليلو مثال :

لفرض أنك نقلت ملفات النظام من القسم (/dev/hda2) إلى (/dev/hda3) وتريد تجربة النظام قبل نقل النواة من مكانها الأصلي (/dev/hda2) تتم العملية بالشكل التالي:

```
lilo :linux root =/dev/hda3 rw
```

مكتبات لغة سي من جنيو (gnu c library) عندما يبدأ عمل (init) يحتاج إلى مكتبات مثل أي برنامج آخر وهذا مثال للتوضيح برنامج بسيط بلغة سي :

```
main() {  
printf ("hello Linux!\n");  
}
```

هذا البرنامج لا يحوي تعريف لـ (printf) من أين يأتي التعريف إذا ؟

يأتي التعريف من مكتبات لغة سي المعيارية إذا كنت تستخدم جنيو / لينكس تأتي من (glibc) أما إذا كنت تعمل تحت " visualC++ " تأتي من مايكروسوفت هناك الملايين من هذه الدوال تختص بالحساب وتمثيل الوقت والزمن والكثير الكثير عندما تنظر إلى مجلد (/lib) في نظام لينكس ستجد الكثير من المكتبات على هيئة (libxxx.so) أو (libxxx.a) هناك طريقتين يمكن استخدام المكتبات بهما عندما يتم بناء البرنامج بطريقة (statically) يتم إدماج هذه المكتبات ضمن البرنامج بشكله النهائي لهذه العملية يتم استخدام المكتبات (libxxx.a) لهذا الغرض ، أما إذا تم بناء البرنامج بطريقة (dynamically) وهذا الافتراضي ولا يتم تضمين المكتبات في البرنامج بشكله النهائي وعندما يتم تنفيذ البرنامج يتم استدعاء هذه المكتبات ، تستخدم مكتبات (libxxx.so) لهذا الغرض .

تستطيع معرفة المكتبات التي يحتاجها برنامج معين عن طريق البرنامج (ldd) مثال :

```
ldd /opt/kde/bin/k3b
```

ستجد قائمة طويلة من المكتبات التي يعتمد عليها برنامج k3b

```
root@hathalsal:~# ldd /opt/kde/bin/k3b  
/lib/libsafe.so.2 (0x40017000)  
linux-gate.so.1 => (0xffffe000)  
libk3bplugin.so.2 => /opt/kde/lib/libk3bplugin.so.2 (0x4001c000)  
libartskde.so.1 => /opt/kde/lib/libartskde.so.1 (0x4004a000)
```

