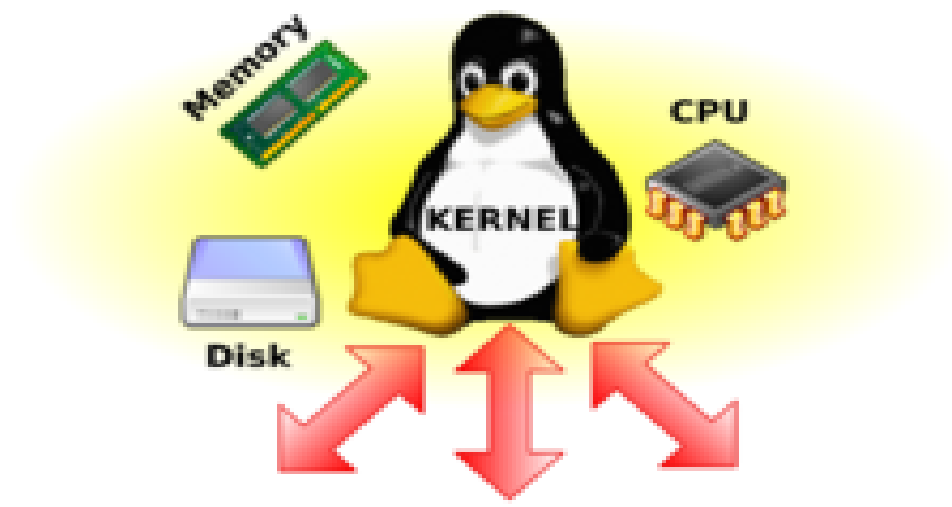
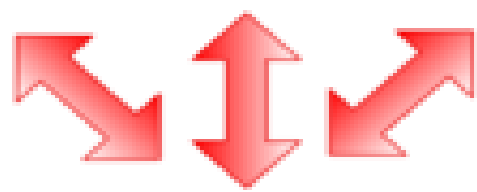
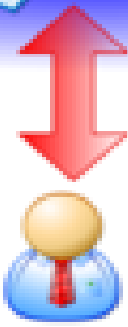


DEVICES



APPLICATIONS



Linux Temel Yapısı

Hardware , Kernel , Shell, Gui nedir ?

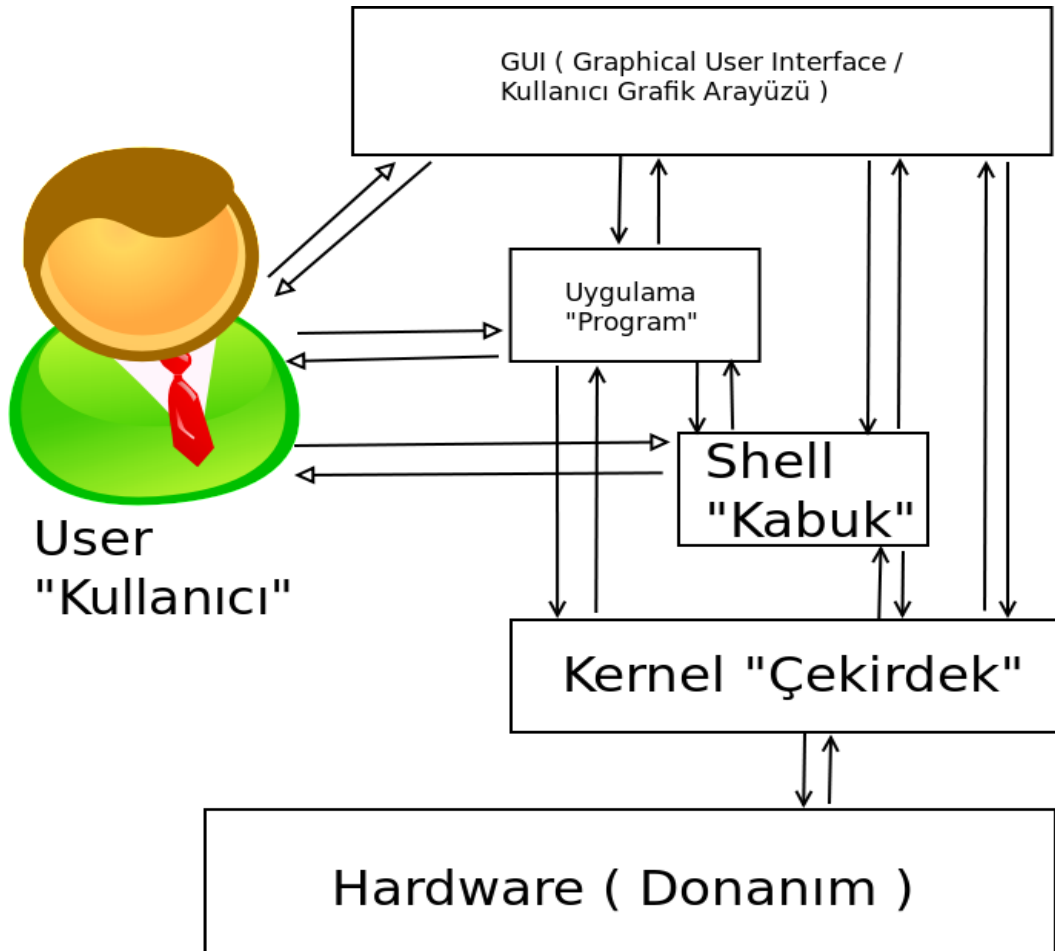
Hardware : Bilgisayarın elektronik/elektromekanik ve mekanik aksamına verilen genel isimdir.

