1. Linux Nedir

Türkiye Linux Kullanıcıları Grubu, http://www.linux.org.tr v0.1 Temmuz 1997

Chapter 1

Linux Nedir ?

Linux, serbestçe dağıtılabilen, çokgörevli, çok kullanıcılı UNIX işletim sistemi türevidir. Linux, İnternet üzerinde ilgili ve meraklı birçok kişi tarafından ortak olarak geliştirilmekte olan ve başta IBM-PC uyumlu kişisel bilgisayarlar olmak üzere birçok platformda çalışabilen ve herhangi bir maliyeti olmayan bir işletim sistemidir.

UNIX 70'li yılların ortalarında büyük bilgisayarlar üzerinde çok kullanıcılı bir işletim sistemi olarak geliştirilmiştir. Zaman içerisinde yayılmış ve birçok türevi ortaya çıkmıştır. UNIX ismi UNIX Research Laboratories INC şirketinin tescilli markası olduğundan dolayı birçok şirket, aynı temele dayanan işletim sistemleri için değişik isimler kullanagelmişlerdir. Örnek olarak

- Hewlett-Packard HP-UX
- IBM AIX
- Sun Microsystems SunOS

kullanmaktadırlar. Bugün kişisel bilgisayarlardan süper bilgisayarlara kadar biçok bilgisayar için yazılmış bulunan UNIX türevleri mevcuttur. Ne var ki bu türevlerin çoğu gelişimi belirli bir noktada durmuş ve yüksek fiyatla satılan ticari yazılımlardır.

Linux, temel olarak Finlandiya Üniversitesinde öğrenci olan Linus Torvalds'ın ve İnternet üzerinde meraklı bir çok yazılımcının katkıları ile geliştirilmiştir. Linux gelişimi açık bir şekilde yapılmaktadır. Bunun anlamı, işletim sisteminin her aşaması açık olarak İnternet üzerinde yayınlanmakta, dünyanın dört bir yanında kullanıcılar tarafından test edilmekte, hataları ve eksiklikleri tesbit edilerek düzeltilmekte ve geliştirilmektedir. Zaman zaman bu deneme aşamaları belirli bir noktada durdurulur ve güvenilir bir işletim sistemi sunulup, geliştirme için ayrı bir seriye devam edilir. Geliştirmede yer alan bu açıklık Linux'un en büyük avantajlarından biridir. Gelişimi evrimseldir, hatalar anında kullanıcılar tarafından tesbit edilip rapor edilmekte ve birçok kişinin katkısıyla düzeltilmektedir. Bazı işletim sistemi sürümleri saatler içerisinde güncellenebilmektedir.

Linux, Andy Tannenbaum tarafından geliştirilmiş olan Minix işletim sistemine dayanmaktadır. Linus Torvalds boş zamanlarında Minix'ten daha iyi bir Minix işletim sistemi yaratmak düşüncesiyle 1991 Ağustos sonlarında ilk çalışan Linux çekirdeğini oluşturdu. 5 Ekim 1991 tarihinde 0.02 sürümü Linux ilk defa tanıtıldı. Linus, comp.os.minix haber grubuna gönderdiği yazıda yeni bir işletim sistemi geliştirmekte olduğunu ve ilgilenen herkesin yardımını beklediğini yazmıştı. İşletim sisteminin çekirdeği için verilen numaralar kısa sürede bir standart kazandı. a.x.y seklinde belirtilen çekirdek türevlerinde y bulunulan seviyeyi, x gelişim aşamasını göstermektedir. Tek sayılı x'ler geliştirme aşamalarını çift sayılı x' ler ise güvenilir Linux çekirdeklerini göstermektedirler. a ise değişik Linux sürümlerini belirtir. Bu yazının hazırlandığı Ağustos 1997 içerisinde en son güvenilir (kararlı) Linux çekirdeği 2.0.30, en son gelişim aşamasındaki çekirdek ise 2.1.47'dir.

Linux gerçekten son yıllarda hızlı bir gelişme göstermiş, çesitli ülkelerden birçok kullanıcıya erişmiş ve yazılım desteği günden güne artmıştır. Değişik kuruluşlar Linux sistemi ve uygulama yazılımlarını biraraya getirerek dağıtımlar oluşturmuşlar ve kullanımını yaygınlaştırmışlardır.

1.1 Linux'un Desteklediği Donanımlar

- Linux şu anda başta IBM-PC uyumlu kişisel bilgisayarlar olmak üzere Apple, Atari ve Amiga gibi 68000 tabanlı bilgisayarlar üzerinde, Sun Sparc işlemcili iş istasyonları, Alpha işlemcili kişisel bilgisayarlar, MIPS, PowerPC, HP PA-RISC ve ARM mimarilerinde çalışmaktadır.
- IBM uyumlu kişisel bilgisayarlar üzerinde 80386 ve üzeri (80486 80586 Pentium PentiumPro ve türevleri) değişik üreticilerin işlemcileri ile sorunsuz olarak çalışmaktadır. 80286 ve 8086 işlemcili bilgisayarlar için sınırlı kabiliyette Linux uygulamaları mevcuttur.
- PCI, VESA, ISA ve MCA mimarilerinde her türlü anakartı desteklemektedir.
- Teorik olarak 4 Gbyte'a kadar RAM desteklenmektedir.
- AT uyumlu diskler (IDE, EIDE ve 16 bitlik MFM,RLL veya ESDI) desteklenmektedir. Kontrol kartına uyumlu destek bulunduğu sürece SCSI diskler ve diğer cihazlar desteklenmektedir.
- IDE-ATAPI CD-ROM sürücüleri, ve bazı özel CD-ROM kontrol kartları desteklenmektedir.
- Metin ekranlarda CGA, EGA, VGA, Hercules veya uyumlu kartlar desteklenmektedir. X Window ortamında genel VGA ve SVGA uyumlu kartlar ve S3, ET4000, 8514/A, ATI MACH8, ATI MACH32 gibi birçok görüntü kartı desteklenmektedir.
- Birçok 10 ve 100 Mbit ethernet kartı, ISDN, ATM, FDDI, SLIP, CSLIP, PPP desteği verilmektedir.
- Başta SoundBlaster, Gravis Ultrasound olmak üzere birçok ses kartı desteklenmektedir.

Linux altında hangi donanımların desteklendiği ile ilgili ayrıntılı bilgiyi *Hardware-HOWTO*'dan alabilirsiniz. HOWTO dökümanları hakkında daha geniş bilgi ve nereden temin edeceğiniz kitabın sonunda detaylıca anlatılmıştır.

Makinanızda Linux çalıştırmak için kullanacağınız uygulamalara bağlı olarak en az bir 386SX işlemci ve 4 Mbyte RAM'a ihtiyaç duyacaksınız. Sabit disk üzerinde ise en az 40 Mbyte'lik bir alan ayırmanız gerekecektir. Rahat bir kullanım için en az 8 Mbyte RAM ve 200 Mbyte sabit disk ve bir 486 işlemci önerilmektedir.

1.2 Linux'un Kullanım Amaçları

Ücretsiz olarak dağıtılıyor ve gelişiminin hala devam ediyor olması biçcok kişinin Linux'un profesyonel alanlarda kullanılamayacağının düşünmesine yol açmaktadır. Oysa Linux işletim sistemini kullanan bilgisayarlar özel kullanım başta olmak üzere birçok alanda yaygın olarak kullanılmaktadırlar.

1.2.1 Kişisel Kullanım

Linux evinde veya işinde UNIX işletim sistemi altında çalışmak isteyenler için ideal bir platformdur. Özellikle işi veya eğitimi sırasında UNIX platformlar altında çalışmak, uygulamalar kullanmak veya yazılım geliştiren kişiler kendi kişisel bilgisayarlarında benzer ortamı yakalayabilmekte ve işlerini kendi kişisel bilgisayarlarında gerçekleştirebilmektedirler. Bunlara ek olarak Linux altında yer alan uygulamalar giderek sıradan bir kullanıcı için bile bu işletim sisteminin ilgi çekici hale gelmesini sağlamaktadır. Gelişimleri henüz tamamlanmamış olmasına rağmen, herhangi bir kişisel bilgisayardan beklenebilecek yazı editörleri, hesap cetvelleri, çizim yazılımları, veri tabanları birçok ihtiyaca cevap verecek düzeye gelmiştir. Örneğin LaTeX kullanıcıları MS-DOS altında buldukları desteğin çok daha fazlasını Linux altında bulabilmektedirler.

1.2.2 Internet Sunucusu

Linux doğrudan TCP/IP desteği ile gelmektedir. Bu yönü ile TCP/IP temelli bilgisayar ağlarında hem istemci hem de sunucu olarak yaygın kullanım bulmuştur. Üzerinde hali hazırda bulunan servislerin çeşitliliği, yeni çıkan servislere hızlı ayak uydurması, kolay konfigüre edilebilmesi ve özellikle de düşük maliyeti sebebi ile yaygın olarak İnternet servislerinin verilmesi amacıyla kullanılmaktadır. Zamanla verdiği ağ servisleri başka protokollere destek verecek sekilde genişletilmiştir. Şu anda Linux

- WWW sunucu
- DNS sunucu
- NFS sunucu
- NIS sunucu
- X Window sunucu
- BOOTP sunucu

- SMTP sunucu
- FTP sunucu
- LIST sunucu
- NEWS sunucu

gibi yaygın TCP/IP servislerinin yanısıra

- NOVELL sunucu (Novell protokolü kullanarak disk ve yazıcı servisi)
- SAMBA sunucu (Windows 3.1, Windows95, Windows NT ve WfW için disk ve yazıcı servisi)
- APPLETALK sunucu (MacOS kullanan Apple makinalar için disk ve yazıcı servisi)

verebilmektedir.

1.2.3 Ağ Elemanı

Linux yazılım desteği ile birçok ağ elemanının yerine geçebilecek bir alternatif olarak kullanılabilmektedir. Birden fazla ağın birbirine bağlanması amacıyla bir *yönlendirici (router)* olarak da kullanılabilmektedir. Özellikle farklı protokoller arası bir geçiş elemanı olarak yaygın şekilde Linux'tan yararlanılmaktadır. Ayırca yönlendirici olarak kullanıldığında kolaylıkla güvenlik amacıyla *firewall (alev duvarı)* olarak konfigüre edilebilmektedir. Buna ek olarak bir ağ üzerinde bulunan iki segmanın trafiğini birbirinden ayıran bir *köprü (bridge)* olarak da hizmet verebilmektedir. Birçok kurumda bir veya daha çok modemin bağlanması amacıyla bir *terminal sunucu (terminal server)* olarak Linux kullanılmaktadır.

1.3 Nereden Linux Bulabilirim ?

Linux işletim sistemiminin temelini oluşturan çekirdek, bu çekirdeğin kullandığı destek kütüphaneleri ve uygulama yazılımları bir araya getirilerek, yükleme yazılımları da eklenerek Linux dağıtımları meydana getirilmektedir. Bu dağıtımlar temel olarak bir kullanıcının Linux kullanmak için ihtiyaç duyabileceği bir çok yazılımı bir araya getirirler. Bu dağıtımların çoğu İnternet üzerinde anonim FTP arşivlerinde bulunabilmektedirler. İnternet erişimi bulunmayan kişilerinde (veya Internet'ten 120 Mbyte kopyalamak istemeyen kişilerin) yararlanabilmesi için çesitli CD-ROM şirketleri tarafından CD-ROM üzerinde dağıtımlar meydana getirilmiştir.

Linux dağıtımları ve bunların nereden temin edilebileceği hakkında detaylı bilgi, *Linux Kurulumu* başlığı altında bulunabilir.

1.3.1 Doküman Temini

Linux hakkında yazılan kitapların sayısı gün geçtikçe artmaktadır. Yine de en güncel bilgi İnternet üzerinde bulunmaktadır. Elektronik ortamda bulunan iki temel döküman çeşidi bulunmaktadır. Bunlar:

1. *Linux Documentation Project* `<http://w3.metu.edu.tr/ldp>`
2. *NASIL (HOWTO) dökümanlarıdır* `<ftp://ftp.metu.edu.tr/pub/sunsite/docs/HOWTO/>`

NASIL dökümanlarının bir kısmının Türkçeye çevrilmesi işlemi devam etmektedir. Şu ana kadar çevrilen dökümanları *Linux Kullanıcıları* `<http://linux.org.tr>` Grubu WWW sayfasından bulabilirsiniz.

1.4 Yazılım Özellikleri

Bir işletim sistemi, ne kadar mükemmel olursa olsun, uygulama yazılımlarının çokluğu ve kalitesi ile varolabilirler. Herhangi bir Linux dağıtımı içerisinde, değişik amaçlara hizmet eden birçok yazılım bulunmaktadır. Ancak her geçen gün bu dağıtımlarda yer almayan yeni yeni yazılımlar çıkmaktadır.

UNIX makinalar üzerinde yer alan uygulamaların çoğu, değişik platformlar altında bulunduğundan çalıştırılabilir (executable) olarak dağıtılmaz, kaynak kodu şeklinde sunulurlar. Sözkonusu yazılımı kullanmak isteyen bir kullanıcı bu kaynak kodunu kendi platformunda derleyerek çalıştırır. Bu tür yazılımların birçoğu Linux altında kolaylıkla çalıştırılabilmektedir. Bu tür yazılımları İnternet üzerinde çeşitli FTP arşivlerinde bulmak mümkündür.

Linux'a özel veya Linux üzerinde geliştirilen yazılımlar için standart bazı FTP arşivleri vardır. Bunların en bilineni *Sunsite FTP arşividir.* `<ftp://sunsite.unc.edu/pub/Linux>` Burada çeşitli dizinler altında konularına göre ayrılmış bir durumda elektronik devre tasarım yazılımlarından oyun programlarına kadar birçok değişik yazılım bulunmaktadır. Bu arşivin Türkiye'de yeralan bir kopyası `<ftp://ftp.metu.edu.tr:/pub/linux/sunsite>` adresinde vardır.

Linux'un bu denli sevilmesi ve yaygınlaşması çesitli şirketlerin (Macintosh, Sun, SSC gibi) Linux üzerinde çalışan ticari yazılımlar geliştirmesi sonucunu verdi. Bu konuda detaylı bilgi için *Commercial-HOWTO* dökümanından yararlanabilirsiniz.

Linux üzerinde bulunan uygulamaların ve yazılımların listeleri için aşağıdaki adreslerden yararlanabilirsiniz:

- *Linux Applications and Utilities* `<http://www.redhat.com/linux-info/linux-app-list/linapps.html>`
- *Scientific Applications on Linux* `<http://sal.kachinatech.com>`
- *Linux Software Map* `<http://www.execps.com/lsm>`

1.4.1 Temel Komutlar

Daha önce UNIX tabanlı bir işletim sisteminde çalışanlar için Linux, öğrenilmesi çok kolay bir sistem olacaktır. Standart bir UNIX sisteminde yeralan hemen hemen tüm komutlar, Linux'a taşınmıştır. Onlarca çeşit kabuğun yanı sıra, `sed`, `awk` gibi programcının işini kolaylaştıran diller, `ls, less, finger` gibi temel her türlü komut, Linux'ta vardır.

Ağ ve İnternet uygulamaları için `elm` ve `pine` (Pine Is Not Elm :-) yanında metin editörleri olarak `vi, vim` (vi'ın daha gelişmiş sürümü), `pico` ve `joe` sayılabilir. Editörlerden, bizde fazla bilinmeyen Emacs da Linux altında denemeye değer programlardandır. Kelime işlem programlarından `troff, groff` (GNU troff) ve daha modern metin işleme yazılımlarından TeX ve LaTeX sayılabilir.

Bazı program isimlerinin (GNU-tar, GNU-bash gibi) başında görebileceğiniz *GNU* (Gnu is Not UNIX!), Linux için de yazılım ve programlar üreten bir kuruluştur. GNU, lisansını ve yazarını korumak koşuluyla programları kaynak koduyla birlikte Linux kullanıcılarına dağıtır. GNU `bash` ve `tcsh`, Linux altında en çok rağbet edilen iki kabuk ismidir. Diğer kabuklar arasında `zsh`, `ash`, `ksh` ve `csh` sayılabilir. Kabuklar hakkında daha geniş bilgiyi, *Bash* konu başlığı altında bulabilirsiniz.

1.4.2 Uygulama Programları

Linux üzerinde ver tabanı uygulamaları ortalama bir kullanıcının ihtiyacını karşılayabilir. *Postgres* `<http://www.postgresql.org>`, `Mbase`, `msql` ve `Ingres` gibi profesyonel yazılımlar Linux ve diğer platformlarda istemci/sunucu bazda görev yapabilirler. Özellikle Postgres, uygulama kolaylığı ve C, perl, tcl gibi birçok dile yönelik arabirimiyle göze çarpar.

Mühendislik yazılımları arasında `gnuplot` (grafiksel veri analiz yazılımı), `xspread` ve `xfractint` (fraktal yaratma programı) sayılabilir.

Doğru seçilmiş bir donanım üzerinde kurulan bir Linux makinası, hemen her tür çokluortam (multimedia) uygulamalarını rahatlıkla çalıştırabilir. En az Pentium tabanlı, 32Mbayt RAM ve 2GB sabit diske sahip makina yardımıyla ticari olarak satılan çokluortam uygulamalarını kullanabilirsiniz. Linux, hemen her türlü ses kartını desteklediğinden ses dosyalarının, workman, Cdplayer gibi programlar yardımıyla kolayca çalınması mümkün olur. MIDI editörleri ve bir sentezleyici ile kendi müzik stüdyonuzu kurabilirsiniz.

Biraz oyun oynamak mı istediniz ? Doom, Quake, Abuse, Xtetris, FreeCiv(ya da CivNet), Imaze ve benzeri onlarca oyun Linux'ta da var. ODTÜ'de yüzlerce öğrenciyi bilgisayar başına mıhlayan MUD (Multi User Dungeon) oyunlarını sunan makinaların birkısmı Linux idi.

Bu oyunları çeşitli ftp adrteslerinden ücretsiz temin edebilirsiniz.

1.4.3 X Window Arabirimi

Linux işletim sistemi altında X Window sistemi ile Windows altındaki gibi grafik arabirimiyle birlikte çalışabilirsiniz. Windows ile uğraşan herkes rahatlıkla X Window'a geçiş yapabilir. X ile ekranda aynı anda birden fazla pencere açılabilir, fare yardımıyla birden fazla uygulama aynı anda kontrol edilebilir.

Pekçok uygulamanın (özellikle İnternet tabanlı) X üzerinde çalışan sürümleri vardır. Bu sayede metin tabanlı ekrana (vt100) dönmeden her işinizi X yardımıyla tamamlayabilme şansınız olur. Bu sayede Linux, bir iş istasyonu görünümüne ve kullanışlılığına sahip olacaktır.

X pencere denetleyici (window manager - wm) kullanıcı ile X arasında bekler ve klavye ile fareden aldığı emirleri ekranda yerine getirir. Bu emirler, pencerelerin açılması, kapatılması ve yerlerinin değiştirilmesi gibi komutlardır. Sıkça kullanılan pencere denetleyicileri `fvwm`, `twm` ve `olwm`'dir.

1.5 Linux ve Diğer İşletim Sistemleri

Linux ve diğer işletim sistemleri arasındaki ilişkiyi, benzerlikleri ve farklılıkları bilmek önemlidir. Linux işletim sistemi, diğer sistemler ile birlikte aynı sabit diski paylaşabilir. UNIX'i öğrenmek için kesinlikle en kolay ve ucuz yol olan Linux, diğer işletim sistemlerine karşı her zaman güçlü bir alternatif olmaktadır. İnternet servis sağlayıcılarının büyük çoğunluğu, Linux kullanmakta, İnternet bağlantılarını, e-posta ve haber grubu alış-verişini Linux sayesinde yapmaktadır.