libkmedia2_idl.so.1 => /opt/kde/lib/libkmedia2_idl.so.1 (0x400a9000)
 libmcop.so.1 => /opt/kde/lib/libmcop.so.1 (0x400ed000)
 libkparts.so.2 => /opt/kde/lib/libkparts.so.2 (0x401aa000)
 libk3bdevice.so.2 => /opt/kde/lib/libk3bdevice.so.2 (0x401ee000)
 libk3bcore.so.2 => /opt/kde/lib/libk3bcore.so.2 (0x40210000)
 libk3btools.so.2 => /opt/kde/lib/libk3btools.so.2 (0x40234000)
 libk3bproject.so.2 => /opt/kde/lib/libk3bproject.so.2 (0x4028b000)
 libkio.so.4 => /opt/kde/lib/libkio.so.4 (0x40423000)
 libkdeui.so.4 => /opt/kde/lib/libkdeui.so.4 (0x4074b000)
 libkdecore.so.4 => /opt/kde/lib/libkdecore.so.4 (0x40a0c000)
 libDCOP.so.4 => /opt/kde/lib/libDCOP.so.4 (0x40c39000)
 libqt-mt.so.3 => /usr/lib/qt/lib/libqt-mt.so.3 (0x40c6d000)
 libpthread.so.0 => /lib/libpthread.so.0 (0x4134b000)
 libstdc++.so.5 => /usr/lib/.libstdc++.so.5 (0x4139d000)
 libgcc_s.so.1 => /usr/lib/.libgcc_s.so.1 (0x41455000)
 libc.so.6 => /lib/libc.so.6 (0x4145d000)
 libdl.so.2 => /lib/libdl.so.2 (0x41577000)
 libm.so.6 => /lib/libm.so.6 (0x4157a000)
 libkdesu.so.4 => /opt/kde/lib/libkdesu.so.4 (0x4159c000)
 libkwalletclient.so.1 => /opt/kde/lib/libkwalletclient.so.1 (0x415b6000)
 libresolv.so.2 => /lib/libresolv.so.2 (0x415c9000)
 libutil.so.1 => /lib/libutil.so.1 (0x415db000)
 libart_lgpl_2.so.2 => /usr/lib/.libart_lgpl_2.so.2 (0x415de000)
 libkdefx.so.4 => /opt/kde/lib/libkdefx.so.4 (0x415f5000)
 libqtmcop.so.1 => /opt/kde/lib/libqtmcop.so.1 (0x41621000)
 libmng.so.1 => /usr/lib/.libmng.so.1 (0x41628000)
 libjpeg.so.62 => /usr/lib/.libjpeg.so.62 (0x41683000)
 libGL.so.1 => /usr/lib/.libGL.so.1 (0x416a0000)
 libXmu.so.6 => /usr/X11R6/lib/libXmu.so.6 (0x4171c000)
 libXrender.so.1 => /usr/X11R6/lib/libXrender.so.1 (0x41732000)
 libXrandr.so.2 => /usr/X11R6/lib/libXrandr.so.2 (0x4173a000)
 libXcursor.so.1 => /usr/X11R6/lib/libXcursor.so.1 (0x4173e000)
 libXinerama.so.1 => /usr/X11R6/lib/libXinerama.so.1 (0x41748000)
 libXft.so.2 => /usr/X11R6/lib/libXft.so.2 (0x4174b000)
 libfontconfig.so.1 => /usr/X11R6/lib/libfontconfig.so.1 (0x4175d000)
 libfreetype.so.6 => /usr/lib/.libfreetype.so.6 (0x41784000)
 libexpat.so.0 => /usr/lib/.libexpat.so.0 (0x417ee000)
 libpng.so.3 => /usr/lib/.libpng.so.3 (0x4180e000)
 libz.so.1 => /usr/lib/.libz.so.1 (0x41840000)
 libXext.so.6 => /usr/X11R6/lib/libXext.so.6 (0x41851000)
 libX11.so.6 => /usr/X11R6/lib/libX11.so.6 (0x4185f000)
 libSM.so.6 => /usr/X11R6/lib/libSM.so.6 (0x41929000)

```

libICE.so.6 => /usr/X11R6/lib/libICE.so.6 (0x41932000)
libsoundserver_idl.so.1 => /opt/kde/lib/libsoundserver_idl.so.1 (0x4194a000)
libartsflow.so.1 => /opt/kde/lib/libartsflow.so.1 (0x419b6000)
libesd.so.0 => /usr/lib/.libesd.so.0 (0x41afb000)
libaudiofile.so.0 => /usr/lib/.libaudiofile.so.0 (0x41b05000)
libasound.so.2 => /usr/lib/.libasound.so.2 (0x41b29000)
libvorbisfile.so.3 => /usr/lib/.libvorbisfile.so.3 (0x41bd8000)
libvorbisenc.so.2 => /usr/lib/.libvorbisenc.so.2 (0x41be0000)
libvorbis.so.0 => /usr/lib/.libvorbis.so.0 (0x41cc9000)
libogg.so.0 => /usr/lib/.libogg.so.0 (0x41cf1000)
libmad.so.0 => /usr/lib/.libmad.so.0 (0x41cf6000)
libartsflow_idl.so.1 => /opt/kde/lib/libartsflow_idl.so.1 (0x41d0b000)
libgmodule-2.0.so.0 => /usr/lib/.libgmodule-2.0.so.0 (0x41dc5000)
libgthread-2.0.so.0 => /usr/lib/.libgthread-2.0.so.0 (0x41dc9000)
libglib-2.0.so.0 => /usr/lib/.libglib-2.0.so.0 (0x41dcf000)
/lib/ld-linux.so.2 (0x40000000)
libXxf86vm.so.1 => /usr/X11R6/lib/libXxf86vm.so.1 (0x41e4e000)
libXt.so.6 => /usr/X11R6/lib/libXt.so.6 (0x41e53000)

```

مدير التشغيل (init) :

مهمة (init) جعل كل شيء يعمل كما ينبغي من مهامه التأكد من صحة وسلامة أنظمة الملفات وتشغيل شبكة الخادما ، ويقوم بتشغيل برنامج (getty) وهو البرنامج الذي يطلب منك اسم المستخدم وكلمة المرور .