Kernel : Çekirdek, bilgisayarda donanım (hardware) ve yazılım (software) arasındaki bağlantıyı sağlayan arabirime verilen isimdir.

Shell (Kabuk): İşletim sisteminin kerneli ile kullanıcı arasında iletişimi sağlayan arayüzdür , Basitçe Kernelin söylediklerini insan anlayabileceği dile , insanın söylediklerini de kernelin anlayabileceği duruma getiren komut çevirmenidir diyebiliriz.

Program (Uygulama) : Bilgisayarın belirli görevler ve işlemleri yapması için yazılmış parçacıklardır. Görevler Kullanıcı ile Bilgisayar arasındaki görevleri olduğu kadar bilgisayar içerisinde diğer uygulamaların birbiri ile haberleşmesi amacıyla güdebilir.

GUI (Graphical User Interface / Kullanıcı Grafik arayüzü) : Klavyeden komut yazmak yerine , görsel olarak işlemleri görüp kullanabilmemiz için dizayn edilmiş arayüzdür. Örneğin yazmış olduğumuz yazıyı kayıt etmek için komut vermek yerine "Disket " Resmine basmak gibi. Görsel olarak yapmış olduğumuz işlem Uygunsa Shell (Kabuk) 'e Değilse direk olarak kernele komut olarak tercüme edilerek iletilir.



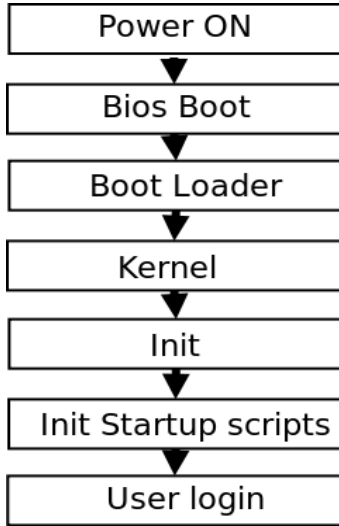
Linux nedir ?

Linux, temel olarak Finlandiya Üniversitesinde öğrenci olan Linus Torvalds tarafından 1991 yılında başlatılmış bir Kernel (Çekirdek) projesidir. Çekirdek içeriği bir çok gönüllü tarafından geliştirilmiştir. **Linux bir işletim sistemi değil sadece çekirdektir.** Kısacası sistemin açılması için önyükleyici (Bootloader) , Shell (Kabuk) , Desktop dahil olmak üzere Kernel hariç tüm uygulamalar değişik kullanıcı ve gruplar tarafından geliştirilmekte olup genel olarak bir çok muadile sahiptirler .

Not : linux Türkçe yazılım şekliyle " linuks" diye okunur . Sadece Türk kullanıcılar tarafından Linüks , linaks , Laynaks , Laynıs şeklinde yanlış telafuzları mevcut olup linux camiasında hoş karşılanmaz.

İşletim Sistemi nedir ?

İşletim sistemi, bilgisayar donanımının doğrudan denetimi ve yönetiminden, temel sistem işlemlerinden ve uygulama programlarını çalıştırmaktan sorumlu olan sistem yazılımıdır.



Linux çekirdeğini boot etmek için grub,lilo,syslinux, loadlin gibi bir çok yazılım kullanabiliriz. Her biri birbirinden bağımsız farklı uygulamalardır .

Linux kerneli ile kullanabileceğimiz tüm programların birden çok muadili mevcuttur . Projenin durması, Paralı olması vb durumlarda diğer program kullanılabilir.

Linux kerneli bu programlar için kullanılacak tek bedava ve açık kaynak kodlu kernel değildir. BSD türevleri , Hurd , Opensolaris kullanılacak diğer seçeneklere örnek olarak verilebilir.