Bir kişisel bilgisayarı satın aldığınız zaman çok büyük ihtimalle üzerinde MS-DOS veya türevi bir işletim sistemi yüklenmiş olduğunu göreceksiniz. Her kullanıcı o veya bu şekilde MS-DOS ile tanışır. MS-DOS, üzerinde en fazla program yazılan işletim sistemi olmuş, bu yüzden modern işletim sistemleri çıkmadan önce her kullanıcının kurtarıcısı gözüyle bakılmıştır. Fakat MS-DOS arabirimi programlanırken ileriyi düşünemeyen programcılar, bu işletim sistemine Linux'ta olan bazı hayati özellikleri kazandıramamışlardır. MS-DOS, çok kullanıcılı bir sistem değildir ve aynı anda birden çok işi yapamaz. İsterseniz çalıştırılabilecek en geniş programın büyüklüğünü görmek için `mem` komutunu kullanmayı deneyin. Linux, sadece üzerindeki hafıza ile sınırlıdır. 80x86 tabanlı mikroişlemcinin her özelliğini sonuna kadar kullanır. Bunun sonucu olarak verimli bir işletim sistemi sayılabilir.

Tüm bu olumsuzluklara rağmen yine de MS-DOS kullanmak istiyorsunuz. Onun da kolayını bulmuş Linux programcıları. Ücretsiz dağıtılan MS-DOS ve Windows emülatörü yardımıyla MS-DOS altında çalışan programların hemen hepsi Linux'la birlikte de çalışabilir. `WinWord 2.0, sysinfo, Civilization ve Qbasic` Linux altında sorunsuz çalışan MS-DOS/Windows programlarından birkaçı.

Profesyonel bir yatırım sayılabilecek Windows NT'nin çokgörevlilik ve hafızayı mükemmel kullanma gibi özellikleri vardır. Buna karşılık fiyatı oldukça yüksektir ve çalışmak için gayet yüksek standartlı bir makina ister.

IBM firmasının geliştirdiği OS/2 de NT'ye benzer şekilde çokgörevli işletim sistemi olup fiyat/performans oranı açısından makul sayılabilecek bir işletim sistemi olarak göze çarpar.

Linux, herkese göre bir işletim sistemi değildir. UNIX'i benimsemeyen bir kullanıcıdan Linux'u kullanmasını bekleyemezsiniz. İnternet'in kendisi UNIX tabanlı olduğu için Linux bu alanda yukarıda sözü geçen sistemlerden daha avantajlı bir duruma gelir. Dağıtımı tek firmayla sabit olmadığı için geliştirilme aşamasında dünyanın dört bir yanından katkı ve destek gelir. UNIX'e belirli bir standart getirilmesi için yapılan çalışmalara uyan Linux, POSIX standardını destekler.

RedHat, Caldera gibi çeşitli Linux sürümleri da piyasada satılmakta, çok zengin bir döküman ve arşiv kaynağı ile kullanıcılara sunulmaktadır. 80x86 tabanlı işlemciler üzerinde kurulabilen ve İnternet üzerinde bedava dağıtılan diğer işletim sistemlerinden birisi FreeBSD'dir. FreeBSD, bir grup programcı tarafından BSD standartını 80x86 bilgisayarlara taşımak üzere geliştirilmiştir ve Linux ile büyük benzerlikleri bulunur.

Linux işletim sisteminin güvenilir ve sağlam yapısından haberdar olmayan, hatta bu gerçeği bilerek ücretli sistemlerdem medet uman kişiler vardır. Sabit fikirli olmadan önce bu sistem hakkında daha fazla verinin dağıtılması, kullanıcıların bilinçlendirilmesi gereklidir. Bu da ancak, Linux'un yapabildiklerini göstermekle olabilir.

1.6 Türkiye'de ve Dünyada Linux

Eğer Linux'u sever ve bu sistemin yaygınlaşmasında, bihaber kullanıcılara tanıtımında öncülük etmek isterseniz, bu bölüm tam size göredir.

Henüz tüzel kişiliği olmayan *Türkiye Linux Kullanıcıları Grubu sayfası* `<http://www.linux.org.tr>` Türkiye'de Linux çalışmaları için iyi bir başlangıç noktasıdır. Türkçe ve İngilizce Türkiye'deki en geniş kapsamlı Linux sunucusunun genişlemesinde yeni fikirlere, çözüm önerilerine ihtiyaç vardır. Bu da ancak Linux seven bir kitlenin varlığıyla mümkündür.

Dünyada ve Türkiye'de İnternet Servis Sağlayıcıları (İSS) Linux'u uzunca bir süredir ticari olarak kullanıyorlar. Uygulama yazılımları da bu yönde kullanıcıların hizmetine sunuluyor. Fakat güvenlik sorunları nedeniyle Linux kullanan servis sağlayıcıları, müşterilerini bu hizmetten yoksun tutuyorlar. Açık sistem kavramının yaygınlaşması da bu sebepten dolayı sekteye uğruyor. *http://www.10mb.com/linux/* `<http://www.10mb.com/linux/>` adresinde, ticari uygulama yazılımlarına destek veren bir proje yeralmakta.

SSC (Specialized Systems Consultants - http://www.scc.com), “Linux Journal” adında bir dergiyi başarıyla dağıtıyor. Tirajı onbinleri bulan bu aylık dergide bu işletim sistemini aktif kullanan herkes için, her konuda, her düzeyde makale bulmak mümkündür.

1.7 Linux ve Donanım Desteği

Bir işletim sisteminin tüm kartları tanıması, tüm sabit disklerle çalışabilmesi, tüm giriş/çıkış kartlarıyla uyum içinde çalışması mümkün değildir. Bu konuda çok iddalı olan tak-çalıştır sistemine sahip Windows95 bile bazen yetersiz kalabilmekte. Linux da piyasada yeralan hemen hemen bütün donanımlarla birlikte çalışabilir.

Linux, üzerinde matematiksel işlemci olsun veya olmasın Intel 386SX/DX, 486SX/DX/SX2/DX2/DX4, Pentium ve PentiumPro işlemcilerde sorunsuz çalışır. Bunlarla beraber (Cyrix 486 tabanlı işlemcilerinde nadiren sorun çıkarsa da) AMD, Cyrix gibi firmaların işlemcileri de Linux tarafından desteklenir. Matematik işlemcisi olmayan bilgisayarlarda Linux'un beyni sayılan çekirdek, matematik işlemcisine gerek duyulan kod parçalarında bu işlemciyi emüle edebilir. Burada belirtilen IBM uyumlu PCler dışında ALPHA, PowerPC, MIPS, farklı Sparc modelleri, PA-RISC gibi birçok farklı işlemciye de başarılı bir şekilde taşınmıştır.

Linux, kişisel bilgisayarlarda kullanılan ISA, VLB (Vesa Local Bus - yerel veri yolu), EISA, MCA (IBM Microchannel) veya PCI veriyolu mimarisi ile çalışabilirler.

Linux, SMP (symmetric multi processor) olarak da bilinen birden fazla işlemcili bilgisayarlar üzerinde de çalışabilir ve birden fazla işlemciyi en verimli şekilde kullanır.

Kişisel bilgisayarların kullandığı 80x86 tabanlı işlemciler dışında, taşınan ve sorunsuzca çalıştığı bilinen işlemciler ve bilgisayarlar ile konu hakkında daha fazla bilgi alınabilecek İnternet adresleri aşağıda verilmiştir.

- *Linux/68000* `<http://www-users.informatik.rwth-aachen.de/ hn/linux68k.html>`
- *Linux/MIPS* `<http://www.waldorf-gmbh.de/linux-mips-faq.html>`
- *Linux/PowerPC* `<ftp://sunsite.unc.edu/pub/Linux/docs/ports/Linux-PowerPC-FAQ.gz>`
- *Linux for Acorn* `<http://www.ph.kcl.ac.uk/ amb/linux.html>`
- *MacLinux* `<http://www.mklinux.apple.com>`
- *Compaq Deskpro XL* `<http://www-c724.uibk.ac.at/XL/>`
- *IBM PS/2 MCA systems* `<ftp://invaders.dcrl.nd.edu/pub/misc/>`
- *Compaq Contura Aero* `<http://domen.uninett.no/ hta/linux/aero-faq.html>`
- *IBM ThinkPad* `<http://peipa.essex.ac.uk/tp-linux/tp-linux.html>`
- *Linux/MIPS* `<http://lena.fnet.fr/>`
- *Linux/Alpha* `<http://www.azstarnet.com/ axplinux/>`
- *HP PA-RISC* `<http://www.osf.org/mall/os/pa-mklinux/index.html>`
- *SPARC/Linux* `<http://www.geog.ubc.ca/sparclinux.html>`

Linux, metin modu kullanırken tüm ekran kartlarıyla (Hercules, CGA, EGA, CGA, IBM monokrom) sorunsuz çalışır. X Window çalıştırmak isterseniz hızlandırılmış bir SVGA kart önerilir. Sürekli yenileri eklense de aşağıda şu anda desteklenen kartların tam listesi vardır.

- Hercules mono
- VGA / VGA Mono
- EGA
- ARK Logic ARK1000PV/2000PV, ARK1000PV/VL
- ATI VGA Wonder, ATI Mach32, ATI Mach8, ATI Mach64
- Cirrus 542x, 543x, 62x5, 6420/6440

- OAK OTI-037/67/77/87
- Trident TVGA8900, TVGA8800, TVGA9xxx
- Tseng ET3000/ET4000/W32, ET4000/W32/W32i/W32p, ET4000AX
- IBM 8514/A, IBM XGA, XGA-II
- IIT AGX-010/014/015/016 (16 bpp)
- Oak OTI-087, OTI-067, OTI-077
- S3 911, 924, 801, 805, 928, 864, 964, Trio32, Trio64, 868, 968
- Weitek P9000 (16/32 bpp)
- Diamond Viper VLB/PCI
- Orchid P9000
- Western Digital PVGA1, WD90C00/10/11/24/30/31/33
- Avance Logic AL2101/2228/2301/2302/2308/2401
- Chips & Technologies 65520/65530/65540/65545
- Compaq AVGA
- Genoa GVGA
- MCGA (320x200)
- MX MX68000/MX68010
- NCR 77C22, 77C22E, 77C22E+
- RealTek RTG3106
- Video 7 / Headland Technologies HT216-32
- Western Digital/Paradise PVGA1, WD90C00/10/11/24/30/31/33
- Hyundai HGC-1280
- Sigma LaserView PLUS

1.7.1 Sabit Diskler ve Sabit Disk Denetleyicileri

Linux, standart IDE, bazı ESDI, hemen hemen tüm SCSI ve nadiren kullanılsa da MFM ve RLL denetleyicilerini desteklerler. Aslında Linux çalıştırmak için sabit diske gerek yoktur. Ağ üzerinden, Bootp protokolü yardımıyla Linux yüklü başka bir bilgisayarın kaynaklarını kullanmak mümkündür. Sabit disk denetleyiciniz en az 16 bit olmalıdır. Genellikle MS-DOS altında sorunsuz çalışan her sabit diski Linux da görebilir.

Sabit disk üzerinde Linux için bir miktar yer ayırmalısınız. Birden fazla disk de kullanabilirsiniz, Linux her diski ayrı bir dizin altından erişebilir. Bu konuda daha geniş bilgiyi Linux kurulumu bölümünde bulabilirsiniz.

Sabit diskler, denetleyicileri desteklendiği sürece Linux altında kullanılabilirler. Artık neredeyse tüm CD-ROM'lar SCSI denetleyicilerle çalışıyorlar. Bir SCSI denetleyiciniz varsa makinanız CD-ROM'u da tanıyacaktır. Linux, CD-ROM'ların standart iso9660 dosya sistemini de tanır.

Desteklenenen kartlar,

- AMI Fast Disk VLB/EISA
- Adaptec AVA-1505/1515, AHA-1510/152x, AHA-154x, AHA-174x, AHA-274x, AHA-2940/3940, ACB-40xx
- Always IN2000
- BusLogic (ISA/EISA/VLB/PCI)
- DPT PM2001, PM2012A (EATA-PIO)
- DTC 329x (EISA) (Adaptec 154x compatible)
- Future Domain TMC-16x0, TMC-3260 (PCI), TMC-8xx, TMC-950
- Media Vision Pro Audio Spectrum 16 SCSI (ISA)
- NCR 5380 generic, 53c400, 53c406a, 53c7x0, 53c8x0 (PCI)
- Qlogic / Control Concepts SCSI/IDE (FAS408) (ISA/VLB)
- Seagate ST-01/ST-02 (ISA)
- SoundBlaster 16 SCSI-2 (ISA)
- Trantor T128/T128F/T228 (ISA)
- UltraStor 14F (ISA), 24F (EISA), 34F (VLB)
- Western Digital WD7000 SCSI
- AMD AM53C974, AM79C974 (PCI)
- Adaptec SCSI-MFM/RLL bridgeboard
- Iomega PC2/2B

- Qlogic (ISP1020) (PCI)
- Ricoh GSI-8

1.7.2 Ethernet Kartları

Piyasada çok çeşitli ethernet kartları vardır. Genellikle yaygın olarak kullanılanlar 3Com veya NE2000 uyumlulardır. Aşağıda desteklenen ethernet kartların bir listesi yeralmaktadır. PCMCIA, Tokenring, ISDN, AX25 kartlarının uyumlu olanları, çok nadir kullanıldıkları için burada belirtilmeyecektir.

- 3Com 3C501, 3Com 3C503, 3C505, 3C507, 3C509/3C509B (ISA) / 3C579 (EISA)
- AMD LANCE (79C960) / PCnet-ISA/PCI (AT1500, HP J2405A,
- NE1500, NE2100, NE2000, NE1000
- AT&T GIS WaveLAN
- Allied Telesis AT1700
- Ansel Communications AC3200 EISA
- Apricot Xen-II
- Cabletron E21xx
- DEC DE425 (EISA) / DE434/DE435 (PCI), DEC DEPCA
- HP PCLAN 27245, 27247, 27252A, 10/100VG PCLAN
- Intel EtherExpress, EtherExpress Pro
- New Media Ethernet
- Racal-Interlan NI5210, NI6510
- PureData PDUC8028, PDI8023
- SEEQ 8005
- SMC Ultra
- Schneider & Koch G16
- Western Digital WD80x3
- Zenith Z-Note / IBM ThinkPad 300 built-in adapter

1.7.3 Ses Kartları

Linux üzerinde hemen her türlü ses kartı desteği var. SoundBlaster16 ses kartlarının üzerinde ASP çipi veya 4.11 ve 4.12 DSP (digital signal processor - sayısal ses işleyici) bulunanları Linux üzerinde kullanamazsınız.

Desteklenen ses kartları,

- 6850 UART MIDI
- Adlib (OPL2)
- Audio Excell DSP16
- Aztech Sound Galaxy NX Pro
- Crystal CS4232
- CHO-PSS (Orchid SoundWave32, Cardinal DSP16)
- Ensoniq SoundScape
- AWE 32
- Gravis Ultrasound, Gravis Ultrasound MAX
- Logitech SoundMan Games, Logitech SoundMan Wave
- Logitech SoundMan 16 (PAS-16 uyumlu)
- MPU-401 MIDI
- MediaTriX AudioTriX Pro
- Media Vision Premium 3D (Jazz16), Pro Sonic 16 (Jazz), Pro Audio Spectrum 16
- Microsoft Sound System (AD1848)
- OAK OTI-601D cards (Mozart)
- OPTi 82C928/82C929 cards (MAD16/MAD16 Pro)
- Sound Blaster, Sound Blaster Pro, Sound Blaster 16
- Turtle Beach Wavefront cards (Maui, Tropez)
- Wave Blaster

1.7.4 Fare

Linux, `Microsoft serial mouse`, `Mouse Systems serial mouse`, `Logitech Mouseman serial mouse`, `Logitech serial mouse`, `ATI XL Inport busmouse`, `Microsoft busmouse`, `Logitech busmouse` ve `PS/2 mouse` destekler.

Genellikle kullanacağınız farenin türü Microsoft ya da Mouse Systems serial mouse olacaktır.

1.7.5 Modem, Yazıcı ve Oyun Çubuğu

Hem internal (kasa içine takılan) hem de external (kasanın dışında kalan) tüm modemler Linux tarafından desteklenir. Aynı şekilde paralel veya seri porta takılan her yazıcı ve çizici desteklenir. İsterseniz bunları yerel bir ağ üzerinden birden fazla makinaya paylaştırabilirsiniz. Linux altında `lpr` yazılımı, yazıcılara erişimi sağlamak için kullanılır.

Oyun çubukları için sürücüler ister doğrudan çekirdeğe eklenebilir, istenirse de modül olarak derlenebilir.

Aşağıda, Linux'un desteklediği giriş/çıkış kartlarının geniş bir listesi vardır.

- AST FourPort and clones
- Accent Async-4
- Arnet Multiport-8
- Bell Technologies HUB6
- Boca BB-1004, 1008, BB-2016, IO/AT66, IO 2by4
- Computone ValuePort
- DigiBoard PC/X (4, 8, 16 port)
- Comtrol Hostess 550 (4, 8 port)
- PC-COMM 4-port (4 port)
- SIIG I/O Expander 4S (4 port, uses 4 IRQ's)
- STB 4-COM (4 port)
- Twincom ACI/550
- Usenet Serial Board II (4 port)
- Cyclades Cyclom-8Y/16Y (8, 16 port) (ISA/PCI)
- Stallion EasyIO, EasyConnection 8/32, 8/64

1.8 Linux Avantaj ve Dezavantajları

Pekçok insan, "neden Linux?" diye sorabilir. Belki de cevap önce kullanıcının kendini tanıması ile bulunabilir. Değişik yerlerde Linux kullanılması ve bunun sonuçları hakkında gözlemlediklerimizin ışığı altında çok kabaca:

Eğer,

Bilgisayarla ilişkiniz belirli paket programlara dayanıyorsa, bilgisayar kullanmak için bilgisayar konusunda bilgi sahibi olmanız gerektiğine inanmıyorsanız, bilgisayar ile uğraşmak hoşunuza gitmiyorsa, sorunlarınızı kendi başınıza çözmeyi denemekten hoşlanmıyorsanız, bir sorun çıktığında para vererek de olsa bu sorununuzu birisi aracılığı ile çözmek istiyorsanız

Linux kesinlikle size göre değil.

Ama eğer,

Bilgisayarınızla ilgilenmekten hoşlanıyorsanız, bilgisayarda çıkan problemlerle uğraşmak hoşunuza gidiyorsa, diğer işletim sistemlerinin sizi sıktığına ve sınırladığına inanıyorsanız, donanımınızdan daha çok performans istiyorsanız, UNIX işletim sistemi ile çalışmayı seviyorsanız

Linux size göre olabilir.

1.8.1 Avantajları

UNIX işletim sistemine sahip bir bilgisayar kullanmak istiyorsanız ve bu işletim sisteminde platforma bağımlı bir yazılım kullanmıyorsanız, Linux ideal bir çözümdür.

Linux ücretsizder. Sadece işletim sisteminin maliyeti açısından değil, verdiği performans için ihtiyaç duyduğu donanım açısından da çok ucuzdur. Üstüne üstlük çok kullanılan ve bol yedek parçası bulunan bir platform altında çalıştığı için belirli bir Linux sisteminin performansını artırmak için yapılması gereken yatırım başka bir UNIX iş istasyonunu aynı oranda geliştirmek için gereken yatırıma göre çok düşüktür. Herhangi bir Sun bilgisayarın hafizasını iki katına çıkarmak için harcanacak para ile bir Linux-PC'nin hafizasını iki katına çıkarmak için harcanması gereken parayı kıyaslamayı deneyin. Fakat şirketler bazında Linux'un bedava bir işletim sistemi olması genelde gözardı edilir.