يقوم برنامج (init) بقراءة ملف (/etc/inittab) الذي يرشد البرنامج إلى الأعمال التي يجب القيام بها يتم صياغة الملفات على شكل (bash script) في البداية يتم تشغيل الملف (/etc/rc.d/rc.S) و هو ما يقوم به النظام أثناء بدء التشغيل هنا يتم التأكد من سلامة أنظمة الملفات وتفعيلها وتفعيل السواب (swap) والكثير من الأمور اللازمة ومن ثم يتم الانتقال إلى مستوى التشغيل الافتراضي وهي عبارة عن ملفات مكتوبة على شكل (bash script)

عندما يكون مستوى التشغيل (1) مستخدم واحد يتم تنفيذ الملف (/etc/rc.d/rc.K) أما إذا كان مستوى التشغيل (3) أو (4) يتم تنفيذ الملف (/etc/rc.d/rc.M) ويتم تنفيذ الملف (/etc/rc.d/rc.4) في حالة إذا كان مستوى التشغيل (4) ، ومن بعد ذلك يقوم (init)

بتشغيل (getty) على الطرفيات الظاهرية الستة (افتراضي) وهناك تحسس من (init) لـ (getty) عندما يقف (getty) يتم إعادة تشغيله .

-نظام الملفات:

تم ذكر أن (init) يقوم بالتأكد من صحة وسلامة أنظمة الملفات ويقوم بتمكين الجهاز من الوصول إليها وذلك من خلال البرنامجين (fsck) و (mount) -القرص الصلب :

هو عبارة عن مساحة يمكن الكتابة والقراءة منها ونظام الملفات هو عبارة عن تنظيم معين لهذه العملية ويتم تمثيلها للمستخدم على شكل ملفات داخل مجلدات الخ ، كل ملف يمثل بـ (inode) حيث يسجل عليها لأي ملف (inode) ومتى تم تكوينها وأين محتواها وكذلك مع المجلدات لكن يسجل على (inode) المجلد أين تجد (inode) الملفات داخل المجلد ، لنفرض أن النظام يريد الوصول إلى الملف (home/saleh/asl.txt) أولاً سيتم البحث من (inode) لـ (/) ثم يبحث عن (inode) لـ (home) ثم يبحث عن (inode) لـ (saleh) ثم يبحث عن (inode) لـ (asl.txt) حيث يتم إرشاده إلى محتويات الملف .

عندما يتم تعديل على الملفات يمكن أن يتم تسجيل التغيير في محتوى الملف قبل تحديث (inode) ، إذا فصل التيار الكهربائي في هذه الظروف سيتم عدم مصادقة التغييرات في المحتوى الحقيقي مع التنظيم المتبع في نظام الملفات ، في مثل هذه الحالات يقوم البرنامج (fsck) بالتعرف على الأخطاء ومحاولة إصلاحها .

أمر (mount) يقوم بإدراج نظام الملفات المستهدف إلى البنية التنظيمية في لينكس.

-أشباح النواة :