Linux dağıtımlarının kronolojisi

Dağıtım kavramı, özgür yazılım felsefesinin çok alternatifli dünyasının bir sonucu olarak ortaya çıkmış, Linux'a özgü bir terimdir. Linux Dağıtımı ile vurgulanan aslında tam bir işletim sistemi olarak çalışır hale gelmiş bir özgür yazılım projeleri topluluğudur.

Bunun için sistemi kullanacak kullanıcı ya da sistemin üstleneceği göreve uygun olarak Linux çekirdeği ve birlikte binlerce özgür yazılım projesi ürünü kullanılarak birlikte çalışacak şekilde bir sistem oluşturulur.

Kullanım yaygınlıkları ve GNU/Linux dünyasına katkılarıyla öne çıkan bazı dağıtımlar vardır: Pardus, Ubuntu, Debian, Slackware, Mandriva Linux (Mandrake), Red Hat, Fedora Core, SUSE bunlardan bazılarıdır.

Bir Linux sistemi bilgisayara kurulmadan CD-ROM veya USB Bellek üzerinden çalışabilecek şekilde de tasarlanabilmektedir. Bir dağıtımın bu şekilde kullanılabilen sürümüne "Çalışan sürüm" (İngilizce: live version), kullanıldığı medyalara göre de "Çalışan CD" (Live CD) veya "Çalışan USB" (Live USB) denilmektedir. Bazı dağıtımlar ise sadece çalışan sürümü ile yayınlanmaktadır. Bu alanda en çok bilinen dağıtım Debian temel alınarak hazırlanan Knoppix ve bir dağıtım temel almamış olan Slax'dır. Ayrıca Mandrake temel alınarak hazırlanan Turkix (yerli) sayılabilir.

Kullanıcılar açısından dağıtımları birbirinden ayıran en önemli faktörler dağıtımların kullandıkları paket yönetim sistemleri, masaüstü ortamları ve yönetim araçlarıdır. Paket yönetim sistemleri yazılımların kurulup kaldırılması için kullanılırken, masaüstü ortamları kullanıcıların sistem gereksinimleri ve ihtiyaç duydukları masaüstü araçları ve konforu açısından önemlidir. Örneğin bazı sunucu dağıtımlarında hiçbir masaüstü ortamı bulunmazken Ubuntu gibi ev-ofis amaçlı bazı dağıtımların farklı masaüstü seçenekleri ile kullanılabilme imkânı bulunmaktadır.

Dağıtım geliştirme modelleri de paket yönetim sistemleri açısından temelde üç grupta incelenebilir. Bu üç grup ikili paket yönetimi, kaynak paket yönetimi ve melez paket yönetimidir. İkili yöntemde paketler yazılımcıların kaynak kodlarından ilgili dağıtımın özelliklerine uygun şekilde derlenerek ikili hale dönüştürülür ve paket depolarından kullanıcılara sunulur. Kaynak paketlerde ise yazılımlar kaynak kodları ile depolarda bulunur, kullanıcılar derleme işlemini kendi bilgisayarlarında yaparlar. Bu şekildeki pek çok dağıtımın paket yönetim sisteminin yazılımların dağıtıma uygun derlenmesi için dağıtıma özgü bir derleme ve inşa sistemi vardır. Melez yöntemde ise bu iki yöntem de kullanılır.

Türkiye'de Fedora üzerinde geliştirilen Turkuaz, Gelecek ve Mandrake üzerinden geliştirilen Turkix, Armador OS 2006 gibi projelerden sonra TÜBİTAK bünyesinde Pardus isimli bir dağıtım geliştirilmektedir. Pardus her Bilgisayar okuryazarı'nın kolayca kurup kullanabilmesi amaçlanarak geliştirilmektedir. Bu nedenle kurulumundan yönetimine kadar pek çok araç ve teknoloji Pardus geliştiricileri tarafından özel bir proje olarak geliştirilmektedir.

Finansal güç açısından; Novell tarafından satın alınan ve milyar dolarlık bir bütçeye sahip Suse ve bunun yanı sıra Fedora Core'un da sahibi Red Hat linux dağıtımları ekonomik olarak en büyük özgürlüğe sahiptir. Aynı zamanda, Mandrake'nin (yeni adıyla Mandriva) arkasında ise dünya devi Vivendi-Universal

şirketin bulunmaktadır. Kullanımı ve yüklenmesi en rahat olan Linux dağıtımlarından bir tanesi de Ubuntu'dur ve bu dağıtıma Canonical şirketi destek olmaktadır. Bu dağıtım içerisinde bulunan araçlar sayesinde kolay ve sorunsuz kullanım desteği sağlamaktadır.

Mepis, Ubuntu, Yoper, Knoppix, Libranet, Linspire, Xandros ve Adamantix gibi birçok linux dağıtımında da baz olarak kullanılan Debian; Google da başta olmak üzere birçok web sitesi tarafından başlıca kullanılmaktadır.