Bir Linux makine bu sayede sadece işletim sistemi açısından değil donanım olarak da ucuza gelmektedir.

Linux hızla geliştirilmektedir. Bu gelişimin en büyük yararı, eksikliklerin kullanıcıların talepleri ve çabaları sonucunda hızla giderilmesidir. Linux diğer tüm işletim sistemlerine göre belirli bir donanım için daha hızlı destek verebilmektedir.

Linux çok değişik donanımlar ve servisler icin özel olarak hazırlanır. İşletim sisteminin temelini oluşturan çekirdek kullanıcı tarafından da derlenebildiği için, bu derleme sırasında sadece kullanım amacına yönelik alt programlarla donatılır. Bu genel olarak daha sistemin performansını artırmaktadır. (Örnek olarak SCSI donanımınız yoksa çekirdeğinizde SCSI ile ilgili alt programlara yer vermezsiniz)

1.8.2 Dezavantajları

Linux'un serbestçe dağıtılıyor olması bir çok kişinin bu işletim sistemine güvenmemesine yol açmıştır. "Ciddi bir şey olsa, bedava olmazdı !" kanısı oldukça yaygındır.

Linux'un sürekli gelişiyor olması en büyük dezavantajlarından biridir. Henüz tüm ihtiyaçlara cevap vermemesi (hala Windows95 programlarını çalıştıramıyor :)), gelişimin bazı aşamalarında topyekün değişiklikler yapılması, gelişimi takip etmek için bazen sürekli yenileme yapılması, birçok kullanıcının bu işletim sistemine güvenmemesine yol açmıştır.

Linux herhangi bir ticari destek altında gelişmemektedir. Bunun en büyük yararı işletim sisteminin ticari kaygılar taşımamasıdır. Ancak bunun yanısıra diğer işletim sistemlerinde olan teknik destek, dağıtım ve dökümantasyon alanlarında eksikleri vardır. Özellikle teknik destek eksikliği, anahtar teslim çözümlere alışmış kullanıcıların çokluğu Linux kullanıcılarının sayısını sınırlayan temel faktörler olmuşlardır.

Bu konudaki eksikliği gidermek için çeşitli gönüllü kuruluşlar, kullanıcı grupları oluşmuştur. Zamanla Linux teknik desteği ticari bir konu olarak ortaya çıkmıştır. Şu anda tüm dünyada Linux çözümleri konusunda teknik destek veren danışmanlar bulunmaktadır. Bu kişiler ve kurumlar hakkında ayrıntılı bilgiyi *Consultants-HOWTO* dökümanında bulabilirsiniz.

Linux işletim sistemini geliştirenlerin ticari kaygılar gütmemeleri bazı ticari yazılımların Linux üzerinde gelişmemesine sebep olmuştur. Linux üzerinde belirli konularda diğer işletim sistemlerinden aşağı kalmayan yazılımlar bulunmasına rağmen, belirli bazı konularda çok zayıf kalmıştır. (Mesela oyunlar :))

Linux üzerinde yer alan çözümlerin hepsi, basit kullanıcıların rahatça kullanabileceği düzeyde değildir. Bazı çözümler kullanıcıların belirli bir yazılım ve işletim sistemi bilgisine sahip olmalarını gerektirmektedir.

Sonuç olarak: Biz uzun süre Linux kullandık ve çok zevk aldık. Diğer işletim sistemleri ile yapamadığımız birçok uygulamayı Linux sayesinde gerçekleştirdik. Belirli bir donanım ile gösterdiği performansın bir çok işletim sisteminin üzerinde olduğunu gördük. Ne var ki her uygulama için Linux'un ideal bir çözüm olarak ortaya çıkmadığının bilincinde olmak gerekir. Belki ilerleyen senelerde üzerinde geliştirilen uygulama yazılımları daha çok kişiyi başka işletim sistemleri kullanmaktan vazgeçirecektir.

Chapter 2

Linux Kurulumu ve Başlangıç

Linux hakkında ilk dökümanlar yazıldığında, Linux çalıştırabilecek bilgisayarların özellikleri önemliydi. Linux 32 bitlik bir işletim olduğundan en az 80386SX işlemcilerle çalışmaktadır. 8086 ve 80286 işlemcili IBM-PC uyumlu kişisel bilgisayarlar Linux tarafından desteklenmemektedir. Hafıza olarak en az 4 Mbyte RAM (yoğun işlemler için 12 veya 16) tavsiye edilmektedir. Teknik olarak 2 Mbyte ile de çalışabilmesi gerekir. Pratik olarak bugün piyasada bulunan hemen hemen her IBM-PC uyumlu kişisel bilgisayarda Linux çalışabilmektedir.

8 Mbyte RAM'a sahip herhangi bir 486 üzerinde hemen hemen her türlü uygulama rahatlıkla çalıştırılabilmektedir. Tabii ki daha fazla RAM ve daha hızlı işlemciler sistemin genel olarak daha hızlı çalışmasını sağlayacaklardır.

Linux tarafından desteklenen donanımlar her geçen gün değişiyor. Bilgisayarınızda bulunan herhangi bir donanımın desteklenip desteklenmediğini *Hardware-HOWTO* dosyasından öğrenebilirsiniz.

Yine de daha önce yazılanları kısaca tekrar edersek,

1. Kişisel bilgisayarlarda INTEL, AMD, CYRIX şirketlerinin tüm 80386, 486, 586, 686, Pentium, PentiumPro işlemcileri
2. Tüm IDE, MFM, RLL sabit diskler
3. Çoğu SCSI sabit disk denetçileri
4. Çoğu ethernet ve G/Ç kartları
5. Birçok VGA, SVGA, EGA, HERCULES görüntü kartları

Linux tarafından desteklenmektedir.

Linux başka işletim sistemleri ile aynı sabit diskte bulunabilir. Makinanıza Linux yüklemek için mevcut işletim sisteminizi kaldırmak zorunda değilsiniz. Fakat yine de Linux yükleyebileceğiniz bir miktar alan ayırmak zorundasınız. Bir bilgisayara Linux yüklemek için bilgisayarınız üzerinde bir başka işletim sisteminin bulunmasına gerek yoktur, Linux tam anlamıyla kendi başına çalışabilen bir işletim sistemidir.

"Linux sabit disk üzerinde ne kadar yer kaplar?" sorusuna kesin bir cevap vermek oldukça zor, zira bu hangi yazılımları yükleyeceğinize ve ne kadar kullanıcı alanı istediğinize çok bağlıdır. Yine de kaba rakamlar vermek gerekirse, 40 Mbyte'lık bir alana çalışılabilir durumda ve işinizin çoğunu görebilecek bir Linux kurulabilir. Tüm paketleri yüklemeye kalktığınızda ise kabaca 250 Mbyte kadar yer kaplayacaktır.

Linux, çok çeşitli gruplar tarafından sürekli geliştirilen bir işletim sistemidir. Belirli kişiler ve topluluklar Linux için geliştirilen temel işletim sistemini ve uygulama yazılımlarını bir araya getirerek dağıtımlar oluştururlar. Bir Linux dağıtımı temel olarak bir makineye Linux kurmak ve o bilgisayar üzerinde Linux'la çalışmak için gerekecek tüm yazılımları ve paketleri içerir, bu yazılımların yüklenmesi için bir yükleme yazılımı sağlar.

Şu an mevcut çok çeşitli Linux dağıtımları mevcuttur. Bu dağıtımlar içerdikleri paketler ve yükleniş şekilleri açısından bazı ufak tefek farklılıklar gösterseler de temelde aynı işletim sistemini yüklerler. Bir Linux dağıtımı bir araya getirildiği zamandaki güncel işletim sistemini içermektedir. Örnek olarak Linux işletim sisteminin temeli olan çekirdek neredeyse her hafta yenilenerek geliştirilmektedir. Oysa dağıtımlar senede ancak birkaç kere oluşturulurlar.

Linux dağıtımları geleneksel olarak 3.5" lik disketler halinde hazırlanır. (her biri bir 3.5" lik diskete sığabilecek seri dizinler şeklinde). Her konu ile ilgili bir seri disket bulunur (örnek olarak n serisi ağ uygulamaları için n1,n2,n3... şeklinde)

2.1 Mevcut Dağıtımlar

Yaygın olarak kullanılan dağıtımlar aşağıda yer almaktadırlar. Burada anlatılacak olan kurulum bilgileri, Linux Slackware dağıtımına sahip olduğunuz farzedilerek hazırlanmıştır.

2.1.1 Slackware

Belki de en yaygın olarak yer alan dağıtım Slackware dağıtımıdır. Bu döküman boyunca anlatılacak olan komutlar ve sistem özellikleri Slackware dağıtımına göre düzenlenmiştir. Diğer dağıtımlar için anlatılan işlemlerin karşılıkları için dağıtım hakkındaki açıklamalara başvurabilirsiniz. Slackware dağıtımlarının temin edilebileceği yerler,

- *Slackware dağıtıcısı Walnut Creek* `<http://www.cdrom.com/titles/slackware.html>`
- *Orjinal Slackware FTP arşivi* `<http://ftp.cdrom.com/pub/slackware>`

Türkiye içerisinde Slackware Dağıtımı bulunduran bazı FTP arşivleri:

- *Orta Doğu Teknik Üniversitesi*
  `<ftp://ftp.metu.edu.tr/pub/linux/sunsite/distributions/slackware>`

- *Ege Üniversitesi* `<ftp://ftp.ege.edu.tr/pub/Linux/>`
- *Akdeniz Üniversitesi* <ftp://ftp.akdeniz.edu.tr/pub/linux>
- *İstanbul Teknik Üniversitesi*
  <ftp://ftp.itu.edu.tr/pub/linux/sunsite/distributions/slackware>

2.1.2 Red-Hat

Genel dağıtım

```
Red Hat Software
3201 Yorktown Rd, Suite 123 DeKalb Center
Durham, NC 27713
Internet adresi :  redhat@redhat.com
```

RED-HAT Linux dağıtımı WWW sayfası <http://www.redhat.com>

Orjinal RED-HAT FTP arşivi <ftp://ftp.redhat.com/pub>

Türkiye içerisinde Red Hat dağıtımı bulunduran FTP arşivleri

Ege Üniversitesi <ftp://ftp.ege.edu.tr/pub/linux/redhat>

ODTÜ <ftp://ftp.metu.edu.tr/pub/linux/redhat>

2.1.3 Debian

Genel dağıtım

```
The Debian Linux Association
Software in the Public Interest
P.O. Box 70152
Pt. Richmond CA 94807-0152
```

Debian Linux dağıtımı WWW sayfası <http://www.debian.org>

Orjinal Debian FTP arşivi <ftp://ftp.debian.org/debian>

Türkiye içerisinde Debian dağıtımı bulunduran FTP arşivleri

ODTÜ <ftp://ftp.metu.edu.tr/pub/linux/distributions/debian>

2.1.4 InfoMagic

Genel dağıtım

```
InfoMagic, Inc.
PO Box 30370. Flagstaff, AZ 86003-0370
Tel: (800)-800-6613 (siparis)
(520)-526-9852 (teknik destek)
Fax: (520)-526-9573
Internet adresi : info@infomagic.com
```

InfoMagic WWW sayfası <http://www.infomagic.com>

Bu konuda daha ayrıntılı ve güncel bilgi için *Distribution-HOWTO*'dan yararlanabilirsiniz.

2.2 Ne Şekilde Bulabilirim ?

Geleneksel olarak dağıtımların disketler ile yapıldığını söylemiştik. Ancak günümüzde 100'den fazla disketle dağıtım pek pratik olmamaktadır. Linux kurduğunuz yere bağlı olarak değişik alternatifleriniz olabilir.

2.2.1 CD-ROM

Çeşitli dağıtımlara ait CD-ROM'lar çeşitli dükkanlarda satılmaktadırlar. Linux aslında ücretsiz bir işletim sistemidir. Alınan ücret CD'nin basımı dağıtımı, ambalajını karşılaması için alınmaktadır. fiyatları kabaca 10-30 mertebesinde (bazen birkaç yüz doları da bulabiliyor) değişmektedir. Bazı dağıtımlarda telif kitapların bulunması bu dağıtımları daha pahalı yapabilmektedir. Eğer herhangi bir şekilde bir Internet erişiminiz yoksa ve bir akademik çevrede değilseniz en pratik Linux elde etme yöntemi CD-ROM'lardır.

Türkiye içerisinde Linux dağıtımı bulunduran CD-ROM satıcıları,

```
ADA Multimedia Center
Tunus Cad.  70/2 Kavaklıdere Ankara
Tel :  (312) 467 37 28
(312) 467 33 32
```

2.2.2 İnternet

Tüm Linux dağıtımları Internet üzerinde anonim FTP hizmeti ile sunulmaktadır. Ancak bireysel olarak dağıtımın tüm disketlerini bu yolla almak çok pratik olmayacaktır. Örnek olarak son slackware dağıtımı 110 Mbyte civarında yer kaplamaktadır. Internet aracıyla dağıtım elde etmek, yerel kullanıcılarına yeni dağıtımlar sunmak isteyen sistem sorumluları için ilginç bir çözüm olmaktadır.

2.2.3 NFS

Bir yerel bilgisayar ağına sahip kurumlarda paylaşılan bir disk alanı üzerinden Linux yüklemek mümkündür. Bu sayede CD-ROM veya Internet aracılığı ile elde edilen bir dağıtım kurumda ortak bir disk alanına yerleştirilir ve ağa bağlı makinalara yükleme yapılabilir.

Birçok kurumda Linux yüklemek için, geçici olarak bir ethernet kartı takılır, bilgisayar ağ desteği veren bir şekilde açılır ve Linux disketleri bu şekilde ağ üzerinden yapılır.

2.2.4 Sabit Disk

Herhangi bir şekilde Linux dağıtımı disketleri bir sabit diske de kopyalanmış olabilir. Bu durumda mevcut sabit disk üzerindeki disketler kullanılarak da yükleme yapılabilir. Bu yöntem yine daha çok eğitim kurumlarında başka kaynaklardan elde edilmiş bir Linux dağıtımını kopyalamak için kullanılır.

Bir kullanıcı makinesine Linux yüklerken tüm özelliklerini yüklemek istemeyebilir. Yüklediği yazılımların bazılarının gereksiz olduğunu düşünüyorsa ileride bazı uygulamaları kaldırabilir veya sonradan gerekli gördüğü yazılımları dağıtım disketlerinden rahatlıkla ekleyebilir.

Linux sadece dağıtımlarda bulunan yazılımlarla sınırlı bir işletim sistemi değildir. Dağıtımlarda bulunan yazılımlara ek olarak çok çeşitli başka uygulamalar mevcuttur. (Örnek olarak Netscape hiç bir Linux dağıtımında bulunmamaktadır). Bu ek yazılımların birçoğu İnternet üzerinde FTP arşivlerinde bulunmaktadırlar. Kitabın sonundaki ek, Linux uygulama yazılımlarının nerede bulunduğu hakkında biraz bilgi veriyor.

CD-ROM üzerindeki dağıtımlarda, dağıtımların yanısıra bir çok tanınmış FTP arşivinde yer alan yazılımlar ve çeşitli dökümanlarda yer alır. İnternet bağlantısı olan bir kurumda çalışmayan (veya bu tür bir kuruma erişimi bulunmayan) birisi için bir CD-ROM çok iyi bir çözümdür.

İnternet bağlantısı bulunan (özellikle akademik) kurumlarda güncel dağıtımların İnternet üzerinden aktarılması ve buradaki kullanıcılara NFS ve sabit disk üzerinde aktarılması daha pratik bir uygulamadır. Aynı kurumlar kendi çalışma alanları ile ilgili buldukları ek yazılımları da FTP arşivlerinden toparlayabilir ve kullanıcılarına bu ek yazılımları sunabilirler.

2.3 Sabit Disk Üzerinde Linux İçin Yer Açmak

Linux işletim sistemini yüklemek için sabit diskiniz üzerinde Linux için bir miktar yer ayırmak zorundasınız. Herhangi bir sabit disk bir işletim sisteminde kullanılabilmesi için ilk olarak bölümlere (partition) ayrılır. Daha sonra bu bölümler işletim sistemine uygun şekilde formatlanır. Linux işletim sistemi kendi disk formatını (ext2) kullanır. En yüksek verimi sabit disk üzerinde, kendi bölümünde, kendi disk formatı altında çalıştığı zaman verebilir. Eğer ayrı bir bölümlendirme yapılamıyorsa, tavsiye edilmese bile MS-DOS formatlı bir disk üzerinde de Linux kurulabilir (UMSDOS) ancak bu sistemin performansı diğerine göre oldukça düşük olacaktır. Bu dökümanın geriye kalan tüm kısımlarında bilgisayarınıza Linux yüklemek için sabit disk üzerinde Linux'a özgü bir bölüm ayrılacağı ve bu bölüme yükleneceği kabul edilecektir. MS-DOS formatlı bir disk hiyerarşisi altına Linux kurmak için *UMSDOS-HOWTO* dökümanından yararlanabilirsiniz. Eğer bilgisayarınızı bir süredir kullanıyorsanız, büyük bir ihtimalle sabit diskinizin tümünü kullandığınız işletim sistemi için ayrımış durumdasınızdır. MS-DOS kullanıyorsanız diskiniz bir (sadece C:) veya birden fazla (C: D: ..) bölüme ayrılmış durumda olabilir. MS-DOS altında bir disk üzerinde en fazla 4 temel bölüm olabilir (primary partition). Eğer daha fazla bölüme ihtiyaç varsa temel bölümlerden biri genişletilmiş bir bölüm olarak ayrılır (extended partition) ve bu bölüm üzerinde mantıksal bölümler ayrılır (logical partitions).

Bilgisayarınızda birden fazla bölüm varsa bir bölümü boşaltıp bu bölümü Linux için ayırabilirsiniz. Eğer tek bölümünüz varsa, veya mevcut bölümlerinizden birini tümüyle harcamak istemiyorsanız diskinizi yeniden bölümlemeniz gerekecektir. Klasik olarak bu durumda bölmek istediğiniz bölümdeki yazılımların yedeğini almanız, daha sonra MS-DOS altında ***fdisk*** yazılımı yardımı ile sözkonusu bölümü silmeniz, yeni boyutu ile yeniden yaratmanız, bu bölümü format komutu ile formatlamanız ve yedeğini aldığınız yazılımları yeniden yerleştirmeniz gerekecektir (Çok iş :))

Bazı yazılımlar mevcut bölümünüzü iki parçaya ayırabilirler. Örnek olarak ***fips*** bu amaçla kullanılan bir yazılımdır. (Diskiniz üzerinde işlem yapan her yazılım az da olsa disk üzerindeki bilgilere zarar verme riski taşır. Bu tür yazılımlar ile çalışmadan önce önemli olduğunu düşündüğünüz bilgilerin yedeğini almaya özen gösterin). fips, defrag programı kullanıldıktan sonra bölümünüzü sizin belirleyeceğiniz boyutlarda iki bölüme ayırabilir.

Eğer bilgisayaranızı yeni alıyorsanız veya yeni bir disk alıyorsanız, bu diskin tamamını veya bir bölümünü Linux için kullanabilirsiniz. Bu amaçla diskinizde sadece Linux kullanmak istemediğiniz bölümleri ayırmanız (ve gerisini boş bırakmanız) yeterlidir. Linux bölümlerinin Linux altından formatlanması gerekecektir.

Linux sabit disk üzerinde bir bölümden fazlasını kullanabilir. Özel olarak normal dosyaların yazılmadığı, hafıza gereken işler sırasında geçici bir alan olması amacıyla Linux bir takas alanına (swap space) ihtiyaç duyar. Bu takas alanı için en verimlisi sadece takas işlemi için ayrılmış küçük bir disk bölümü yaratmaktır.