عندما تستخدم الأمر (ps -auxww) تظهر لك قائمة مقارنة للتالي:

```
root@hathalsal:~# ps -auxww
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
```

```

root    1 0.0 0.0 588 228 ?    S   11:42  0:00 init [3]
root    2 0.0 0.0  0  0 ?    SN  11:42  0:00 [ksoftirqd/0]
root    3 0.0 0.0  0  0 ?    S<  11:42  0:00 [events/0]
root    4 0.0 0.0  0  0 ?    S<  11:42  0:00 [khelper]
root    5 0.0 0.0  0  0 ?    S<  11:42  0:00 [kacpid]
root   27 0.0 0.0  0  0 ?    S<  11:42  0:00 [kblockd/0]
root   28 0.0 0.0  0  0 ?    S   11:42  0:00 [khubd]
root   38 0.0 0.0  0  0 ?    S   11:42  0:00 [pdflush]
root   39 0.0 0.0  0  0 ?    S   11:42  0:00 [pdflush]
root   41 0.0 0.0  0  0 ?    S<  11:42  0:00 [aio/0]
root   40 0.0 0.0  0  0 ?    S   11:42  0:00 [kswapd0]
root  626 0.0 0.0  0  0 ?    S   11:42  0:00 [kseriod]
root  711 0.0 0.0  0  0 ?    S<  11:42  0:00 [reiserfs/0]
root  787 0.0 0.1 1404 444 ?    S<S  11:42  0:00 udevd
root 1185 0.0 0.2 1588 632 ?    Ss  11:43  0:00 /usr/sbin/syslogd
root 1188 0.0 0.2 1548 496 ?    Ss  11:43  0:00 /usr/sbin/klogd -c 3 -x
root 1314 0.0 0.0  0  0 ?    S   11:43  0:00 [pccardd]
root 1395 0.0 0.2 1440 568 ?    S<S  11:43  0:00 /sbin/dhccpd -d -t 10 eth0
root 1468 0.0 0.0  0  0 ?    S   11:43  0:00 [shpchpd_event]
root 1482 0.0 0.0  0  0 ?    S   11:43  0:00 [pciehpd_event]
root 3863 0.0 0.2 1456 564 ?    Ss  11:43  0:00 /usr/sbin/inetd
root 3870 0.0 0.6 3248 1500 ?    Ss  11:43  0:00 /usr/sbin/sshd
root 3936 0.0 0.2 1732 644 ?    S   11:43  0:00 /usr/sbin/crond -l10
daemon 3938 0.0 0.2 1732 696 ?    Ss  11:43  0:00 /usr/sbin/atd -b 15 -l 1
root 3941 0.0 0.2 1420 592 ?    Ss  11:43  0:00 /usr/sbin/acpid
root 3984 0.0 0.2 1468 520 ?    Ss  11:43  0:00 /usr/sbin/gpm -m /dev/mouse -t ps2
root 3987 0.0 1.2 4012 2976 tty1  Ss  11:43  0:00 -bash
root 3988 0.0 0.2 1420 540 tty2  Ss+ 11:43  0:00 /sbin/agetty 38400 tty2 linux
root 3989 0.0 0.2 1420 540 tty3  Ss+ 11:43  0:00 /sbin/agetty 38400 tty3 linux
root 3990 0.0 0.2 1420 540 tty4  Ss+ 11:43  0:00 /sbin/agetty 38400 tty4 linux
root 3991 0.0 0.2 1420 540 tty5  Ss+ 11:43  0:00 /sbin/agetty 38400 tty5 linux
root 3992 0.0 0.2 1420 540 tty6  Ss+ 11:43  0:00 /sbin/agetty 38400 tty6 linux
root 4073 0.0 1.1 6960 2820 ?    S   11:43  0:00 /usr/sbin/smbd -D
root 4094 0.0 0.5 2408 1288 tty1  S+  11:43  0:00 /bin/sh /usr/X11R6/bin/startx
root 4107 0.0 0.2 2216 680 tty1  S+  11:43  0:00 xinit /root/.xinitrc --
root 4108 1.5 7.1 154204 17096 ?  S<  11:43  0:12 X :0
root 4151 0.0 0.5 2400 1236 tty1  S   11:43  0:00 sh /root/.xinitrc
root 4152 0.0 0.5 2416 1308 tty1  S   11:43  0:00 /bin/sh /opt/kde/bin/startkde
root 4174 0.0 4.9 24328 11860 ?    Ss  11:43  0:00 kdeinit Running...
root 4177 0.0 4.3 22964 10540 ?    S   11:43  0:00 dcopserver [kdeinit] dcopserver --nosid
root 4179 0.0 4.9 25348 11904 ?    S   11:43  0:00 klauncher [kdeinit] klauncher
root 4182 0.1 6.9 28924 16684 ?    S   11:43  0:01 kded [kdeinit] kded
root 4189 0.0 6.5 26756 15644 ?    S   11:44  0:00 kxkb [kdeinit] kxkb
root 4190 0.0 2.8 11444 6956 ?    S   11:44  0:00 artsd -F 10 -S 4096 -s 60-m artsmessage -c drkonqi -l 3 -f
root 4197 0.0 7.4 33932 17972 ?    S   11:44  0:00 knotify [kdeinit] knotify
root 4198 0.0 0.1 1408 400 tty1  S   11:44  0:00 kwrapper ksmserver
root 4200 0.0 5.9 25960 14376 ?    S   11:44  0:00 ksmserver [kdeinit] ksmserver
root 4201 0.1 6.8 27500 16368 ?    S   11:44  0:00 kwin [kdeinit] kwin -session
10135cde7c9000110789017700000079240000_1111655625_728002
root 4203 0.0 5.9 26080 14268 ?    S   11:44  0:00 khotkeys [kdeinit] khotkeys
root 4205 0.1 7.0 27092 17004 ?    S   11:44  0:00 kdesktop [kdeinit] kdesktop
root 4207 0.1 7.4 28572 17824 ?    S   11:44  0:00 kicker [kdeinit] kicker
root 4208 0.0 5.2 26112 12504 ?    S   11:44  0:00 kio_file [kdeinit] kio_file file /tmp/ksocket-root/klauchern3tHGa.slave-socket /tmp/ksocket-root/kdesktopOfG8Tb.slave-socket
root 4210 0.0 6.4 26844 15520 ?    S   11:44  0:00 klipper [kdeinit] klipper
root 4214 0.0 7.3 28568 17664 ?    S   11:44  0:00 kmix [kdeinit] kmix -session
10135cde7c9000110844814700000043720008_1111655625_675250
root 4217 0.4 10.6 39256 25464 ?    S   11:44  0:03 konqueror [kdeinit] konqueror --silent
root 4271 0.4 7.2 29816 17488 ?    S   11:53  0:00 konsole [kdeinit] konsole--ls
root 4272 0.1 1.2 4012 2980 pts/1  Ss  11:53  0:00 -bash
root 4304 0.0 0.3 2412 860 pts/1  R+  11:56  0:00 ps -auxww