Linux from scratch , Rocklinux gibi projeler kullanılarak herhangi bir dağıtım kullanmadan kendi linux işletim sisteminize sahip olabilirsiniz.

Ref : wikipedia http://tr.wikipedia.org/wiki/Linux_da%C4%9F%C4%B1t%C4%B1mlar%C4%B1

Paket yöneticileri :

Modern işletim sistemlerinin tümü, yeni yazılımların sistem üzerine birer "paket" olarak yüklenebilmesine imkan veren düzenekler içermektedirler. Bir paket, yazılım ile ilgili tüm dosyaları içerdiği gibi yazılıma ilişkin dosyaların hangi dizinlere ne biçimde yerleştirileceğini ve mevcut sistem ayarlarında ne gibi değişiklikler yapılması gerektiğine ilişkin bilgileri de içerir.

Yeni nesil UNIX/linux'ların hemen hepsi bir paket yöneticisi içermektedir. Paket yöneticisi temel olarak yeni yazılımların kolayca yüklenmesini, kurulu paketlerin bütünlüğünün sınanmasını ve artık gereksinim duyulmayan paketlerin sistemden silinmesini ve güncellenmelerini kolaylaştıracak bir yazılımdır. Bir sisteme paket halinde kurulacak yazılımların büyük bir bölümü /usr dizinine kurulmak isteyecektir. Bu nedenle paket yöneticisinin sistem yöneticisi dışında bir kullanıcı tarafından kullanılması ve yazılım kurulması genellikle mümkün olmaz.

Genel linux dağıtımları , paket yöneticileri ve dosya uzantıları.

Dağıtım	Paket yöneticisi	Uzantı
Redhat	Rpm / yum	*.rpm
Suse	Rpm / yast	*.rpm / *.srpm
Debian	apt	.deb

Linux kullanıcıları :

Linux/Unix sistemlerde Administrator kullanıcısının muadili (Super User) root (kök) kullanıcısıdır . Root kullanıcısı ilk sistemde ilk oluşturulan şey olup kurulumda tüm dosyaları oluşturan kullanıcı olduğu için root (kök) olarak adlandırılmıştır. Root kullanıcısının tüm sistem üzerinde hemen hemen sınırsız haklara sahip olmasından dolayı devamlı olarak root user'i ile işlem yapmak güvenlik açısından tavsiye edilmez.

Su :

su komutu " Super User" (Süper kullanıcı) kelimelerinin kısaltılmasıdır. Sistem üzerinde gerekli haklara sahip bir kullanıcı (doğru grup üyeliği ve yetkilendirme) su komutunu vererek istediği işlemleri root kullanıcısı olarak yapabilir. Su komutu root kullanıcısı hariç kullanıcı hesaplarına erişmek içinde kullanılır. Örneğin http user'i ile işlem yapmak için " su http" komutu verilip şifre doğru girilmesi halinde o kullanıcı olarak komut çalıştırılır.

Sudo :

Sudo güvenlik amacıyla geliştirilmiş olup temel olarak "su" komutunun yaptığı görevi yapsada ana farkı şifre sormamasıdır. Sisteme login olan kullanıcı gerekli ayarlamalar doğru olarak yapılandırılıp kendine hak tanındıysa sudo komutu ile root veya başka kullanıcı hesabına ulaşarak işlem yapabilir.

Dosya izinleri :

```
alper@ubuntu:~$ ls -l | grep test
drwxr-xr-x 2 alper alper 4096 2010-02-08 09:53 test
alper@ubuntu:~$
```

r - Read (okuma izni)
w - write (yazma izni)
x - Execute (çalıştırma izni)

bu dizilime göre ;
ilk uclu -> dosya sahibi izinlerini
ikinci ---> dosyanın grubunda bulunanların izinlerini
ucuncu -> diğerlerinin izinlerini gösterir

yukarıdaki örneği inceleyecek olursak :

Bu dosya alper kullanıcısına ve alper grubuna aittir.

drwxr-xr-x

d = bu bir dizindir (Directory)

Dosya sahibi bu dosya için okuma, yazma , çalıştırma haklarına sahiptir.
(Read , Write , execute)

Grup üyeleri okuya bilir, yazamaz , çalıştırabilir
Diğer kullanıcılar okuya bilir, yazamaz , çalıştırabilir

Linux Dosya Sistemi Yapısı

Bir işletim sisteminin dosya sisteminin performansı, hızı ve buna benzer çeşitli özellikleri sistemin verimli ve kararlı çalışması üzerinde doğrudan etkisi vardır. Linux dizin yapısının geliştirilmesinde *FSSTND* (file system standard) grubunun çalışmalarının payı çok büyük oldu. Programcılardan oluşan grup, standartların belirlenmesinde ve uygulama aşamasında Linux ile ilgilenenlere yardım etti. 1993 yılında Olaf Kirsh, Linux haber grubuna yazdığı bir e-postasında Linux dosya sistemi üzerindeki çalışmaların tamamlandığını haber verdi.