Yoğun olarak Linux kullanan yerler için standart olarak dağıtımdan gelen işletim sistemini ayrı bir bölüme yüklemeleri kullanıcı alanları (`/home`) ve sonradan yüklenen yazılımlar (`/usr/local`) için ayrı bir alan ayırmaları tavsiye edilebilir. Bu sayede, işletim sistemi güncellemek son derece kolaylaşır, yeni işletim sistemi yüklerken sadece işletim sisteminin bulunduğu bölüm üzerinde işlem yapılır ve bu sayede kullanıcı alanlarının veya sonradan (dağıtım dışı) yüklenen yazılımların zarar görmeleri engellenebilir.

Her bir bölüm için ne kadar yer ayrılacağı hakkında çok şey yazılmıştır. Ne var ki yazılanların birçoğu sabit disklerin nadiren 200 Mbyte sınırını geçtiği günlerden kalmaktadır. Linux'un kaplayacağı alan, hangi paketleri kullanacağınıza çok bağlıdır. Kabaca her disketin 2-3 Mbyte arasında yer kaplayacağını düşünerek, yüklemek istediğiniz disketleri hesaplayarak kaba bir tablo çıkarabilirsiniz. Tecrübeli bir Linux kullanıcısı hangi yazılımları kullanıp hangilerini kullanmadığını daha iyi belirleyebilecek durumda olacaktır. Dolayısı ile yeni bir kullanıcı ortalama olarak 200 - 300 Mbyte kadar bir yer ayırmak isteyecektir. Bu, günümüzün disk kapasiteleri düşünülünce o kadar büyük bir alan değildir.

Takas alanı konusunda da çok şey yazılmıştır. Birçok kaynak takas bölümü için ayrılması gereken alanının gerçek hafızanın 2 katının biraz fazlası olarak kabul etmektedir. Pratikte 10-60 Mbyte arasında bir alan fazlasıyla yeterli kalmaktadır. Ancak takas bölmeleri 128 megabaytdan daha büyük olamaz. Eğer 128 megabaytdan daha büyük takas alanı gerekiyorsa birden fazla takas bölmesi yaratmalısınız. Toplam 16 tane takas bölmeniz olabilir.

Takas alanı kullanırken, bir seferde daha fazla uygulama çalışmanızı sağlayacak şekilde Linux kullanılmayan sayfaları hafızadan diske taşır. Ancak, takas işlemi genelde yavaş olduğundan gerçek fiziksel hafızanın yerini dolduramaz. Ama çok fazla hafıza isteyen uygulamalar (X Window System gibi) eğer yeteri kadar fiziksel hafızanız yoksa takas alanına bel bağlar.

Tercih olarak, 1.2 Gbyte'lık bir disk üzerinde,

Linux nedir denemek isteyen bir kişi için

- Bölüm 1: DOS 1000 Mbyte
- Bölüm 2: Linux 180-200 Mbyte
- Bölüm 3: Linux 'swap' bölümü 10-20 Mbyte

İşinde arada sırada Linux kullanan birisi için

- Bölüm 1: DOS 400 Mbyte
- Bölüm 2: Linux 400 Mbyte
- Bölüm 3: Linux takas bölümü 32 Mbyte
- Bölüm 4: DOS (DOS altında D: olarak gözükecek) 400 Mbyte

İnternet üzerinde sadece Linux kullanılan bir bilgisayar için

- Bölüm 1: Linux 100 Mbyte
- Bölüm 2: Linux takas bölümü 60 Mbyte
- Bölüm 3: Linux /usr 400 Mbyte
- Bölüm 4: Linux /home 600 Mbyte

2.4 Bilgisayarin Linux ile Açılması

Bilgisayarın sabit diski üzerinde yer ayırdıktan, bir Linux dağıtımı bulduktan sonra artık Linux yüklemek için yapılması gereken, yükleme yapmanıza yardımcı olmaya yetecek şekilde bilgisayarınızı Linux altında çalıştırmaktır. Bu amaçla *boot* ve *root disketi* adı verilen iki disket kullanılması yeterlidir.Bu disketlerden boot disketi bilgisayarınız üzerindeki donanıma uygun bir Linux çekirdeği (kernel) içerir ve bilgisayarın Linux ile açılmasını sağlar, root disketi adı verilen diğeri ise makinanız Linux olarak açıldığı zaman çalıştıracağı yazılımları içeren ve Linux' un çalışması için gereken sistem programlarını içerir. Bu iki disketi, MS-DOS altındaki sistem disketine benzetmek mümkündür.

Boot ve root disketleri, Linux dağıtımı ile birlikte gelirler. Eğer bir CD-ROM dağıtımı ullanıyorsanız, büyük ihtimal disketler CD-ROM ile beraber geleceklerdir. Eğer dağıtımı İnternet'ten alıyorsanız bu disketler bir disket görüntüsü olarak bulunacaklardır. Yapmanız gereken bu disket görüntülerini normal disketlere bu amaç için yazılmış bir yazılımla aktarmak ve açılış disketlerini oluşturmaktır. Bunun için `RAWRITE.EXE` programını kullanabilirsiniz.

Root disketi için genelde bir veya iki seçenek bulunmaktadır. Genelde kullanılan disket `color.gz` adını alır.

Boot disketi için aynı şeyi söylemek mümkün değildir. Zira boot disketi Linux çekirdeğini içermektedir. Her işletim sistemi, o işletim sistemi altında çalışacak olan bilgisayar üzerindeki donanıma erişebilmek için bazı destekler içerir. Ne var ki her donanım kendisine göre bir takım farklılıklar gösterir. Linux bilgisayarınız üzerinde bulunan birçok donanım için destek verebilir, ne var ki tüm donanım desteğini tek bir çekirdekte toplamak çekirdeğin gereksiz yere büyümesine ve hantallaşmasına neden olacaktı (Bilgisayarınızda ses kartı donanımı yoksa çekirdeğin ses kartı desteğine ihtiyacınız olmayacaktır, yapılan sadece gereken destekleri ekleyerek çekirdeğin verimini artırmak demektir). Linux çekirdeği gerektiğinde destek verdiği donanımları destekleyecek şekilde güncellenebilir. Ancak Linux yükleyebilmek için, seçeceğiniz yükleme yöntemine göre bazı donanımlara destek vermesi gereklidir. Örnek vermek gerekirse, NFS üzerinden Linux yüklemek için çekirdek içerisinde mutlaka ağ (network) desteğinin olması gerekmektedir ama ses kartı desteğinin olmasına gerek yoktur. Linux yükledikten sonra derleyeceğiniz bir çekirdeğe ses kartı desteği vermesini sağlayabilirsiniz.

Bir işletim sisteminin sabit diske yüklenme aşamasında kullanıcıya sağlayacağı en büyük kolaylık, deneyimli kullanıcılar için tüm paketleri kurmadan önce sormak, Linux'u bilmeyen ve sabit diskine Linux kurmak isteyen yeni kullanıcılar için ise kurulum aşamasını mümkün olan en az soru ile bitirip daha önceden belirlenmiş birtakım paketleri otomatik olarak yüklemektir.

Çok farklı donanımların olması Linux yükleyebilmek için bir dizi boot disketinin oluşmasına neden olmuştur. Güncel bir Linux dağıtımında hangi boot disketlerinin hangi donanımlara destek verebildiğini görmek için ilgili dağıtımla gelen `README` dosyalarına bakmak gerekecektir. Şu anki Slackware dağıtımı ile gelen boot disketlerinden bazıları

- **bare.i** IDE sabit disklere, sabit disk veya IDE/ATAPI CD-ROM'lardan yükleme yapmak için
- **net.i** IDE sabit disklere, NFS üzerinden yükleme yapmak için
- **scsinet.s** SCSI sabit disklere, NFS üzerinden yükleme yapmak için. Buna ek olarak değişik SCSI denetçileri için 25 kadar değişik boot disketi bulunmaktadır.
- **xt.i** Bu açılış disketinde sadece IDE ve XT sabit disk sürücüleri vardır.

Boot disketleri hakkında geniş bilgi için *Bootdisk-HOWTO* dökümanından yararlanabilirsiniz.

Boot ve root disketlerinizi de elde ettikten sonra artık bilgisayar ilk defa Linux altında çalışmak için hazırdır. Boot disketini takarak sistemi açın (PC'nin açılma sırasının A:,C: olmasına dikkat edin). Disket açılır açılmaz yaklaşık bir sayfalık bir mesaj verecek ve kullanıcıdan ek bir parametre isteyip istemediğini soracaktır. Bu noktada çalışacak olan çekirdeğe birçok ek parametre verilebilir. Eğer herşey yolunda giderse bu noktada özel bir parametre belirtmeye gerek kalmayacaktır. Boot disketi parametreleri hakkında *BootPrompt-HOWTO* içerisinde detaylı bilgi bulabilirsiniz. Bu aşamayı geçtikten sonra çekirdek yüklenmeye başlayacak ve bir dizi mesaj geçecektir. Bu mesajlar çekirdeğinizin bilgisayar üzerindeki donanımları tanıması ve çeşitli hizmetleri çalıştırması ile ilgili mesajlardır. Çekirdeğin donanımınızı ne şekilde tanıdığı bu mesajlardan anlaşılır. Yükleme yapabilmek için çekirdeğin sabit diskinizi ve ağ bağlantısı kullanacaksanız ethernet kartınızı doğru olarak tanımış olması gerekecektir.

Daha sonra kullanıcıdan root disketini yüklemesi için bir mesaj çıkacaktır. Bu aşamada boot disketi yerine root disketi takılmalıdır. Kısa bir yüklemeden sonra bir mesaj çıkacak ve ardından

```
login:
```

mesajı ile karşılaşılacaktır. Tebrikler ! Artık Linux altında çalışmaya başlayabilirsiniz. Bilgisayar şu anda sizden bir kullanıcı ismi beklemektedir. 'root' yazarak sisteme girin.

2.4.1 Ön Hazırlıklar

Slackware dağıtımında yükleme işi için

```
setup
```

isimli bir yazılım bulunmaktadır. Yüklemenin her aşaması bu yazılım tarafından yürütülebilir. Ancak ilk olarak Linux için disk alanlarının tanımlanması gerekmektedir. Bu amaçla

```
fdisk
```

programı kullanılır. Linux altında bir bilgisayar bağlı her türlü cihaza bir dosya gibi erişmek mümkündür. Her cihaza karşılık gelen bir sistem dosyası mevcuttur. Cihazlarla ilgili dosyalar `/dev` dizini altında yer alırlar. Burada IDE sabit diskler "hd" SCSI sabit diskler "sd" olarak isimlendirilirler. Aynı anda bir bilgisayarda birden fazla disk bağlanmış olabilir. Diskler sırayla a b c d olarak isimlendirilirler. Her disk üzerinde birden fazla bölüm yer alabilir. bu bölümler ise 1 2 3 4 olarak numaralandırılırlar. Örnek olarak

- `/dev/hda`, bir numaralı IDE (Primary Master) diski
- `/dev/hda1`, bir numaralı IDE diskin ilk bölümü (DOS altında C:)
- `/dev/hda2`, bir numaralı IDE diskin ikinci bölümünü
- `/dev/hdb`, iki numaralı IDE (Primary Slave) diski
- `/dev/hdc`, üç numaralı IDE (Secondary Master) diski

- `/dev/hdd`, dört numaralı IDE (Secondary Slave) diski
- `/dev/sdb3`, ikinci SCSI sabit diskin üçüncü bölümünü

göstermektedir. Birden fazla sabit diskiniz varsa hangi sabit disk'le ilgileneceğinizi belirtmeniz gerekir. Eğer iki sabit diskiniz varsa fdisk'i kullanırken dikkat edin, her an yanlış bir diski formatlama şansınız var.

```
# fdisk /dev/hda
```

Bu noktadan sonra artık fdisk ile istediğiniz bölümleri tanımlamanız mümkündür.

fdisk komutu harddisk bölümlerinin düzenlenmesi için kullanılan bir yazılımdır. Komut satırından genelde tek harflik komutlarla kullanılır. Eğer komut satırında bir parametre verilmezse fdisk ilk bulduğu disk üzerinde işlem yapacaktır. Birden fazla disk mevcutsa komut satırında istenilen diskin belirtilmesi gerekecektir. Aşağıdaki örnekte fdisk herhangi bir parametre ile çağrılmamış ve sistemde bulunan öncelikli disk olarak bir numaralı SCSI disk (`/dev/sda`) seçilmiştir.

fdisk, komutunun yanında bazı parameteler alır:

- `/fdisk -v` : Fdisk programının sürümü ekrana gelir.
- `fdisk -l` : `/dev/hda, /dev/hdb, /dev/sda, /dev/sdb, /dev/sdc, /dev/sdd, /dev/sde, /dev/sdf, /dev/sdg` ve `/dev/sdh` disklerinin (varsa) bölümlendirme tablosunu ekrana yazar ve çıkar.
- `fdisk -s` <`disk-bölümü`> : Eğer bir DOS bölümü değilse (bölüm numarası 10'dan büyük) , sözkonusu disk bölümünün büyüklüğü bayt cinsinden ekrana yazılır.

Aşağıda, fdisk yazılımının kullanımı hakkında örnek bir fdisk çalışması yeralıyor.

2.4.2 Örnek FDISK Çalışması

Bu örnek içerisinde 1 Gbyte'lık SCSI sabit disk'e sahip bir makine üzerinde LINUX için gerekli kısımların ayrılması adım adım incelenmiştir. Sözkonusu sabit disk üzerinde kullanıcı önceden 400 Mbyte'lık bir kullanıcı alanı tanımlamış ve geri kalan alanı LINUX için ayırmıştı. Düşünülen dağılım:

- 400 Mbyte DOS
- 250 Mbyte LINUX işletim sistemi
- 60 Mbyte Takas alanı

ve geri kalan alan LINUX altında kullanıcı alanı.

fdisk programı çalışır çalışmaz ilk iş olarak mevcut bölümler hakkında bilgi almak için p komutunu (Print Partition Info) kullanıyoruz.

```
Command (m for help): p

Disk /dev/sda: 34 heads, 61 sectors, 1017 cylinders
Units = cylinders of 2074 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1   *        1        1      395   409584+   6  DOS 16-bit >=32M
```

Bu tablo bize sadece tek bölüm ayrıldığını, ayrılan alanın DOS formatında olduğunu, boot edecek bölüm olduğunu belirtiyor. Bölümün adı `/dev/sda1`, yani ilk SCSI sabit disk üzerinde tanımlanan ilk bölüm. İlk iş olarak LINUX işletim sistemi için yeni bir bölüm yaratmalıyız.

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (396-1017): 396
Last cylinder or +size or +sizeM or +sizeK ([396]-1017): +250M
```

n komutu ile kendimize yeni bir bölüm yarattık. İlk seçenek temel bir disk bölümümü yoksa gelişmiş bir disk bölümü üzerinde mi işlem yapacağımızı sordu. Temel bir bölüm için p komutunu girdik. Daha sonra hangi bölümü yaratacağımızı sordu. Halen mevcut 1 bölüm var, bu bölüm ikinci bölüm olacak bu yüzden 2 yazdık.

Bölümün başlangıç adresini giriyoruz. Bu değer otomatik olarak bir önceki bölümün bitiş değerinden hesaplanmaktadır. Sadece onaylıyoruz. Daha sonra istediğimiz boyutu belirtiyoruz. +250M tanımı 250 Mbyte'lık bir kısım istediğimizi belirtiyor. Yarattığımız bu bölümü p komutu ile inceliyoruz

```
Command (m for help): p

Disk /dev/sda: 34 heads, 61 sectors, 1017 cylinders
Units = cylinders of 2074 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1   *        1        1      395   409584+   6  DOS 16-bit >=32M
/dev/sda2          396      396      642   256139   83  Linux native
```

fdisk yaratılan her bölümü otomatik olarak (LINUX native) olarak yaratmaktadır. Şimdi takas alanı için 60 Mbyte'lık 3. temel bölümü tanımlayalım:

```
Command (m for help): n
```

```
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 3
First cylinder (643-1017): 643
Last cylinder or +size or +sizeM or +sizeK ([643]-1017): +60M

Command (m for help): p

Disk /dev/sda: 34 heads, 61 sectors, 1017 cylinders
Units = cylinders of 2074 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1   *        1        1      395   409584+   6  DOS 16-bit >=32M
/dev/sda2          396      396      642   256139   83  Linux native
/dev/sda3          643      643      702    62220   83  Linux native
```

Dikkat edilecek olursa bu bölüm de *LINUX native* olarak tanımlandı. LINUX tarafından takas bölümü olarak kullanılacak olan bölümler farklı bir yapıya sahiptirler ve ayrıca tanımlanmaları gerekmektedir. Bu amaçla t komutu ile herhangi bir bölümün tipini değiştirmek mümkündür. (Tip değiştirmekle o bölümün yapısı (formatı) değişmiş olmuyor, format sonradan yapılan bir işlemdir)

```
Command (m for help): t
Partition number (1-4): 3
Hex code (type L to list codes): 82
Changed system type of partition 3 to 82 (Linux swap)

Command (m for help): p

Disk /dev/sda: 34 heads, 61 sectors, 1017 cylinders
Units = cylinders of 2074 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1   *        1        1      395   409584+   6  DOS 16-bit >=32M
/dev/sda2          396      396      642   256139   83  Linux native
/dev/sda3          643      643      702    62220   82  Linux swap
```

Yukarıda yapılan işlemle 3 numaralı bölümün tipini Linux swap olarak değiştirmiş olduk. Şu anda üç farklı türden alanımız mevcut. Son bölümü de yine n komutu ile ekleriz.

```
Command (m for help): n
Command action
   e   extended
```

```
   p   primary partition (1-4)
p
Partition number (1-4): 4
First cylinder (703-1017): 703
Last cylinder or +size or +sizeM or +sizeK ([703]-1017): 1017

Command (m for help): p

Disk /dev/sda: 34 heads, 61 sectors, 1017 cylinders
Units = cylinders of 2074 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1   *        1        1      395   409584+   6  DOS 16-bit >=32M
/dev/sda2          396      396      642   256139   83  Linux native
/dev/sda3          643      643      702    62220   82  Linux swap
/dev/sda4          703      703     1017   326655   83  Linux native
```

Bu örnekte boyut Mbyte cinsinden verilmedi. Zaten amaç kalan alanı tümüyle kullanıcı alanı olarak ayırmaktı. Bu nedenle son silindirin numarasının girilmesi yeterli oldu. Artık yapılması gereken bu bilginin diske yazılmasıdır. Şu ana kadar yapılan hiç bir değişiklik sistem üzerinde herhangi bir etki yapmamıştır. Ancak bölümleme bilgisi diske yazıldıktan sonra geri dönüş yoktur. **Lütfen yaptığınız değişiklikleri bir kez daha gözden geçirin!**

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
(Reboot to ensure the partition table has been updated.)
Syncing disks.
Reboot your system to ensure the partition table is updated.
```

Sabit diski bölümleme esnasında karşılaşılan bazı sorunlar vardır. Bunların en önemlisi Linux'un nasıl çalıştırılacağı ile ilgilidir. Linux açmak için belli başlı üç yöntem vardır:

- LILO ile bir boot menüsünden
- DOS altından `LOADLIN` isimli bir yazılımla
- Yüklemek için yapıldığı gibi bir boot diskiyle

Son iki yöntem herhangi bir sınırlama getirmezken ilk yöntemin bir sınırlaması vardır. LILO isimli yazılım ile kullanıldığı zaman bilgisayar açıldığı zaman minik bir yazılım çalıştırır ve gerektiğinde Linux çekirdeğini yükler. Ancak çekirdek yüklenmesine kadar geçen süre içerisinde sistemin BIOS komutları çalışır. Bu komutların önemli bir sınırlaması bir disk üzerindeki bir yazılımı belirleyen üç parametreden (silindir sayısı, kafa numarası, sektör numarası) silindir sayısının en fazla 1024 ile sınırlı olmasıdır. Bu bakımdan LILO kullanılacaksa Linux çekirdeğinin yer alacağı bölüm bu 1024 numaralı silindir sınırının altında kalmalıdır. Bu sınırlama tamamıyla DOS'un kullandığı BIOS'un bir sınırlamasıdır. Yeni BIOS'lar bu sınırlamadan kurtulmak için LBA adı verilen bir yöntem kullanırlar. Bu yöntemin temelinde BIOS'ta kafa sayısı için gereksiz yere ayrılan kısımların silindir sayısını belirtmek için kullanılmasıdır. (BIOS, bir sabit disk için 64'e kadar kafa kabul edebilmektedir. Pratik olarak 15 kafadan fazla sabit diskin üretilmesi çok zor olduğu için yapılan kafa sayısını iki veya dört ile çarparak, silindir sayısını iki veya dörde bölmektir.) Böylelikle 1654 silindirli 16 kafalı bir sabit disk, LBA olarak 827 silindirli ve 32 kafalı olarak tanımlanabilmektedir. İkinci durumda bu disk üzerinde tanımlanabilecek her bölümün yukarıda belirtilen sınırlama içerisinde kalacağına dikkat ediniz.