```

وهي العمليات التي تعمل على جهازك تأتي المعلومات من المجلد (/proc) ستلاحظ أن هناك شي غير عادي ستجد برامج ذات حجم ظاهري (VSZ) مساوي للصفر وكذلك الحجم الحقيقي (RSS) مساوي للصفر هذه البرامج هي أشباح النواة ، وهي كأى برنامج آخر يعطي رقم عملية (PID) لكن هذه البرامج هي جزء من النواة يتم تمييز أشباح النواة بأقواس مربعة الشكل [] ليتم التمييز بينها وبين البرامج الأخرى .

- agetty و الدخول:

(agetty) هو البرنامج الذي يمكنك من الدخول إلى النظام من خلال الطرفيات الظاهرية (virtual terminal) أو من خلال الطرفيات النصية (text terminal) هذا البرنامج الذي يظهر لك أمر الدخول (login prompt) و ينتظر منك إدخال اسم المستخدم ويقوم بالتحقق من اسم المستخدم وكلمة المرور ، (agetty) هو جزء من الحزمة (util-linux) إن رسالة الترحيب التي تظهر فوق أمر الدخول يمكن إعدادها من خلال الملف (/etc/issue) ، يتم تشغيل (agetty) من خلال الملف (/etc/inittab) ويتم التأكد من صحة معلومات الدخول وكلمة المرور من خلال الملفين (/etc/passwd) و (/etc/shadow) .

-سطر الأوامر (bash) :

عندما يتم الدخول إلى الجهاز باسم مستخدم وكلمة مرور صحيحين سيتم توجيهك إلى سطر الأوامر (bash) ستقوم بقراءة أوامرك ومحاولة تنفيذها ليس فقط تنفيذ البرامج أيضاً يمكن القيام بالبرمجة من خلالها وهذا الجمع بين تنفيذ البرامج ولغة برمجة يصبح لديك أداة قوية جداً.

يمكن إعداد خصائص (bash) من خلال الملف (/etc/profile) .

صالح الهذال
hathalsal