Linux dizin yapısında bazı dizinlerin işlevi birbiriyle aynıdır. Bu durum, özellikle birbirini takip eden iki Linux sürümünde belirginleşir. Örnek olarak bir dağıtımda `/usr/bin` dizini altında yer alan dosya, diğer sürümde `/bin` altına yerleştirilebilir.

Dosya sistemleri ve dizinler paylaşımlı olarak da kullanılabilir. Birden fazla kişisel bilgisayarın bağlandığı ağda, disk alanından yer kazanmak için bir makina *sunucu* (ana makina) olarak tayin edilir. Ağ üzerindeki diğer makineler de sunucu üzerindeki diski paylaşır. Buna örnek olarak kullanıcı ev dizinlerinin yer aldığı `/home`, çalıştırılabilir dosyaların bulunduğu `/usr`, e-posta ve haber grubu bilgilerinin bulunduğu `/var/spool/mail` ve `/var/spool/news` dizinleri verilebilir.

Kök dizini, kendisine bağlı diğer tüm dizinleri de içerdiğinden, Linux dosya sisteminde önemli bir yere sahiptir. Linux açılırken önce kök dizini `/etc/fstab` dosyasına uygun şekilde bağlanır. Diğer dosya sistemlerinin onarımı ve kontrolü için gerekli olan `fsck` programları, bu dizinde bulunmalıdır. Benzer şekilde yedekleme için gerekli olan `tar`, `zip`, `compress` gibi arşiv programlarına da kök dizin altından erişilebilmeli, açılış esnasında hafızaya yüklenen çekirdek de kök dizininde yer almalıdır. Bir programı yerine koymadan önce bazı "dengelerin" gözönünde tutulmasında yarar vardır. Programlanan her yazılımı kök dizinine koyarsanız, bir süre sonra bu dosya sisteminin şiştiğini görürsünüz. Kök dizinini başlangıçta küçük (20-30Mb kadar) tutmak, geliştirilen programları daha önceden belirlenen başka bir dosya sistemi altına koymak akıllıca olur. Bu yol, ağ üzerinden genellikle paylaşımı mümkün olmayan kök dizininin her makinadaki sabit diskte mümkün olan en az yer işgal etmesini sağlar.

Sistemdeki önemli dizinlere göz atarsak,

/bin

Sistemin açılışı ve kontrolü için gerekli komutlar. Hem kullanıcıların, hem de sistem görevlisinin kullanabileceği dosyalar (kök dizinde ise fazla şişmemesi koşuluyla) buraya atılabilir. Sadece root kullanıcının ihtiyaç duyacağı init, getty, updatedb gibi programlar /sbin veya /usr/sbin'de durabilir. Bu dizinde bulunan dosyalara örnek olarak cat, chgrp, chown, date, dd, df, ln, mkdir, mount, ps, rm, sh, su, sync ve umount verilebilir.

/dev

G/Ç dosyaları. Linux çekirdeğinde desteklenen her aygıta ait dosya /dev dizini altında bulunur. Kurulum anında bu dosyalar yerine yerleştirilir, bu dosyaların silinmesi durumunda /dev/MAKEDEV ile tekrar yaratılabilirler.

/etc

Sistem yapılandırma dosyaları. Bu dizinde çalıştırılabilir dosyalar bulunmamalıdır.

skel

Buradaki dosyalar, kullanıcı hesabı açıldığında kullanıcının ev dizinine kopyalanır.

rc.d

Bu dizinin içinde, init sürecinin başvurduğu yapılandırma dosyaları vardır. Bunlara "rc dosyaları" da denir.

passwd

Kullanıcı veritabanı

fstab

Linux'un açılışı esnasında bindirilecek dosya sistemleri burada listelenir.

group

passwd'e benzer şekilde kullanıcıların gruplarını tutar.

inittab

init daemon için yapılandırma dosyası

motd

Kullanıcı sisteme girdikten sonra ekranına basılması istenen mesaj burada tutulur.

profile

Kullanıcı sisteme girdiği zaman çalıştırılan dosya (csh ve sh türevi kabuklar için)

shells

Sistemde kullanılacak kabuk isimleri burada tutulur.

login.access

login komutu için yapılandırma dosyası. Sisteme girişi kullanıcı bazında sınırlamak için kullanılır.

/home

Kullanıcılara ayrılmış dizin. Başka şekilde ayarlanmamış ise, açılan her hesaba ait kullanıcı, burayı kullanır. Büyük sistemlerde, bu kısım alt parçalara ayrılabilir (/home/ftpadm , /home/ogrenci gibi)

/lib

Kütüphane dosyaları.