2.4.3 Setup Programı

Disk üzerindeki tanımlamaları da bitirdikten sonra artık setup yazılımı çalıştırılabilir. Setup Linux yüklemek için gereken temel birçok işlemi yapabilir. Ok tuşları yardımı ile menüler arasında gezerek işlemleri tamamlayabilirsiniz. Burada yukarıdan aşağıya doğru bir sıra izleyebilirsiniz.

Şu anki güncel Slackware dağıtımının setup komutu çalıştırılınca ekrana gelen menüleri aşağıdadır.

```
                 Welcome to Slackware Linux Setup.

Hint: If you have trouble using the arrow keys on your keyboard,
you can use ''+'', ''-'', and TAB instead. Which option would you like?

      HELP       Read the Slackware Setup HELP file
      KEYMAP     Remap your keyboard
      MAKE TAGS  Tagfile customization program
      TARGET     Select target directory [now: /]
      SOURCE     Select source media
      DISK SETS  Decide which disk sets you wish to install
      INSTALL    Install selected disk sets
      CONFIGURE  Reconfigure your Linux system
      PKGTOOL    Install or remove packages with Pkgtool
      EXIT       Exit Slackware Linux Setup

                          <  OK  >     <Cancel>
```

- **HELP Menüsü** : Setup programı hakkında bazı ipuçları verecektir.
- **KEYMAP Menüsü** : Bu menü ile Amerikan klavye dışında bir klavye tanımlamak mümkün olacaktır. Henüz türkçe q veya f klavye desteği bulunmamaktadır.
- **MAKE TAGS Menüsü** : Bu menü yardımı ile dağıtım disketlerinde özel uzantılı dosyalar hazırlayarak hangi paketlerin yükleneceğini otomatik olarak belirlemek mümkündür. Bu sayede eğer benzer makinalar yüklenecekse yüklenecek paketler bir kere belirlenir ve bir daha menülerden ekstradan paketlerin seçilmesine gerek kalmaz.

- **ADDSWAP Menüsü** : fdisk ile ayırdığınız takas bölümünü uygun şekilde formatlar ve bu bölümü kullanıma açar. Setup yazılımı hangi disk bölümünün takas bölümü olarak ayrıldığını otomatik olarak bulacaktır. Daha sonra sözkonusu alanları formatlayacak ve bu takas alanını sistem belleğine ekleyecektir. (Her adımda bir onay isteyecektir)
- **TARGET Menüsü** : Linux'un hangi bölüme yükleneceğini belirler. Bu menüye girildiği zaman Linux'un disk formatına (ext2) sahip (sabit disk bölümü numarası 83 olan) tüm disk bölümleri gösterilecek ve içerlerinden hangisine Linux kurulması istenileceği sorulacaktır. Bu aşamadan sonra o disk bölümü kullanıcı isterse formatlanacaktır. Burada iki format seçeneği vardır. Bu seçeneklerin ikincisinde disk önce hatalar için taranacak daha sonra formatlanacaktır. Eğer Linux disk formatında başka bölümler varsa bu bölümlerin kullanılmasının istenip istenmediği sorulacaktır. Bu sayede disk hiyerarşisinin herhangi bir kısmını bu ek disk bölümleri üzerine kurmak mümkündür. Son olarak Linux tarafından desteklenen başka disk bölümleri varsa (Örneğin DOS) bu bölümlere Linux altından erişim yapılmasının istenip istenmediği sorulacak ve bu bölümler için hiyerarşi içerisinde bir dizin atanması istenecektir.
- **SOURCE Menüsü** : Bu menü Linux dağıtımının nerede aranması gerektiğini belirler. Buradaki seçenekler

```
          SOURCE MEDIA SELECTION

    Where do you plan to install Slackware Linux from?


           1 Install from a hard drive partition
           2 Install from floppy disks
           3 Install via NFS
           4 Install from a pre-mounted directory
           5 Install from CD-ROM
```

  - *1 Numaralı seçenek*, Linux dağıtımını bir sabit disk bölümünde aramak için kullanılacaktır. Bu seçenekle örnek olarak DOS kısmında bulunan dağıtım disketlerinden yükleme yapılabilir.
  - *2 Numaralı seçenek*, disketlerden yükleme yapmaktır. Çalışır bir sistemi birkaç disketle oluşturmak mümkündür. Ancak günümüzde pek tercih edilen bir yöntem değildir.
  - *3 Numaralı seçenek*, NFS üzerinden yükleme yapmak için kullanılmaktadır. Burada bilgisayarın bir yerel bilgisayar ağına bağlı olması, Bu bilgisayar ağı üzerindeki bir sunucu üzerinde erişim izni bulunan bir dizin altında dağıtım disketlerinin bulunması gerekmektedir. Bu seçenekle yükleme yapmak için boot disketi içerisinde yer alan diskette ağ desteğinin bulunması gerekmektedir. Bu seçeneğin ardından bilgisayarın (geçici) IP numarası varsa ağ üzerindeki yönlendirici'nin (router-gateway) IP numarası, ağ maskesi (subnet mask), NFS sunucusu IP numarası ve sunucu üstünde dağıtım disketlerinin bulunduğu hiyerarşi gibi ağ ile ilgili parametreler sorulacaktır. Bu soruların cevabını sistem yetkilisinden öğrenmeniz ve onun onayını almanız gerekecektir.

– *4 Numaralı seçenek*, aslında 1 numaralı seçeneğe çok benzemektedir. Aradaki fark bu durumda sistem hiyerarşisine bağlanmış (mounted) bir dizin içerisinde dağıtım disketlerinin bulunmasıdır.
– *5 Numaralı seçenek* ise CD-ROM'dan yükleme yapmak içindir.

- **DISKSETS Menüsü** Artık nereye ve nereden yükleyeceğiniz belirlenmiştir. Sıra yüklemek istediğiniz disket serilerini seçmeye gelmiştir. Disk serileri:

```
CUS         Also prompt for CUSTOM disk sets
A           Base Linux system
AP          Various Applications that do not need X
D           Program Development (C, C++, Lisp, Perl, etc.)
E           GNU Emacs
F           FAQ lists, HOWTO documentation
K           Linux kernel source
N           Networking (TCP/IP, UUCP, Mail, News)
T           TeX typesetting software
TCL         Tcl/Tk script languages
X           XFree86 X Window System
XAP         X Applications
XD          X Server development kit
XV          XView (OpenLook Window Manager, apps)
Y           Games (that do not require X)
```

1. **A Serisi (8 disket)** : Temel işletim sistemi bu disketlerde yer alır. Temel disk hiyerarşisi yaratılır, sistemin çalışması için hayati olan yazılımlar, terminal yazılımları, kabuklar (shell), disk düzenleme yazılımları, kütüphaneler, Linux çalıştırmak için LILO ve LOADLIN bu disketlerdedir.

2. **AP Serisi (5 disket)** : X Window ortamı gerektirmeyen uygulama yazılımlar. Metin editörleri, ghostscript, man sayfaları, midnight commander (Norton commander benzeri bir yazılım) bu disketlerde yer alır.

3. **D Serisi (13 disket)** : Tüm programlama dilleri ve destek yazılımları bu disketlerde yer alır. Eğer kendinize yeni bir Linux çekirdeği derlemeyi düşünüyorsanız bu seriye ihtiyacınız var.

4. **E Serisi (8 disket)** : EMACS editörü.

5. **F Serisi (2 disket)** : Linux hakkında birçok döküman ve açıklama bu disketlerde yer almaktadır. Yeni başlayan birisinin bu disketleri mutlaka yüklemesi gerekir. Söz konusu dökümanlar sıkıştırılmış halde

```
/usr/doc
/usr/doc/faq
/usr/doc/faq/HOWTO
```

dizinlerine yüklenecektir. Birçok kullanıcı cevap aradığı soruların birçoğunun zaten makinelerinde yazılı olduğunu sonradan öğrenince çok şaşırmaktadır. Dökümanlar sıkıştırılmış olduklarından

```
zless
```

gibi sıkıştırılmış dosyaları destekleyen bir yazılımla okunmaları gerekir.

6. **K Serisi (6 disket)** : Çekirdeğin kaynak kodu burada bulunur. Eğer kendi donanımınıza uygun bir çekirdek derlemek istiyorsanız bu seriye muhakkak ihtiyaç duyacaksınız. FTP arşivlerinden kaynak kodu olarak bulacağınız bazı yazılımlar da bu hiyerarşi altında yer alan bazı dosyalara ihtiyaç duyacaklardır.
7. **N Serisi (6 disket)** : Ağ desteği bu disketler ile sağlanmaktadır. E-posta okuma yazılımları, lynx, www sunucusu, haber grubu okuma yazılımları bu disketlerin içerisinde yer alan yazılımlardır.
8. **T Serisi (9 disket)** : TeX. TeX yüklerken üç temel seçenekle karşılaşacaksınız. İlk seri seçenek hangi TeX yardımcı paketlerini isteyeceğinizi sorar, ikinci seçenekler hangi dil için makro tanımları istediğinizi sorar, son seçenek ise yazı tipleri hakkında tercihlerinizi sorar. (Burada gerçekten çok fazla seçenek var)
9. **TCL Serisi (2 disket)** : X Window altında kullanımı basit bir programlama dili ve bu dili ile yazılmış bazı uygulama yazılımları (tkdesk)
10. **X Serisi (16 disket)** : X Window desteği. Bu disketlerin büyük kısmı değişik grafik kartları için X window sunucuları ve yazı karakterlerinden oluşmaktadır. Linux yüklediğiniz bilgisayar üzerindeki grafik kartını bilmeniz ve buna uygun bir sunucu seçmeniz gerekmektedir.
11. **XAP Serisi (4 disket)** : X window altında çeşitli uygulamalar: satranç, gnuplot, xv, xfileman, windows95 benzeri X Window arayüzü bu seriler içerisinde yer almaktadır.
12. **XD Serisi (3 disket)** : Xserver geliştirmek için kütüphaneler ve uygulama yazılımları
13. **XV Serisi (3 disket)** : OpenLook desteği veren yazılımlar. Bu sayede X Window altında Sun bilgisayarlarda yer alan OpenWindows benzeri bir ortam kullanılabilir.
14. **Y Serisi (1 disket)** : Minik birkaç oyun. (özellikle bog) Seçtiğiniz disk serileri ilgi alanınızı genel olarak belirler. Her seri içerisinde birçok yazılım pakedi yer almaktadır. Bu paketlerden istediklerinizi yükleyebilirsiniz.

- **INSTALL Menüsü** Seçtiğiniz disk serilerini belirlediğiniz kaynaktan, belirtilen hedef disk bölümüne aktarır. Disk serileri içerisinde yer alan paketleri ne şekilde yüklemek istediğiniz konusunda birtakım seçenekleriniz olacaktır. Bunlar:

```
NORMAL        Use the default tagfiles for verbose prompting
MENU          Choose package subsystems from interactive menus
CUSTOM        Use custom tagfiles in the package directories
```

```
PATH          Use tagfiles in the subdirectories of a custom path
EXPERT        Choose individual packages from interactive menus
NONE          Use no tagfiles - install everything
```

NORMAL : Sistem her disk serisi içerisindeki paketleri 'gerekli (required)', 'olsa iyi olur (recommended)' ve 'seçime bağlı (optional)' olarak sınıflar. Bu seçenek ile gerekli paketler yüklenir, diğer paketler için kısa bir açıklama yazılır ve kullanıcının fikri sorulur. İlk yüklemeler için tavsiye edilen bir seçenektir. Ancak özellikle T serisi insanın sabrını taşıracak derecede çok ufak tefek paket yüklemekte ve her paketi sormaktadır (mesela klingon fontlarını yüklemek isteyip istemediğinizi soruyor, bilmeyenler için klingonlar uzay yolu dizisindeki kötü uzaylı yaratıklardır :))

MENU ve EXPERT : Bu seçeneklerde her disk serisi yüklenmeye başlanırken o seride yer alan tüm paketler bir menü içerisinde görülür. Kullanıcı istediği paketleri işaretler ve bunların yüklenmesini sağlar.

CUSTOM ve PATH : Daha önce belirtilen TAGFILE dosyaları yardımıyla yükleme yapmak için kullanılır. Bu durumda belirli bir uzantıya sahip dosyalar içerisinde (TAGFILE) yüklenmesi gereken yazılımlar belirtilir. Bu seçenek ile TAGFILE' ların uzantısı belirtilir ve o uzantılı dosyalarda bulunan paketler yüklenir.

NONE : Tembellerin seçeneği. Her şeyi kuracaktır. Sadece belirli paketler için anlamlıdır. Zira birçok paket içerisinde içinden seçilmesi gereken seçenekler mevcuttur. (Mesela Xserver 10 seçenek arasından seçilecektir)

Bu noktadan sonra artık yükleme başlayacaktır.

2.4.4 Konfigürasyon

Yükleme bittikten sonra yapılacak iş artık sisteminiz için tanıtımlarınızı yapmaktır. Bu işlemin ilk aşaması sistemi açacak bir çekirdek belirlemektir. Bu konuda üç seçeneğiniz var:

```
bootdisk    Use the kernel from the installation bootdisk
cdrom       Use a kernel from the Slackware CD
floppy      Install a zimage or bzimage file from a DOS floppy
```

Bootdisk : Bu seçenekte yüklemede kullandığınız çekirdek boot disketinden kopyalanacaktır. Sisteminizi yükleyebildiğinize göre bu çekirdek ile çalışabilirsiniz. Eğer bir değişiklik yapmamışsanız bu anda sürücü içerisinde boot disketi değil root disketi bulunuyor olmalı, değiştirmeyi unutmayın!

CD-ROM : Slackware CD-ROM'unda bulunan önceden derlenmiş çekirdeklerden herhangi birini seçebilirsiniz.

Floppy : herhangi bir DOS disketinde yer alan çekirdeği yüklemenizi sağlar.

Daha sonra sisteminiz için bir boot disketi yaratmak isteyip istemediğiniz sorulacaktır. Ne olursa olsun, elinizin altında her zaman bir boot ve root disketi bulundurmak zorundasınız. Herhangi bir sorun olduğunda sisteminizi açmak için bir boot disketi bulmanız gerekecektir. Bunun için yükleme sırasında kullandığınız boot ve root disketlerini de kullanabilirsiniz.

Ardından setup size modem, mouse, CD-ROM, bulunduğunuz zaman dilimini soracak ve

```
liloconfig
```

yazılımı çalışacaktır. *LILO*, Linux Loader (Linux yükleyicisi) kelimelerinden meydana gelir. LILO Linux yüklemek için kullanılan çok pratik ve etkili bir yazılımdır. Bilgisayar açılır açılmaz, boot eden ilk sabit diskin üzerinde (boot partition) kendini yazar, bilgisayar açılır açılmaz, birden fazla işletim sistemi için seçenek sunabilir. Konfigürasyon sırasında LILO kendisinin nereye yazılacağını sorar, bu seçenekler arasında

```
1. The Master Boot Record of your first hard drive
2. The superblock of your root Linux partition
3. A formatted floppy disk
```

yer alır.

1 numaralı seçenek, birçok uygulamada kullanılacak olan seçenektir. MBR bir bilgisayar açarken ilk bakılan yerdir.

2 numaralı seçenek, MBR'yi kullanmamaktadır. Bunun sebebi, MBR üzerinde bir başka işletim sisteminin benzer bir yazılımının bulunması olabilir. (örneğin OS/2 Bootmanager)

3 numaralı seçenek'te LILO kendisini bir diskete yükleyecektir. Bu disketten açıldığı zaman menü ortaya çıkacaktır.

Daha sonra boot işlemi sırasında çekirdeğe gönderilecek ekstra parametreler belirtilebilir. Birçok sistem için bu tür bir parametre gereksizdir. Bu parametre boot diski ile açıldığı zaman sorulan parametrenin aynısıdır. Sonraki seçenek LILO'nun yükleme sırasındaki davranışını belirler. LILO konfigürasyonu sırasında birden fazla boot edebilecek sabit disk bölümü tanımlanabilir. Shift tuşuna basıldığı zaman LILO mevcut bölümler için bir liste çıkaracaktır. LILO için tanımlı dört davranış vardır:

```
1 -- None, don't wait at all - boot straight into the first OS
2 -- 5 seconds
3 -- 30 seconds
4 -- Present a prompt and wait until a choice is made without timing out
```

1 numaralı seçenek hiç beklemeden doğrudan listede belirtilen ilk işletim sistemini yükleyecektir. Sadece Linux bulunan bilgisayarlar için kullanılan seçenek budur.

2 ve 3 numaralı seçenekler sırasıyla 5 ve 30 saniye beklerler, eğer bu süre içerisinde Shift tuşuna basılmazsa ilk sırada yer alan işletim sistemini yüklerler.

4 numaralı seçenek bir işletim sistemi seçilene kadar bekler.

Daha sonra sırasıyla yüklenmesini tercih ettiğiniz disk bölümlerini tanıtabilirsiniz. LILO her bölüm için sizden ayıredici bir kelime isteyecektir. LILO yükleme anında sizden komut beklerken bu kelimeye göre işletim sistemi yükleyecetir.

LILO hakkında bazı örnekler için *LILO-NASIL* dökümanından yararlanabilirsiniz.

2.4.5 Makinayı Açmak

Linux yükleme işlemi sona erdi. Artık bundan sonra sıra makinanızı Linux çalışacak şekilde çalışmasını sağlamaktır. Bunun için temel olarak iki değişik yöntem mevcuttur:

- **LILO** : en çok kullanılan en pratik açılış şekli. Burada bilgisayar açıldığı zaman isteğe göre bir süre bekler ve bu esnada shift,tab veya control tuşuna basılırsa birden fazla işletim sistemi ile çalıştırma seçeneği sunar.
- **LOADLIN** : Dos altından çalışan bir yazılımdır. DOS altında çalışırken Linux yüklemenize yarar. Eğer kurulum aşamasında `LOADLIN` pakedini (A serisi disketler içinde) seçmişseniz bu paket `/root` dizini altında `LOADLIN.ZIP` ismi ile kaydedilmiş olacaktır. Yapmanız gereken bu yazılımı ve mevcut çekirdeğinizi (`/vmlinuz`) DOS kısmına aktarmaktır.

Aşağıda yer alan örnekte,

1. IDE bir sabit disk kullanıldığı
2. `/dev/hda1`'in DOS olduğu
3. `/dev/hda3` üzerine Linux yüklendiği
4. DOS kısmına (henüz) Linux altından erişilemediği
5. Makinanın Linux olarak açılmış olduğu

varsayılmıştır.

```
linux:~# ls /root
lodlin16.txt  lodlin16.zip
linux:~# mkdir /dos
linux:~# mount -t msdos /dev/hda1 /dos
linux:~# cd /dos
linux:/dos# unzip /root/lodlin16.zip
Archive:  lodlin16.zip
 creating: loadlin/
  inflating: loadlin/readme.1st
  inflating: loadlin/loadlin.exe
  inflating: loadlin/copying
  inflating: loadlin/test.par
```

```
  inflating: loadlin/linux.bat
 extracting: loadlin/initrd.tgz
  inflating: loadlin/files
   creating: loadlin/doc/
  inflating: loadlin/doc/changes
  inflating: loadlin/doc/announce.txt
  inflating: loadlin/doc/lodlin16.lsm
  inflating: loadlin/doc/quicksta.rt
  inflating: loadlin/doc/initrd.txt
  inflating: loadlin/doc/manual.txt
  inflating: loadlin/doc/params.doc
   creating: loadlin/src/
  inflating: loadlin/src/loadlin.asm
  inflating: loadlin/src/loadlina.asm
  inflating: loadlin/src/loadlini.asm
  inflating: loadlin/src/loadlinj.asm
  inflating: loadlin/src/loadlinm.asm
  inflating: loadlin/src/makefile
 extracting: loadlin/src/srclinux.tgz
  inflating: loadlin/src/pgadjust.asm
linux:/dos# cp /vmlinuz /dos/loadlin/zimage
linux:/dos# cd loadlin
linux:/dos/loadlin# ls
copying*      files*        linux.bat*    readme.1st*   test.par*
doc/          initrd.tgz*   loadlin.exe*  src/          zimage
linux:/dos/loadlin# cat linux.bat

rem  First, ensure any unwritten disk buffers are flushed:
smartdrv /C
rem  Start the LOADLIN process:
c:\loadlin\loadlin  c:\loadlin\zimage  root=/dev/hda3  ro  vga=3
linux:/dos/loadlin# cp linux.bat /dos
```

Burada verilen örnekte yapılan sırasıyla

- Sabit diskin DOS kısmını Linux altından ulaşılır yapmak
- `/root` altında yer alan `lodlin16.zip` isimli dosyayı DOS diski altında açmak. Bu işlem `C:\LOADLIN` isimli bir dizin yaratıp içerisine gereken dosyaları yerleştirmektedir.
- Çalışan çekirdeği (`/vmlinuz`) bu dizine kopyalamak ve `LINUX.BAT` dosyasını kendimize uygun hale getirmek. Bu dosya `C:\LOADLIN\ZIMAGE` isimli bir çekirdeği kullanarak `/dev/hda3` bölümünde Linux çalıştırmak için kullanılır.
- `LINUX.BAT` dosyasını PATH içinde tanımlı bir yere kopyalamak, mesela `C:\DOS`.

Artık bilgisayar DOS açıldıktan sonra

```
linux
```

komutu ile tekrar Linux çalıştırmak mümkün olacaktır.

Bilgisayarınıza Linux yüklediğiniz disketlerle her zaman bilgisayarınızı Linux olarak açıp çalışabilirsiniz.

Tavsiye edilen mümkünse LILO kullanmaktır. LOADLIN ise bir ikinci alternatif olarak çokça kullanılır. Boot disketlerine ise genelde son çare olarak başvurulur.

Bütün bu adımlardan sonra artık elinizde çalışmaya hazır bir Linux makine vardır. Makinayı kapatıp tekrar açın. Ekranda çekirdek mesajları geçtikten sonra

```
login :
```

belirecektir. Buraya `root` yazın ve sisteme girin. İlk denemeniz için

```
# shutdown -rf now
```

yazabilirsiniz. Linux bir makine çalıştığı sürece hafıza içerisinde birçok tampon bellek açar. Mümkün olduğu kadar makineyi kapama tuşuna basarak kapatmayın. Shutdown komutu işletim sisteminin tampon belleklerde tuttuğu bilgileri güncellemesini sağlayacaktır. `-r` parametresi sistemin reboot etmesini sağlayacaktır. Bilgisayarı kapatmak için

```
# shutdown -hf now
```

komutunu kullanabilirsiniz. Burada yer alan h parametresi sistemin "halt" edeceğini (tamamen kilitlenme) ve bir daha açılmayacağını belirtecektir.

Eğer herşey yolunda giderse makinanızın başında oturup çalışmaya başlayabilirsiniz.

Sistemde çalışmaya başlamak üzere ilk iş olarak kendinize çalışmak amacıyla bir kullanıcı tanımlayın. Sistemde başka kullanıcı olacaksa, onlar için de hesap açacaksınız. Kullanıcı hesabı açmak için

```
# adduser
```

komutu size bu konuda yardımcı olacaktır. `adduser` (veya `useradd`) komutu , kullanıcı ismi, isim ve soyad, GID (grup kimliği), UID (kullanıcı kimliği) gibi birtakım sorular soracaktır. Bu komut hakkında detaylı bilgiyi *Sistem Yönetimi* bölümü altında bulabilirsiniz. root kullanıcısı sistem üzerinde sınırsız yetkiye sahip olduğundan sistem dosyalarını kazara değiştirmenize veya silmenize sebep olabilir.

Şimdi yeni hesabınızla sisteme girebilirsiniz. Alt F1 'den Alt F6 'ya kadar olan tuşlarla birden çok ekranda (sanal ekranlar) aynı anda çalışabilirsiniz.

Bu noktadan sonra bazı uygulamaların ayarlamalarını yapmanız gerekecektir. Artık Linux yüklemek ile ilgili bir sorunuzunun kalmamış olmasını umuyorum. DOS kullanıcılarına Linux hakkında bilgi veren sevimli bir döküman için *DOS2Linux Mini-HOWTO* iyi bir başlangıç olabilir.

Açılış esnasında makinanızın ismi `/etc/rc.d/rc.M` dosyasında belirlenir. Bu dosyayı uygun şekilde değiştirerek makinanızın ismini de yeniden tanımlayabilirsiniz. Makinanızın ilk ismi `darkstar` olacaktır. Eğer TCP/IP ağ üzerinde çalışıyorsanız, `/etc/HOSTNAME` dosyasının içeriğini değiştirerek veya hostname komutu kullanılara makina ismi de değiştirilebilir.

Bunların dışında konfigürasyon gerektiren birçok durum vardır. Bunlar için *NET-3 HOWTO* dosyasını iyice okumalısınız.

2.5 LILO Konfigürasyonu

Eğer linux ve OS/2 ile birlikte aynı disk üzerinde çalışmayacaksanız, LILO'yu öncelikli yükleyici olarak diskinizdeki diğer işletim sistemlerinin açılışında rahatlıkla kullanabilirsiniz. Daha önce de belirttiğim gibi OS/2'nin kendine özgü önyükleyicisi olduğundan birincil önyükleyici olarak bunu kullanın, LILO'yu ise linux açmak için çalıştırın.

Slackware LILO kurulum işlemi, bazı özel durumlar için yetersiz kalmaktadır. Bu gibi durumlarda dosyanın el yardımıyla konfigürasyonu gereklidir.

LILO'yu değiştirmek için `/etc/lilo.conf` dosyası üzerinde değişiklikler yapmalısınız. Aşağıda bir LILO konfigürasyon örneği görülüyor. Burada Linux bölümü `/dev/hda2`'de , MS-DOS bölümü ise `/dev/hdb1`'de (ikinci sabit diskin ilk bölümü) bulunuyor.

```
# LILO, /dev/hda uzerinde kurulacak.
boot = /dev/hda

# Linux yuklemek icin
image = /vmlinuz      (Cekirdegin diskteki ismi /vmlinuz)
  label = linux       (Buna "linux" ismini ver. Acilis aninda
                      ekranda "linux" yazisi gorunur)
  root = /dev/hda2    (Kok dosya sistemi olarak /dev/hda2 kullan)
  vga = ask           (VGA ekran modu icin kullanicidan komut bekle)

# MS-DOS yuklemek icin
other = /dev/hdb1     (MS-DOS bolumu)
  label = msdos       ("msdos" ismini ver)
  table = /dev/hdb    (ikinci surucu icin bolumlendirme tablosu)
```

Yukarıda bir örneği görülen `/etc/lilo.conf` dosyasında gerekli değişiklik veya düzeltmeleri yaptıktan sonra `/sbin/lilo` dosyasını root olarak çalıştırın. Bu komut LILO önyükleyicisini sabit diske kuracaktır. Bundan sonra her yeni çekirdek derlemenin ardından dosyada uygun değişiklikleri yapıp aynı komutu çalıştırmalısınız.

Şimdi sistemi sabit disten tekrar açabilirsiniz. LILO, `/etc/lilo.conf` dosyasındaki ilk işletim sistemini yükleyecektir. Başka bir sistemle açmak için önyükleme menüsünü ekrana getirmelisiniz. Bunun için makina açılırken shift veya kontrol tuşlarına basılı tutun. Ekrana bir açılış istemcisi gelecektir:

```
Boot :
```

Burada ya açılmasını istediğiniz işletim sisteminin ismini yazın (ilk kurulum sırasında MS-DOS veya linux), ya da tab tuşuna basarak listeyi ekrana getirin.

OS/2 kullanabilmek için Linux önyükleyicisini ikincil kullanmanız gerekecektir. Linux'u OS/2 önyükleyicisinden açmalısınız. Bunu yapmak için Linux sabit disk bölümünü OS/2 fdisk kullanarak yaratın. Ardından sözkonusu bölümü FAT veya HPFS olarak formatlayın ki OS/2 tanıyabilsin.

Bundan sonra LILO'yu linux dosya sisteminin olduğu bölüme kurun. (yukarıdaki örnekte `/dev/hda2`). `/etc/lilo.conf` dosyasının linux için kullanılan satırlarının son durumu şu şekilde olacaktır:

```
boot = /dev/hda2

image = /vmlinuz
  label = linux
  root = /dev/hda2
```

`/sbin/lilo` dosyasının çalıştırılmasının ardından OS/2 önyükleyicisine linux bölümünü tanıtın. Benzer yordamları, sorun çıkaran işletim sistemleri için de kullanabilirsiniz.

2.6 Sorun Çıktığında

Tabii ki temennimiz sorun çıkmaması, ama her nedense her zaman bu satırları okuyacak birkaç kişi olacaktır. Linux üzerinde kendi ihtiyacınıza göre ayarlanamayacak bir parametre yok gibidir. Bu özelliği sayesinde son derece esnek bir çalışma ortamı haline gelebilir. Aynı özellik kişilerin çoğu zaman kendi çözümlerini bulmalarını da beraberinde getirmektedir. Kendi bilgisayarınızda Linux çalışmak için bir miktar döküman karıştırmanız kaçınılmazdır.

Çıkabilecek en önemli sorun bilgisayarın açılmamasıdır. Bunun birçok sebebi olabilir. Açılış esnasında

1. İlk olarak LILO çalışır.
2. Çekirdek yüklenir.
3. Hizmet veren yazılımlar teker teker çalışmaya başlarlar.

Her aşama birçok satırda durumunu belirtecektir. Çalışan yazılımlar veya yazılım parçalarının her biri birbirinden bağımsız olduğu için açılış sırasında geldiğiniz nokta çok önemlidir. LILO çalışmadığında veya çekirdek yüklenirken takılırsa boot disketi ile rahatlıkla sistemi açabilirsiniz. Örnek olarak Linux yüklü disk bölümünüzün /dev/hda2 olduğunu varsayalım. Boot disketi parametre istediğinde

```
mount root=/dev/hda2
```

yazmanız yeterli olacaktır. Bu durumda boot disketinde yer alan çekirdek ile belirttiğiniz bölümde yer alan Linux hiyerarşisi açılacaktır. Çalışan bu sisteminiz içerisinde artık hatanın kaynağını daha rahat bulabilirsiniz.

Çekirdeğin yükleme esanasında takılması büyük ölçüde çekirdeğin donanımı doğru belirleyememesinden çıkar. Mesela ethernet kartınızı yanlış tanımış olabilir. Bunu çözmenin temel yöntemi çekirdek içerisinde kullanmayacağınız donanımlara ilişkin destekleri kaldırmak (Tüm ethernet kartlarını destekleyen bir çekirdek yerine sadece kullanmanızın muhtemel olduğu ethernet kartlarına destek veren bir çekirdek derlemek) veya çekirdeğe yardımcı olabilecek açılış parametreleri vermek.

Çekirdek derlemek hakkında *Kernel-HOWTO* ve boot parametreleri hakkında *BootPromt-HOWTO* yardımcı olabilirler. Çalışan sisteminizde bir arıza meydana gelip de makina aniden çalışmamaya başlarsa en son yaptığınız değişiklikleri gözden geçirin.

Ve son olarak, sistemin çalışmaması her zaman Linux'tan kaynaklanmayabilir, donanım ile ilgili sorunlar da yaşayabilirsiniz. Rasgele davranışlar, durup dururken çakılmalar, panik mesajları altında, bozuk sabit diskler, normalden yüksek frekansta çalıştırılan işlemciler ve sistem saatine göre yavaş kaçan veya bozuk RAM'lar yatabilir.

2.7 Başlangıçta

Bu satırları okuduğunuza göre çalışır durumda bir Linux sisteminiz var demektir. Hala silmediniz mi yoksa? Şaka bir yana, büyük badireler atlattınız. Önünüze sayısız engeller çıkabilirdi. Kurarken elektrikler kesilebilirdi, sabit disk çizilebilirdi veya yanlış bölümü formatlayabilirdiniz. Özellikle Linux Slackware dağıtımını kurmak, dökümantasyon olmadan veya hatırlı bir arkadaşın yardımı olmaksızın gerçekten zordur.

Bu bölümde Linux'a küçük bir giriş yapılacak, daha sonra da UNIX kullanmayanlar sistemle tanıştıracaktır. Bölüm boyunca yapmanız gereken korkmadan sistem üzerinde kolaylıkla gezinti yapmak ve dosyaları kurcalamaktır.

Önceki bölünde sisteme girebilmek için şifresi olmayan "root" kullanıcıyı kullanmıştınız. Bu kullanıcı sistemde en fazla yetkiye sahip kullanıcı olup sistem görevlisi (sorumlusu) adını alır. Eğer root dışında bir kullanıcı hesabı tanımlanmışsa onu kullanın. Şifre yazıldıktan sonra komut istemcisine, yani kısaca kabuk dediğimiz programa girilir. Şifre yazılırken, başkalarının görmemesi için ekrana basılmaz.

Kullanıcı isimleri veya şifrelerde büyük ve küçük harfler arasında fark vardır. Root, root ve ROOT, farklı kullanıcılara işaret eder. Klavyenin en sağındaki Caps Lock tuşunun yanmadığına emin olun.

Sisteme ilk girişte, aşağıdaki gibi bir satırla karşılaşacaksınız.

```
Welcome to Linux 1.2.13.

linux login: root
password:
Last login: Thu Feb 13 12:46:35 on tty1
Linux 1.2.13.
You have mail.
linux:~#
```

Genellikle komut istemcisinin sonundaki karakter, root kullanıcısı için `#`, diğer kullanıcılar için `$` olur. Bu karakterden önce de makina ismi yeralır. MS-DOS'ta olduğu gibi burada UNIX komutlarını girebileceğiniz kabuk (shell) üzerindesiniz.

Şifreyi değiştirmek için kullanılan komut `passwd` 'dir. Bir kullanıcı sadece kendi şifresini değiştirirken root'a herkesin şifresini değiştirme yetkisi verilmiştir. Herhangi bir sistemde hesap şifrenizi unutursanız, bunu sadece root değiştirebilir. Root iken `passwd` yazın ve enter tuşuna basın.

```
linux:~# passwd
Changing password for root
Enter new password: *******
Re-type new password: ******
Password changed.
linux:~#
```

Şifrenizi iyi saklayın. Root şifresini ele geçiren birisi sistemde istediği değişikliği yapabilir. Şifre seçimi için *Linux İşletim Sisteminde Güvenlik* konu başlığına göz gezdirin.

Linux komutları hakkında bilgi almak için `man` komutu kullanılır. Eğer kurulum aşamasında man dosyalarının kopyalanması sorusuna olumlu yanıt verilmişse bunlar `/usr/man` dizini altında bulunurlar. Örneğin `passwd` komutu hakkında daha detaylı bilgi almak için

```
$ man passwd
```

yazın. Tüm man sayfaları `/usr/man` dizini altında 8 ayrı dizinde saklanır (man1 .. man8). Bazı komutların man dosyaları birden fazla dizin altında bulunur, bir dosya komut hakkında temel bilgi verirken diğeri sistem programcılarına yönelik olabilir. Örnek olarak `mount` komutu, hem 2, hem de 8 numaralı man dosyalarıyla birlikte arşivlenmiştir. C programlayıcısı, mount komutuna ulaşmak için

```
$ man 2 mount
```

yazarken normal kullanıcı,

```
$ man 8 mount
```

yazmalıdır. Bunun yanında başlığında belirli bir anahtar sözcüğü içeren tüm man dosyalarını araştırmak için `apropos` komutu kullanılır.

Her komut, bir veya birden çok parametre alabilir. Örnek olarak,

```
find . -name "*.txt" -print
```

komutu, bulunduğunuz yerden itibaren tüm dosyaları araştıracak ve bunların arasından sonu `.txt` ile bitenleri ekrana basacaktır. Parametreler genel olarak "-" işaretleri ve bu işaretten sonra gelen parametre ismi ile belirtilirler.

2.8 Linux Komut Yapısı

UNIX ve benzeri işletim sitemlerinde kullanıcının komut yazmasını sağlayan, bu komutları yorumlayarak gerekli işlemleri yapan programlara *kabuk* (shell) adı verilir. UNIX'te bir kullanıcı bir dizi kabuktan istediğini seçebilir. Kabuklar ile ilgili ayrıntılı bilgi ileride verilecektir. Kullandığınız kabuk ne olursa olsun, gerek kabuktan kaynaklanan, gerekse UNIX komutlarının hepsinin uyduğu bazı standartlardan kaynaklanan bazı geleneksel yapılar vardır. Bunları bilmeniz ilk kez duyduğunuz bir komutun kullanımını bile kolayca çıkarabilmenizin yanı sıra, bir dizi genel hatadan kaçınmanızı da sağlayacaktır.

- UNIX'te (ve Linux'ta) bütün komutlar ve dosya isimlerinde büyük/küçük harf ayrımı önemlidir. Sistem komutlarının ve dosyaların çoğu küçük harfle yazılır.
- Komut ve dosya adlarında kullanacağınız bazı karakterlerin gerek dosya ve dizin yapısı, gerekse kabuk ve diğer komutlar nedeniyle bazı özel anlamları vardır. Bu karakterlerden yeri geldikçe söz edilecektir. Örneğin, '/' karakteri hiçbir dosya adında bulunamaz (dosya ve dizinler için ayraç olarak kullanıldığından). '-' ile başlayan bir dosya oluşturulabilir olsa da silmeye kalktığında yeni bir UNIX kullanıcısının başına dert açabilir.
- UNIX'te komutlara seçenek verirken seçenekten önce '-' karakteri kullanılır. Örneğin `ls -l`
- UNIX komutları tersi istenmedikçe girdilerini *standart girdiden* (klavye) alır, çıktılarını *standart çıktıya* (ekran) yazar. Bu özellik ileride anlatılacak olan yönlendirme ve boru (pipe) öperatorleri ile birlikte komut satırından birçok işlemi kolayca yapmanızı sağlar.
- UNIX kabukları komut satırından verilen komutu çalıştırmadan önce bir dizi karakteri yorumlayarak dosya ad(lar)ına çevirirler. Bu karakterler:
  - `*` 0 dahil herhangi bir sayıda karakter yerine geçer. Örneğin `rm *` komutu bütün dosyaları siler, `ls -l a*` komutu 'a' ile başlayan dosyaların listesini verir.
  - `?` tek bir karakter yerine geçer. Örneğin `??` adı iki karakterden oluşan bütün dosyalar anlamına gelir.
  - [] karakterleri arasında yazılan liste içindeki herhangi bir harfe dönüştürülür. Örneğin `cp *[abc] /tmp` komutu 'a', 'b' ya da 'c' ile biten bütün dosyaları `/tmp` dizinine

kopyalayacaktır. liste içinde aralarına '-' işareti koyarak aralıklar verebilirsiniz. Örneğin, `[A-Z]*` büyük harfle başlayan bütün dosyalar anlamına gelir. Liste içindeki '^' karakteri sonrasında belirtilen liste dışındaki bütün karakterler anlamına gelir. Örneğin `*[^0-9]*` adında rakam olmayan herhangi bir dosya anlamına gelecektir.