/mnt

Geçici mount edilen dosya sistemleri. Sadece bu iş için kullanıldığından sistem görevlisine zaman kazandırır.

/proc

Süreç kontrollerini ve diğer sistem bilgilerini tutan dosya sistemi. Bu dosya sistemi aslında disk üzerinde yer kaplamaz, tüm dosyalar çekirdeğin bir uzantısı sayılabilir.

cpuinfo

işlemci modeli, tipi ve performansını bildirir.

devices

Halihazırda çalışan çekirdek içinde desteği bulunan aygıt sürücülerini listeler.

dma

Hangi dma kanallarının kullanıldığını belirtir.

filesystems

Halihazırda çalışan çekirdek içinde desteği bulunan dosya sistemlerini listeler.

interrupts

Hangi kesintilerin kullanımda olduğunu söyler.

iports

Halen hangi giriş/çıkış iskelelerinin kullanıldığını belirtir.

kcore

Sistem hafızasının görüntüsü

/root

Sistem görevlisinin ev dizini. Mümkünse bu dizini sistemdeki diğer kullanıcıların görmeyeceği şekilde ayarlayın.

/sbin

Hayati sistem komutları. Bir zamanlar bu dosyalar /etc dizini altında yer alıyorlardı. Sadece sistem görevlisinin ihtiyacı olan komutlar, /sbin veya /usr/sbin içinde bulunur.

/tmp

Geçici dosyaların koyulduğu dizin. Belirli zaman aralıklarında temizlenmelidir.

/usr

Diğer önemli sistem dosyalarını tutar. Bu bölüm genelde en kalabalık dizindir, zira yeni kurulan tüm programlar buraya konulur.

X11R6

X Pencere sistemi bilgileri tutulur.

doc

Belge ve dökümanlar, genellikle HOWTO ve FAQ dosyaları.

lib

Bazı kütüphaneler

man

Man dosyaları

src

Bazı kaynak dosyaları ve linux çekirdeğini oluşturan kodları (/usr/src/linux) içeren dizin.

sbin

Kök dosya sisteminde yer alması gerekmeyen çalıştırılabilir sistem görevlisi dosyaları

/var

Sürekli deęişen sistem bilgileri burada tutulur. İstisnalar dışında dięer makinalarla paylaştırılmaz.

adm

Sistem yönetimini ilgilendiren kayıtlar

preserve

Sistemin göçmesinden sonra zarar görmesi mümkün dosyaların kaydedildięi yer.

spool

Sonra işlenecek olan veriler buraya atılır (e-posta gibi)

log

Sistem logları

Linux'ta herşey bir dosyadır.**İnit Betikleri (initscripts)****İçerik:**

- * Çalışma Seviyeleri(run levels)
- * rc-update ile Çalışma
- * Servisleri Ayarlama
- * İnit Betikleri Yazma
- * Çalışma Seviyesi Davranışını Deęiştirme

Çalışma Seviyeleri(run levels)**Sisteminizi Önyükleme**

Sisteminizi önyüklediğinizde, birçok kayan yazıyı farkedersiniz. Eğer dikkatlice takip ederseniz, bu yazıların sisteminizi her açtığınızda aynı olduğunu görürsünüz. Bu işlemler dizisinin tamamına açılış dizisi denir ve (az ya da çok) statik olarak belirtilir.

Önce, açılış yöneticisi, ayar dosyasında belirttiğiniz kerneli hafızaya yükler, sonra da işlemciye kerneli çalıştırmasını söyler. Kernel yüklenip çalıştırıldığında, kernele özgü yapı ve görevleri başlatır ve init sürecini çalıştırır.

Bu süreç, daha sonra (/etc/fstab içinde tanımlı) bütün dosya sistemlerinin bağlanıp kullanıma hazır olduğundan emin olur. Sonra da /etc/init.d içinde bulunan, sisteminizin başarılı bir şekilde açılması için gerekli servisleri başlatacak çeşitli betikleri çalıştırır.

Son olarak, bütün betikler çalıştırıldığında, init süreci, agetty adındaki özel süreci bağlayarak terminalleri (genellikle ALT-F1, ALT-F2, ... altındaki sanal konsolları) aktifleştirir. Bu süreç daha sonra login'i çalıştırarak terminallere bağlanabileceğinizden emin olur.

Açılış Betikleri (Init Scripts)

Burada `init` , `/etc/init.d` , içindeki betikleri rastgele çalıştırmaz. Hatta `/etc/init.d` içindeki her betiği değil, sadece çalıştırması söylenenleri çalıştırır. Hangi betikleri çalıştırması gerektiğine `/etc/runlevels` içine bakarak karar verir.