2.9 Dosya ve Dizin Yapısı

UNIX altında bazı karakterlerin özel anlamları vardır. Dizin ve dosya isimlerinin başında nokta olması durumunda bu dosyalar gizli dosya haline gelir ve parametresiz yazılan `ls` komutuyla görünmez.

Dosya ve dizin isimleri 255 karakteri aşamazlar.

Sisteme girince önceden tanımlanmış bir dizin altında bulunursunuz. Bu dizin normal kullanıcılar için genellikle `/home/` ve ardından gelen kullanıcı dizini ismidir. Bulunduğunuz dizinin ismini görmek için `pwd` (print working directory) yazın. Temel dosya ve dizin kavramları hakkında detaylı bilgi alabilmek için MS-DOS veya, en iyisi bir UNIX kitabı edinin.

```
$ pwd
/home/gorkem
$
```

UNIX komut yapısı DOS'a çok benzer. Dizin değiştirmek için `cd` , dizin yaratmak için `mkdir` komutlarını sistemde sıkça kullanılır.

```
$ cd /
$ pwd
/
```

Hiyerarşik bir sıraya sahip olan UNIX'te en üstte `/` dizini (kök dizin) yeralır. Sistemdeki tüm diğer dosya ve dizinler bunun altında toplanırlar.

```
~
```

işareti, kullanıcının ev dizinini gösterir. Ev dizinine geçin ve `mkdir` komutu ile `benim` isimli bir dizin yaratın. Dizini patikasını, bir başka deyişle kök dizininden itibaren ismini vererek de yaratabilirsiniz.

```
$ cd ~
$ pwd
/home/gorkem
$ mkdir /home/gorkem/benim
```

2.9.1 Dosya Listesi ve İçeriklerinin Görüntülenmesi

Dosya ve dizinleri görebilmek için `ls` komutu kullanılır. Parametresiz yazarak bulunduğunuz dizin hakkında bilgi alın. Linux altında değişik uzantılı dosyalar okunabilirliği artırmak amacıyla farklı renkte görülecektir. `/etc/DIR_COLORS` dosyası, dosya renklerini ayarlar. Bu dosyayı değiştirerek belirli dosyalar için istenilen renkleri göstermesini sağlayabilirsiniz.

```
linux:~$ ls
README  article.txt     mail       typescript
adres   linux           perl
```

`ls` komutu `-a` parametresiyle birlikte kullanılırsa normalde görükmeyen ve nokta karakteriyle başlayan dosya ve dizinler de görükür.

```
linux:~$ ls -a
.               .bashrc         .term           article.txt     typescript
..              .kermrc         .xinitrc                linux
.Xdefaults      .less           README  mail
.bash_history   .lessrc         adres           perl
```

`-l` parametresi de dosyalar hakkında tüm bilgiyi verir. Bunlar, dosyanın sahibi, ne zaman yaratıldığı, sahibi ve grubu gibi bilgilerdir. Dosya izinleri hakkında daha geniş bilgi bir sonraki bölümde verilmiştir.

```
linux:~$ ls -al
total 91
drwxr-xr-x   6 gorkem   users        1024 Feb 13 12:56 .
drwxr-xr-x   4 root     root         1024 Jan  7  1980 ..
-rw-r--r--   1 gorkem   users         390 Feb 13 12:56 .Xdefaults
-rw-r--r--   1 gorkem   ftpadm        230 Feb 13 12:57 .bash_history
-rw-r--r--   1 gorkem   users           1 Feb 13 12:57 .bashrc
-rw-r--r--   1 gorkem   users         163 Nov 24  1993 .kermrc
-rw-r--r--   1 gorkem   users          34 Nov 24  1993 .less
-rw-r--r--   1 gorkem   users         114 Nov 24  1993 .lessrc
drwxr-xr-x   2 gorkem   users        1024 Jan  7  1980 .term
-rw-r--r--   1 gorkem   users          87 Feb 13 12:56 .xinitrc
-rw-r--r--   1 gorkem   users       26264 Feb 13 12:53 README
-rw-r--r--   1 gorkem   users        2795 Feb 13 12:55 adres
-rw-r--r--   1 gorkem   users       47970 Feb 13 12:53 article.txt
drwxr-xr-x   2 gorkem   users        1024 Feb 13 12:54 linux
drwxr-xr-x   2 gorkem   users        1024 Feb 13 12:54 mail
drwxr-xr-x   2 gorkem   users        1024 Feb 13 12:54 perl
-rw-r--r--   1 gorkem   users           0 Feb 13 12:57 typescript
```

Yukarıda her iki parametrenin de birleştirilerek birlikte kullanımı yeralmaktadır. Dosyaların içeriklerini görmek için kullanılan birkaç komuttan en pratiği `less`'tir. Ok tuşları ile dosya içinde hareket edebilir ve q karakteri ile dosyadan çıkarsınız. Dosyanın içeriği birden fazla sayfadan ibaretse dosya sayfalar halinde ekrana gelir. Tüm dosyayı ekrana basmak için `cat` kullanın.

```
$ cat README
```

Bir dosyanın başından veya sonundan itibaren belirli miktarda satırı ekrana getirmek de mümkündür. Bunun için `head` ve `tail` komutları kullanıcının hizmetine sunulmuştur.

```
linux:~$ head -6 README  (dosyanin ilk 6 satirini ekrana basar)
This is the README file for the 28 August 1994 public release of the
Info-ZIP group's portable UnZip zipfile-extraction program (and related
utilities).

unzip512.zip    portable UnZip, version 5.12, source code distribution
unzip512.tar.Z  same as above, but compress'd tar format

linux:~$ tail -3 README (dosyanin son 3 satirini ekrana getirir)

-- Greg Roelofs (Cave Newt), UnZip maintainer/container/explainer and
   developer guy, with inspiration from David Kirschbaum
```

2.9.2 Dosyaların Kopyalanması

Dosyaları kopyalamak için `cp` ve bir yerden başka bir dizine almak için `mv` komutu kullanılır. `mv` komutu, aynı zamanda dosya isimlerini değiştirmek üzere de işletilebilir.

```
linux:~$ cp article.txt /tmp
(article.txt isimli dosyayi /tmp dizinine kopyala)
linux:~$ mv article.txt /tmp/article
(article.txt isimli dosyayi /tmp dizini altina ismini article olarak degistirerek koy
```

Kopyalama işlemi sadece dosyalar üzerinde değil, dizinler üzerinde de yapılabilir. Farklı dosya sistemleri üzerinde olmamak kaydıyla bir dizin ve altındaki herşeyi, başka bir dizine kopyalayabilir veya hareket ettirebilirsiniz. Kopyalarken bu işlem için `-R` parametresi kullanılır, `mv` komutu için `-r` parametresine gerek yoktur.

```
$ cp -R /home/gorkem/temp /tmp
(/home/gorkem/temp dizinini ve icindeki her dosyayi /tmp altina kopyala)
$ mv article.txt ~/benim
(article.txt isimli dosyayi calisma dizini altindaki benim dizinine kopyala)
```

2.9.3 Dosyaların Silinmesi

Bir daha kullanılmayacak olan dosyalar, `rm` komutuyla silinebilir. Dosyaları silerken iki kere düşünün, zira Linux altında silinen bir dosyanın geri dönüşü olmaz. Eğer bu tür hataları çok yapıyorsanız `rm` komutunu `-i` parametresi ile birlikte kullanın, bu sayede Linux, dosyayı silmeden önce kullanıcının da onayını alır.

```
linux:~$ rm -i README
rm: remove ‘README’? y  (sileyim mi ?)
linux:~$
```

Eğer dosya ismi `-` karakteri ile başlıyor ise, `rm` komutunu kullandığınız zaman dosya ismi bir parametre olarak algılanacak ve hata verecektir. Bunun için dosya isminden önce `--` karakterlerini yerleştirin.

```
linux:~$ rm -- -dosya
```

Bir dizin, boş iken `rmdir` komutu ile silinebilir, eğer boş değilse bu komut işe yaramaz. Onun yerine `rm` komutunu `-r` parametresiyle kullanın.

```
linux:~$ rm mail
rm: mail: is a directory
linux:~$ rm -rf linux/         (linux dizinini icindekilerle birlikte silinir)
```

`-f` parametresi ile kullanıcının onayını almadan tüm dosyaları tek adımda silmek mümkündür. Bulunduğunuz dizindeki tüm dosyaları silmek için * karakterini kullanın.

```
$ rm *
```

Chapter 3

Kabuk İşlemleri

Sisteme girdiğiniz anda kabuk programının çalıştırıldığından bahsetmiştik. Bu andan sonra yapacağınız tüm işlemler bu kabuk programı tarafından yönetilir ve denetlenir. Kabuk, klavyeden girilen komutları çalıştırarak bir arabirim görevi yapar. UNIX (ve Linux) altında geliştirilen sayısız kabuk çeşidi ve bunların herbirinin kendine ait özelliği vardır. Her programcı kendi zevkine hitap eden kabuğu seçebilir, sistemde yeralan kullanıcılar farklı kabuklar üzerinde çalışabilirler.

O an hangi kabuk üzerinde yeraldığınızı öğrenmek için `echo $SHELL` yazın.

```
$ echo $SHELL
/bin/bash
```

Sıkça kullanılan kabuklar,

sh (Shell ya da Bourne Shell): İlk UNIX kabuğu.

ksh (Korn Shell): sh uyumlu, birçok ek programlama özelliği içeren bir kabuk. Bu kabuk da yaygın bir kitle tarafından kullanılıyor.

bash(Bourne Again Shell): Kullanım kolaylığı bakımından en çok rağbet gören bash, GNU tarafından kaynak kodu olarak dağıtılıyor. *bash sh* ve *ksh* uyumluluğunu korurken, özellikle etkileşimli kullanıma yönelik (komut tamamlama, gibi) birçok yenilik de içerir. Bu yazının hazırlandığı 1997 Temmuz ayı başlarında Bash 2.0.1 sürümü çıkmıştı.

csh (C shell): Berkeley Üniversitesi'nde geliştirilen csh'in C diline benzer bir programlama yapısı vardır.

tcsh: csh'ın biraz geliştirilmiş hali.

Yukarıdaki kabuk programlarından `sh, ksh ve bash` birbirleriyle uyumludur ve yukarıda en ilkelden en gelişmişe göre sıralanmışlardır. Aynı durum `csh ve tcsh` için de geçerlidir. Kullanılan shell genellikle kişisel bir tercih nedeni olsa da, bütün sistem scriptleri `sh` ile yazıldığından, sistem yöneticilerine `sh` ailesinden bir kabuk kullanmaları önerilir.

Kabuk programları genellikle `/bin` dizini altında yeralır. Kullanıcının sisteme girerken hangi kabuğu kullanacağı `/etc/passwd` dosyasında yeralır. Bunu değiştirmek için `chsh` (change shell) komutunu kullanabilirsiniz.

```
$ chsh
Password:
Changing the login shell for cagri
Enter the new value, or press return for the default

        Login Shell [/bin/sh]: /bin/bash
$
```

Sisteminizde *NIS* kullanılıyorsa `chsh` yerine `yppasswd -s` komutunu kullanmalısınız.

3.1 Yönlendirme

Kullanıcı, ekrana yazdığı bir komutun neler yaptığını en rahat şekilde komut tarafından ekrana yönlendirilen bilgilerden anlayabilir. Program, kullanıcıyı bilgilendirme amacıyla mümkün olduğu kadar çok, fakat ortalığı fazla karıştırmamak için de mümkün olduğu kadar az bilgiyi ekrana vermelidir.

3.2 Standart Girdi, Çıktı ve Hata

Linux'ta, programın ekrana yazılan bilgiyi iki sınıf altında toplayabiliriz. Birincisi, olağandışı bir durumu bildiren standart hata, diğeri de her türlü verinin yazıldığı standart çıktı. Program çalıştığı andan itibaren bu iki kanal üzerinden akan bilgiler, programın çalıştığı sanal terminale yazılırlar. Program girdilerini ise standart girdi aracı olan klavyeden alır.

Eğer bu bilgiler bir ekran boyundan (25 satır) fazla tutuyorsa bazı satırlar programcının gözünden kaçabilir. Bunu önlemek amacıyla standart çıktı ve hata bir dosyaya yazılacak şekilde ayarlanabilir. Yönlendirme olarak da bilinen bu işlem UNIX altında (DOS'ta olduğu gibi) > karakteri ile gerçekleştirilir.

Örnek olarak o an bulunduğunuz dizinde yeralan dosyaları ekrana getirin :

```
$ ls -al
```

Bu komut, standart çıktı olarak dosyaların bilgilerini ekrana getirecektir. Bu çıktıyı, bir dosyaya yönlendirelim ve dosyanın içeriğine göz atalım:

```
linux:~$ ls -al > liste
linux:~$ cat liste
total 16
drwxr-xr-x   5 gorkem   users        1024 Feb 13 13:10 .
```

```
drwxr-xr-x   4 root     root        1024 Jan  7  1980 ..
-rw-r--r--   1 gorkem   users        390 Feb 13 12:56 .Xdefaults
-rw-r--r--   1 gorkem   ftpadm      2016 Feb 13 13:09 .bash_history
-rw-r--r--   1 gorkem   users          1 Feb 13 12:57 .bashrc
-rw-r--r--   1 gorkem   users        163 Nov 24  1993 .kermrc
-rw-r--r--   1 gorkem   users         34 Nov 24  1993 .less
-rw-r--r--   1 gorkem   users        114 Nov 24  1993 .lessrc
drwxr-xr-x   2 gorkem   users       1024 Jan  7  1980 .term
-rw-r--r--   1 gorkem   users         87 Feb 13 12:56 .xinitrc
-rw-r--r--   1 gorkem   users       2795 Feb 13 13:06 adres
-rw-r--r--   1 gorkem   users          0 Feb 13 13:10 liste
drwxr-xr-x   2 gorkem   users       1024 Feb 13 12:54 mail
drwxr-xr-x   2 gorkem   users       1024 Feb 13 12:54 perl
-rw-r--r--   1 gorkem   users          0 Feb 13 13:10 typescript
linux:~$
```

> karakteri standart hatayı dosyaya yönlendirmez. Bu işlem için 2> kullanılır. Ama hatayı görebilmek için, hata yaratan bir komut yazmalıyız, değil mi ?

```
$ ls /deneme
/deneme : No such file or directory
$ ls /deneme 2> hata
$ cat hata
/deneme : No such file or directory
```

Aşağıdaki komutun işletilmesinin ardından standart çıktı `oku1` dosyasına, standart hata ise `oku2` dosyasına yazılacaktır. Bu dosyaları komutu çalıştırdıktan sonra incelemek suretiyle neler olup bittiğini anlamak mümkün olur.

```
$ mkdir ~/deneme           (deneme isimli bir dizin yarat)
$ touch ~/deneme/gecici    (gecici isimli bir dosya yarat)
$ cat ~/deneme 2>oku2 >oku1
```

Kabuk, standart çıktı ve standart girdi için sırayla 2 ve 1 numaralarının kullanımına izin verir. Yukarıda yeralan son komutta, standart hata mesajları için 2 kullanılarak hataların `oku2` dosyasına yazılmıştır. Aşağıda, çekirdek derlemek için sürekli kullandığım program yeralıyor. Yönlendirme sayesinde ekrana gelmesi gereken mesajlar kullanılmayan 9. sanal konsola yönlendiriliyor.

```
# make config
# make dep > /dev/tty9
# make clean > /dev/tty9
# time make zImage > /dev/tty9
```

Son satırdaki `time` komutu, kendinden sonra gelen komutun ne kadar zaman içinde çalıştırıldığını gösterir. Çekirdek derlemede geçen zaman, makinanın gücü hakkında bir fikir verebilir. Peki ne hata, ne de çıktıyı ekranda görmek istemiyorsam ne yapmalıyım ? Bunun için standart çıktı ve hatayı biraraya getirerek yönlendirilen her çıktının kaybolduğu “kara deliğe” atmak yeterlidir:

```
$ ls /deneme > /dev/null 2>&1
```

Yukarıdaki komutun yazılış sırasına dikkat edin.

Standart çıktı ya da *standart hata*yı yönlendirirken, > işareti kullanırsanız: dosya yoksa, oluşturulur ve komutun çıktısı dosyaya yazılır. Dosya varsa, içeriği yok olur, ve komutun çıktısı dosyanın yeni içeriği olur. Var olan bir dosyayının eski içeriğini tamamen silmek değil de komutun çıktısını dosyaya eklemek istiyorsanız >> kullanmalısınız. Bu durumda dosya varsa komutun çıktısı dosyanın eski içeriği korunarak sonuna eklenir, dosya yoksa oluşturulur ve komutun çıktısı dosyaya yazılır. Örneğin:

```
$ echo deneme1 >>deneme.txt
$ cat deneme.txt
deneme1
$ echo deneme2 >>deneme.txt
$ cat deneme.txt
deneme1
deneme2
$
```

Örnekte görüldügü gibi ilk komut `deneme.txt` dosyasını oluşturdu. İkincisi ise oluşan dosyanın içeriğini koruyarak ikinci komutun çıktısını bu dosyanın sonuna ekliyor.

Standart hata ve çıktıya ek olarak UNIX'in desteklediği bir yönlendirme daha vardır: Standart girdi sayesinde bir dosyayı oluşturan satırlar, bir komut veya programa yönlendirilebilir. Daha önce bir metin editor kullanarak hazırlamış olduğumuz raporu patrona kısa yoldan göndermek için,

```
$ mail -s "rapor" patron < rapor.txt
```

Dosyanın içeriği, mail komutuna girdi olmuş ve `rapor.txt` dosyası `patron` kullanıcısına “rapor” konu başlığıyla e-posta ile gönderilmiştir.

3.3 Boru (pipe) İşlemleri

Bazı durumlarda, bir komutun çıktısı diğer bir komuta yönlendirilebilir. Başka bir deyişle, komutun standart çıktısını bir dosyaya değil, bu çıktıyı işleyecek başka bir komuta yönlendirmek istiyorsunuz. Bu amaçla UNIX altında (yine DOS'ta olduğu gibi) boru (`|`) karakteri kullanılır. Bu karakter, kendinden önce gelen komut veya komut serisinin çıktısını, kendinden sonra gelen komuta gönderir. Örneğin bir dizinde yeralan tüm dosyaları yazıcıya aktarmak için,

```
$ ls -al | lpr
```

komutları kullanılabilir. Artık `ls -al` komutunun ekrana vermesi gereken tüm bilgiler, `lpr` komutu aracılığıyla yazıcıya gönderilmiştir. İlk komutun standart çıktısı, ikinci komuta standart girdi olarak atanır. Diğer bir örnekte, `README` dosyasında kaç satır olduğu bulunuyor. Bir dosyadaki veya komut çıktısındaki satır, karakter ve kelime sayılarını bulmak için `wc` komutunu kullanabilirsiniz.

```
$ who | wc -l
40
```

Yönlendirme ve pipe işlemleri bazen insanın kafasını karıştırabilir. Yönlendirme, bir programdan bir dosyaya yapılabilir, fakat bir programdan başka bir programa yönlendirme yapamazsınız. Benzer şekilde, iki dosyanın arasında pipe işlemi uygulamak mümkün olmaz.