Önce `init` , `/etc/runlevels/boot` içinde sembolik bağlantısı olan `/etc/init.d` betiklerini çalıştırır. Genellikle, betiklere alfabetik sırada başlar, ama bazı betikler bağımlılık bilgisi içerir ve çalışmadan önce başka hangi betiklerin çalıştırılması gerektiğini sisteme söyler.

`/etc/runlevels/boot` içinde referansı bulunan her betik çalıştırıldıktan sonra `init` , `/etc/runlevels/default` içinde sembolik bağlantısı olan betikleri çalıştırır. Tekrar, hangi betiğin önce çalışacağına, düzgün bir başlangıç süreci sağlamak için içerisinde bağımlılık bilgisi içermediği sürece alfabetik sırayı kullanarak karar verir.

Init Nasıl Çalışır ?

Elbette `init` bunların hepsini kendisi düşünmez. Hangi işlemlerin yapılacağını belirten bir konfigürasyon dosyasına ihtiyaç duyar. Bu konfigürasyon dosyası `/etc/inittab`'dir.

Eğer size biraz önce açıkladığımız açılış dizisini hatırlarsanız, `init`'in ilk işi dosya sistemlerini bağlamaktır. Bu, `/etc/inittab` içindeki şu satır ile belirtilmiştir:

Kod Listeleme 1.1: `/etc/inittab` içindeki sistem başlangıcı satırı

```
si::sysinit:/sbin/rc sysinit
```

Bu satır `init`'e, sistemi başlatmak için `/sbin/rc sysinit` komutunu çalıştırması gerektiğini söyler. `/sbin/rc` betiği, başlangıç işlemlerini halleder, bu yüzden `init`'in çok iş yapmadığını -- sistem başlangıç işlemlerinin yapılması işini başka süreçlere aktarır.

İkinci olarak `init` , `/etc/runlevels/boot` içinde sembolik bağlantısı bulunan bütün betikleri çalıştırır. Bu, aşağıdaki satır ile tanımlanmıştır:

Kod Listeleme 1.2: Sistem başlangıcı, devamı

```
rc::bootwait:/sbin/rc boot
```

Yine `rc` betiği, gerekli işlemleri gerçekleştirir. Dikkat ederseniz, `rc`'ye verilen parametre (`boot`), `/etc/runlevels` altındaki `altdizin` ile aynıdır.

Şimdi `init` , hangi çalışma seviyesinin kullanılacağını görmek için konfigürasyon dosyasını kontrol eder. Buna karar vermek için, `/etc/inittab`'deki aşağıdaki satırı okur:

Kod Listeleme 1.3: `initdefault` satırı

id:3:initdefault:

Bu durumda çalışma seviyesi (runlevel) 3'tür. Bu bilgiyi kullanarak init, 3 numaralı çalışma seviyesi için neyi çalıştırması gerektiğini kontrol eder:

Kod Listeleme 1.4: Çalışma seviyeleri tanımlamaları

```
l0:0:wait:/sbin/rc shutdown
l1:S1:wait:/sbin/rc single
l2:2:wait:/sbin/rc nonetwork
l3:3:wait:/sbin/rc default
l4:4:wait:/sbin/rc default
l5:5:wait:/sbin/rc default
l6:6:wait:/sbin/rc reboot
```

TSeviye 3'ü tanımlayan satır, servisleri çalıştırmak için tekrar rc betiğini (default argümanını ile) kullanır. Tekrar, rc'nin argümanının /etc/runlevels içindeki altdizin ile aynı olduğuna dikkat edin.

rc bittiğinde, init hangi sanal konsolları aktifleştirmesi ve her konsolda hangi komutları çalıştırması gerektiğine karar verir:

Kod Listeleme 1.5: Sanal konsol tanımlaması

```
c1:12345:respawn:/sbin/agetty 38400 tty1 linux
c2:12345:respawn:/sbin/agetty 38400 tty2 linux
c3:12345:respawn:/sbin/agetty 38400 tty3 linux
c4:12345:respawn:/sbin/agetty 38400 tty4 linux
c5:12345:respawn:/sbin/agetty 38400 tty5 linux
c6:12345:respawn:/sbin/agetty 38400 tty6 linux
```

Çalışma Seviyesi Nedir?

Init'in hangi çalışma seviyesini aktifleştireceğine karar vermesi için bir numaralandırma yöntemi kullandığını gördünüz. Bir çalışma seviyesi, sisteminin hangi durumda çalıştığıdır ve bir çalışma seviyesine girdiğinizde çalıştırılması gereken betik topluluğunu (çalışma seviyesi betikleri ya da init betikleri) içerir.

Linux'ta, yedi çalışma seviyesi tanımlanmıştır: üç dahili çalışma seviyesi ve dört kullanıcı tanımlı çalışma seviyesi. Dahili çalışma seviyeleri sysinit, shutdown ve reboot'tur ve isimlerinin ima ettiği işleri yaparlar: sistemi başlat, sistemi kapat, sistemi yeniden başlat.

Kullanıcı tanımlı çalışma seviyeleri de /etc/runlevels içinde altdizini bulunan boot, default, nonetwork ve single'dır. boot çalışma seviyesi, diğer tüm çalışma seviyelerinin kullandığı, sisteme gereken tüm servisleri çalıştırır. Geri kalan üç çalışma seviyesi de, çalıştırdıkları servislere göre farklılık gösterirler: default, günlük yapılan işler için kullanılır; nonetwork, ağ bağlantısı gerekmediğinde kullanılır; single ise sistemi onarmak istediğinizde kullanılır.

Init Betikleri ile Çalışma

rc sürecinin başlattığı betiklere init betikleri denir. /etc/init.d içindeki her betik start, stop, restart, pause, zap, status, ineed, iuse, needsme, usesme veya broken argümanları ile çalıştırılabilir.

Bir servisi (ve bağlı tüm servisleri) başlatmak, durdurmak ya da yeniden başlatmak için, start, stop ve restart kullanılır:

Kod Listeleme 1.6: Postfix'i başlatma (postfix mail servisi örnek olarak kullanılmıştır)

/etc/init.d/postfix start

Not: Sadece belirtilen servise ihtiyaç duyan servisler durdurulur yada yeniden başlatılır. Diğer bağımlı servislere (servisi kullanan ama ihtiyaç duymayanlara) dokunulmaz.

Eğer bir servisi durdurmak istiyorsanız, ama ona bağlı servisleri durdurmak istemiyorsanız, pause argümanını kullanabilirsiniz:

Kod Listeleme 1.7: Postfix'i durdurma ve bağlı servisleri çalışır durumda bırakma

/etc/init.d/postfix pause

Eğer bir servisin durumunu öğrenmek istiyorsanız (başlamış, durmuş, bekletilmiş), status argümanını kullanabilirsiniz:

Kod Listeleme 1.8: Postfix'in durum bilgisi

/etc/init.d/postfix status

Eğer durum bilgisi size servisin çalıştığını gösteriyorsa, ancak öyle olmadığını biliyorsanız, servisin durum bilgisini zap argümanı ile "durdurulmuş" konuma sınırlayabilirsiniz:

Kod Listeleme 1.9: Postfix'in durum bilgisini sınırlama

/etc/init.d/postfix zap

Servisin hangi bağımlılıklara sahip olduğunu öğrenmek için iuse ve ineed argümanlarını kullanabilirsiniz. ineed ile servisin çalışması için hangi servislerin bulunması gerektiğini görebilirsiniz. Diğer taraftan, iuse ile servis tarafından kullanılabilen, ancak çalışması için bulunması zorunlu olmayan servislerin listesini görebilirsiniz.

Kod Listeleme 1.10: Postfix'in ihtiyaç duyduğu servislerin listesini isteme

/etc/init.d/postfix ineed

Benzer olarak, servise ihtiyaç duyan servislerin (needsme) ve servisi kullanabilen servislerin (usesme) listesini öğrenebilirsiniz:

Kod Listeleme 1.11: Postfix'e ihtiyaç duyan servislerin listesini isteme

/etc/init.d/postfix needsme

Son olarak, servis için gerekli olan kayıp servislerin listesini de öğrenebilirsiniz:

Kod Listeleme 1.12: Postfix'in ihtiyaç duyduğu kayıp servislerin listesini isteme

/etc/init.d/postfix broken

X window

X Window, istemci/sunucu mimarisinde görüntü donanımları (ekran, fare, klavye) ile masaüstü ortamı arasında bir arayüz oluşturur. Donanımlara erişmek için onun üstünde çalışacak yazılımlara bir arayüz sağlar. Aslında bir sunucu olduğu için, grafik programları ağ üzerinden de kullanabilirsiniz. Birçok mimariyi desteklediğinden, heterojen platformlar arasında köprü görevi görür. İki ya da daha fazla fiziksel görüntü aygıtına tek bir büyük sanal görüntü aygıtı olarak görüntü verebilir. (Ref : Doruk fişek)

Masa üstü ortamları (Desktop Environment)

Pencere yöneticisi ile bütünleşik birçok programdan oluşan bütün bir masaüstü sunar.

Tüm programlar ortak kütüphaneler kullanılarak yazıldığından uyumluluk yüksektir.

Masaüstü ortamlarının kendileriyle gelen programlar dışında çıkmaya başladığınızda hantal birer pencere yöneticisine dönüşürler. KDE, GNOME, Xfce, Ixde ,XPde, ... (Ref : Doruk fişek)