Şimdi de biraz karışık bir örnek:

```
# ps -aux | grep inetd | grep -v grep|awk '{print $2}' | xargs kill -1
```

Yukarıdaki örnek zorlama bir örnek değil, bir Linux sistem yöneticisinin her an kullanması gerekebilecek türden. Bu örneği burada ayrıntılı olarak açıklamayacağız, sadece *pipe* kullanarak ne kadar fazla sayıda komutun birbirine bağlanabildiğini göstemek amacıyla verdik. Bu ve benzeri komutların ayrıntılı açıklamaların kabuk programlamayla ilgili bölümde bulabilirsiniz.

3.4 Çokgörevlilik

UNIX'in en büyük silahlarından biri süreçlerdir. Her süreç sistemde bağımsız çalışan, birbirini etkilemeyen ve herbiri kendi kaynağını kullanan programdır. Süreçler arkaplanda veya kabuğun kontrolünde çalışabilir. Çekirdek, her sürecin kullandığı sistem kaynağından haberdar olur ve bu kaynakların süreçler arasında adilce paylaşılmasından sorumludur.

Bir süreç, aksi belirtilmedikçe çalıştığı süre içinde klavyeden bilgi alır ve ekrana bilgi verir.

Kullanıcıların haberi bile olmadan çalışan süreçler, Linux makinasındaki G/Ç işlemlerini gerçekleştirebilmek için sürekli faaliyet içinde bulunurlar. Onlarca süreçten bazıları kullanıcıların sisteme girmesini sağlarken (getty) bazıları da WWW ve FTP gibi İnternet tabanlı istekleri yerine getirir (httpd, ftpd).

3.4.1 Arkaplanda Çalıştırma

Bir komutu arkaplanda çalıştırmak için, komutun sonuna `&` karakteri getirin. Komutu girdikten sonra tekrar kabuk istemcisine düşecek ve kaldığınız yerden devam edebileceksiniz. Program, arkaplanda diğer süreçlerle çakışmadan bir süre çalışır ve işi bittiğinde durur.

```
$ sort buyuk_dosya > buyuk_dosya.sirali &
[1] 772
```

Komutun arkaplana atılmasından sonra ekranda yeralan 1 , sürecin sıra numarasını, 772 sayısı ise süreç kimliğini (Process ID) gösterir. Her program, sistem kaynaklarını biraz daha azalttığından Linux'çuların deyimiyle makina *yavaşlar*.

Çalışan programların listesini görmek için `ps` komutunu kullanın. Hiç parametre vermeden yazılırsa sadece üzerinde çalışılan kullanıcının süreçleri ekrana gelir. Sık kullanılan bir başka parametre dizisi de `uax` tir. Bu sayede sistemdeki o anda çalışan tüm programlar daha ayrıntılı bir şekilde ekrana listelenir.

```
$ ps
PID TTY STAT  TIME COMMAND
   76 v02 S     0:00 -bash
  111 v02 R     0:00 ps

$ ps -aux
USER       PID %CPU %MEM SIZE  RSS TTY STAT START   TIME COMMAND
bin         63  0.0  5.5   64  364  ?  S    11:12   0:00 /usr/sbin/rpc.portmap
gorkem      76  0.0  9.7  101  644 v02 S    11:12   0:00 -bash
gorkem     112  0.0  5.0   59  332 v02 R    11:16   0:00 ps -aux
root         1  0.0  5.0   56  332  ?  S    11:12   0:00 init [5]
root         6  0.0  4.2   35  284  ?  S    11:12   0:00 bdflush (daemon)
root         7  0.0  4.2   35  284  ?  S    11:12   0:00 update (bdflush)
root        48  0.0  5.1   45  340  ?  S    11:12   0:00 /usr/sbin/crond -l10
root        59  0.0  5.5   53  364  ?  S    11:12   0:00 /usr/sbin/syslogd
root        61  0.0  5.0   44  336  ?  S    11:12   0:00 /usr/sbin/klogd
root        65  0.0  5.5   62  364  ?  S    11:12   0:00 /usr/sbin/inetd
root        67  0.0  5.8   79  388  ?  S    11:12   0:00 /usr/sbin/rpc.mountd
root        69  0.0  6.0   88  400  ?  S    11:12   0:00 /usr/sbin/rpc.nfsd
root        75  0.1  9.9  115  660 v01 S    11:12   0:00 -bash
root        77  0.0  4.6   52  304 v03 S    11:12   0:00 /sbin/agetty 38400 tty3
root        78  0.0  4.6   52  304 v04 S    11:12   0:00 /sbin/agetty 38400 tty4
root        79  0.0  4.6   52  304 v05 S    11:12   0:00 /sbin/agetty 38400 tty5
root        80  0.0  4.6   52  304 v06 S    11:12   0:00 /sbin/agetty 38400 tty6
root        81  0.0  5.5   42  368  ?  S    11:12   0:00 gpm -t ms
```

Burada yeralan bilgiler sırasıyla :

- *PID:* süreç numarası. Her sürecin farklı bir numara alması sistem tarafından düzenlenir.
- *%CPU:* İşlemcinin % olarak ne kadarı süreç tarafından kullanılıyor ?
- *%MEM:* % olarak hafızada kapladığı yer
- *SIZE:* Programın toplam kullandığı bellek alanı (kilobyte)
- *RSS:* Programın bellekte kapladığı gerçel alan. Bu değer paylaşılan sistem kütüphaneleri (shared library) nedeniyle genellikle `SIZE` değerinden küçüktür.

- *TTY:* sürecin çalıştırıldığı (sanal) terminal
- *STAT:* süreç o sırada ne yapıyor ?
  - *S:* Uyumakta, bu durum genellikle sürecin bir G/Ç işlemi (örneğin fare'nin hareketi ya da ağdan gelecek veriler) beklemesi sırasında olur.
  - *R:* Çalışabilir. Çalışabilir süreçler, gereksinim duydukları bütün sistem kaynaklarına sahip olan süreclerdir. Çekirdek (tek işlemcili bir makinada) belirli aralıklarla bu süreçlerden sadece birini çalıştırır.
  - *T:* Durmuş. Bu durum sürecin kullanıcı tarafından (örneğin `Control-Z` tuşlarına basılarak) durdurulmasında görülür.
  - *Z:* Zombie. Bu tür süreçler, kendisini çalıştıran sürece mesaj bırakamadıklarından takılmış süreçlerdir. *Öldürülemezler* ancak sitem kaynaklarını da kullanmazlar.
  - *D:* Kesilemez uykuda. Bazı G/Ç işlemleri o sırada bu işlemi yapan sürecin kesilmemesini gerektirir. Böyle durumdaki süreçleri de ilgili G/Ç işlemi bitinceye kadar öldüremezsiniz.
- *START:* sürecin çalıştığı zaman
- *COMMAND:* komut satırından girilen programın ismi

`ps` komutuna bilgisayar üzerinde çalıştığınız her an ihtiyaç duymanız mümkündür, bu yüzden çeşitli parametrelerle kullanmayı öğrenmelisiniz.

3.4.2 Klavye Üzerinden Kesinti

Linux (ve UNIX) altında, klavyeden bazı tuş kombinasyonları yardımıyla çalışmakta olan program kesintiye uğratılabilir.

Klavyeden kabuk komut satırına yazılan bir programın uzun sürmesi halinde, eğer daha önceden komutun arkasına `&` işareti koyup arkaplanda çalışır halde bırakılmamışsa, klavyeden yapılan bir müdahele ile durdurulup arkaplanda çalışır hale getirilebilir. Örnek vermek gerekirse, uzun sürmesi beklenen bir komutu klavyeden yazalım ve ardından `Control-Z` tuşlarına basalım:

```
linux:/etc/rc.d# find /usr -name "o*" -print
/usr/bin/od
/usr/lib/lilo/doc/other.fig
/usr/lib/lilo/doc/other.tex
/usr/man/man1/od.1.gz

[1]+  Stopped                 find /usr -name "o*" -print
linux:/etc/rc.d#
```

Bu esnada sürecin çalışmasına ara verilmiş, fakat program tamamen durmamıştır. Programın çalışmasını arkaplanda sürdürmek için `bg` komutunu kullanın.

```
$ bg
```

Bu komutu tekrar komut satırında çalışacak ve klavyeden bilgi girilecek şekilde terminale bağlamak için `fg` yazın:

```
$ fg
```

Bir programı çalıştırmaya başlattıktan sonra tamamen durdurmaya karar vermişseniz klavyeden `Control-C` tuşlarına basın.

Durdurulup arkaplanda çalışmaya yönlendirilen süreçlere kısaca *görev* ismi verilir. Tüm görevleri görebilmek için

```
$ jobs
```

yazın. Görevler, birden fazla oldukları zaman sıra numarası ile belirtilirler.

3.4.3 Süreçlerin Sona Erdirilmesi

Her an çalışan süreçlerden biri veya birkaçı, beklenmedik döngüye girebilir. Bunun sonucu olarak sistemin kaynaklarını, özellikle hafızayı tüketici bir duruma gelebilir. Bu tür kısır döngüye giren süreçleri bulup, eğer hayati önem taşımıyorlarsa 'öldürmek' gerekir. Süreci öldürmekten kasıt, programı tamamen durdurarak sistemle ilişkisini kesmektir. Bu sayede programın hafızada kapladığı bölge serbest kalacak, çekirdek de hafıza düzenlemesini tekrar yaparak başka süreçlere daha fazla yer ayıracaktır. Bir süreci öldürmek için `kill` komutu kullanılır. Yukarıdaki 67 numaralı sürece ait `/usr/sbin/rpc.mountd` programını öldürmek için şunları yazın :

```
$ kill 67
```

Birçok süreç sizden bu mesajı aldıktan sonra, dosya sistemi üzerinde yarattığı geçici dosyaları, dosyalar üzerine koyduğu kilitleri temizlemek gibi yapması gereken işlemleri yaptiktan sonra çalışmasına son verecektir. Eğer öldürmeye çalıştığınız süreç herhangi bir nedenle takılmışsa ve bu komuta tepki vermiyorsa aşağıdakini deneyin:

```
$ kill -9 67
```

Artık programın sistemle ilişkisi tamamen kesilmiştir. Kill komutu, `-9` seçeneğiyle sürece 9 numaralı sinyali gönderir. Bu sinyali alan sürecin yukarıda sözü edilen iki özel durum dışında çalışmayı sürdürmesi olanaksızdır. `-9` seçeneğini özellikle sistem süreçleri üzerinde gerekmedikçe kullanmamalısınız. Diğer sinyaller ve anlamları yeri geldikçe diğer bölümlerde anlatılacaktır.

Chapter 4

Dosya ve Dizin İşlemleri

Tıpkı MS-DOS'ta olduğu gibi Linux dosya yapısının da hiyerarşik bir yapıya sahip olduğunu söylemiştik. Temel dosya bilgisi önceki konularda anlatıldığından burada sadece dosya ve dizinlerle ilgili özelliklere değinilecektir.

4.1 Erişim Hakları

Erişim hakları, Linux dosya sistemi güvenliğinin belkemiğini oluşturur. Her dosyaya ayrı verilebilen erişim izinleri sayesinde çok daha rahat bir sistem yönetimi gerçekleştirilebilir. Bu, konuya sadece sistem görevlisi tarafından yaklaşıldığı zaman çıkartılabilecek bir sonuçtur. Kullanıcı bazında erişim hakları bazen daha da anlamlı olabilir. Yanlış kullanıldığında hoş olmayan süprizlere yol açabilir. Linux altında üç çeşit erişim hakkı vardır :

- **Okuma izni** : Dosyanın okuma izni varsa içeriği görülebilir, dizinin okuma izni varsa içerdiği dosyaların listesi alınabilir.
- **Yazma izni** : Dosyanın yazma izni varsa dosyayı değiştirebilir veya silebilirsiniz. Dizine yazma izni verildiğinde dizin altındaki dosyalar yazılabilir veya silinebilir.
- **Çalıştırma izni** : Dosyayı çalıştırma hakkını verir.

Bir dosya veya dizin ilk yaratıldığı anda Linux tarafından öntanımlı bazı izinler verilir. Genellikle bu izin çalıştırma ve okumadır. Dosyanın oluşma anında verilen izini değiştirmek için erişim yetki kalıbı olarak da bilinen `umask` komutu kullanılır. Daha fazla bilgi için komuta ait man dosyasına bakın.

Erişim haklarının dışında bir dosyanın üç izin düzeyi daha vardır. Bunlar, dosyanın sahibi, dosyanın grubu ve diğer kullanıcılardır. Dosyanın sahibi, o dosyayı oluşturan kişidir. Her kullanıcının bir grubu da olduğu için, dosya oluşturulurken kullanıcı hangi grupta ise dosya da o gruba ait olacaktır. Dosyanın sahibi olmayan ve grubu da dosyanın grubuyla uyuşmayan sistemdeki kullanıcılar "diğer" sınıfına girerler. Gruplar hakkında daha fazla bilgi için *Linux Sistem Yönetimi* konusuna bakın.

4.1.1 Dosya İzinlerinin Değiştirilmesi

Dosya izin bilgilerini görebilmek için `ls` komutu, `-l` parametresiyle kullanılır. Aşağıdaki dosya üzerindeki erişim haklarına göz atalım :

```
-rwxr-xr-x   2 gorkem   users          182 Feb 12 03:58 deneme
```

Dosya bilgisinin en sol kısmında izin hakları yeralır. En baştaki "-" işareti bunun düz dosya olduğunu belirtir. "gorkem", dosyanın sahibi; "users" ise grubudur. Ardından sırayla uzunluk, son değiştirilme tarihi ve dosya ismi gibi bilgiler gelir. Son harf kümesine dosyanın modu da denir.

Solda yeralan r,w ve x karakterleri sırayla okuma, çalıştırma ve yazma haklarını belirtir. En soldaki "-" işaretini ayırdığımız zaman geri kalan harfleri üç adet üçlü grup haline getirelim:

```
rwx     r-x    r-x
```

Sırayla birinci harf kümesi dosya sahibinin izinlerini, ikinci harf kümesi grup izinleri ve son küme de diğer kullanıcıların izinlerini belirtir. Buna göre yukarıdaki dosyada

- `rwx` : kullanıcı okuyabilir, yazabilir, çalıştırabilir
- `r-x` : grup okuyabilir, çalıştırabilir fakat yazamaz.
- `r-x` : diğerleri okuyabilir, çalıştırabilir fakat yazamaz.

Sol baştaki karakter "d" olsaydı, bir dizini inceliyor olacaktık.

Dosyanın izinlerini değiştirmek için `chmod` komutu kullanılır. Bu komutu kullanırken hangi izin düzeyine (kullanıcı, grup veya diğerleri) hangi izinlerin verileceği veya kaldırılacağı yazılır.

Örnek olarak yukarıdaki dosyanın okuma iznini kaldırmak için

```
$ chmod -r deneme
```

kullanılabilir. "-" işareti, iznin kaldırılacağını belirtir. İzin vermek için ise "+" yazın. Dosyanın yeni görüntüsü şöyle olur :

```
--wx--x--x   2 gorkem   users          182 Feb 12 03:58 deneme
```

Sadece grubun, sadece kullanıcının veya sadece diğerlerinin erişim hakkını değiştirebilmek için her izin düzeyi için bir harf tanımlanmıştır. Kullanıcı için "u", grup için "g" ve diğerleri için "o" yazarak belirli bir izin düzeyi için erişim hakkını değiştirmek mümkün olur. Bu üç harf, izinden hemen önce yazılır.

```
$ chmod u+x deneme              (dosyanin sahibi calistirabilir)
$ chmod o+r t2sac               (digerleri okuyabilir)
$ chmod g-w deneme              (dosyanin grubu yazamaz)
```

Her izin için aynı zamanda sayılar da tanımlanmıştır. Dosyanın erişim izinlerinideğiştirmek için her izne ait sayıları kullanabilirsiniz. Aşağıdaki tabloda, her izin düzeyi (dosyanın sahibi, grubu ve diğerleri) için üçer tane olmak üzere toplam 9 sayıdan oluşan izin numaralarını görebilirsiniz. Bir izni vermek için sırayla numaraları toplayın. Örneğin sahibinin okuması (400), yazması (200), çalıştırması (100), grubun okuması (40), çalıştırması (10) ve diğerlerinin okuması (4) ve çalıştırması (1) için kullanılan rakamların toplamı 755'tir.

```
$ chmod 755 deneme
```

400	sahibi okur
200	sahibi yazar
100	sahibi çalıştırır
040	grubu okur
020	grubu yazar
010	grubu çalıştırır
004	diğerleri okur
002	diğerleri yazar
001	diğerleri çalıştırır

Başka bir değişiklikle konuyu pekiştirelim. Sahibinin okuması (400), yazması (100), grubun okuması (40), diğerlerinin okuması (4) için izin numarasını bulmak için bu cümlede parantez içinde yeralan sayıları toplarsak 644 yapacaktır.

```
$ chmod 644 deneme
```

Aynen dosyalarda olduğu gibi, bir dizinin de sahibi ve grubu vardır.

4.2 Dosyanın Sahibinin ve Grubunun Değiştirilmesi

Bir dosyanın sahibini sadece sistemdeki *root* kullanıcı değiştirebilme yetkisine sahiptir. Dosya sahibinin değiştirilmesi için `chown` komutu kullanılabilir. Yukarıdaki dosyanın sahibini *ozgur* isimli kullanıcı yapalım.

```
# chown ozgur deneme
```

Dosyanın yeni hali şöyle olur :

```
-rwxr-xr-x   2 ozgur   users          182 Feb 12 03:58 deneme
```

`chgrp` komutu yardımıyla da dosyanın grubunu değiştirilebilir. Yukarıdaki dosyanın grubunu www yapmak için

```
# chgrp www deneme
```

komutu kullanılabilir. `chown` komutu, dosyanın hem kullanıcısını, hem de grubunu değiştirme özelliğine sahiptir. chown komutundan sonra kullanıcı ve grup isimlerini aralarında bir nokta kalacak şekilde yazın.

```
$ chown ozgur.www deneme
$ ls -l deneme
-rwxr-xr-x   2 ozgur   www          182 Feb 12 03:58 deneme
```

4.3 Diğer Dosya Sistemlerinin Kullanılması

Bir Linux işletim sisteminde dosyaların ve dizinlerin herbirisi sabit disk, CDROM veya diğer bilgi saklamaya yarayan cihazların birer parçası olan "dosya sistemlerinde" tutulur. Linux'un desteklediği pek çok dosya sistemi vardır ve bunların her birisinin veri depolarken kullandıkları yöntem birbirinden farklıdır.

Linux'ta her dosya sistemi farklı bir dizinde tutulur ve bu dizinlerin bilgisi `/etc/fstab` dosyasında yeralır. Açılış anında bu dosyada hangi dosya sistemlerinin hangi dizinler altına yerleştirileceği okunur ve buna göre makina açılır. Linux, kendisi için özel olarak geliştirilen ext2 dosya sistemini kullanır.

Bir sistem görevlisi olarak çeşitli dosya sistemlerini tanımalı, bu dosya sistemleri üzerinde hata oluşması durumunda zararı en aza indirecek yolu bilmelisiniz. Bilgisayar başında geçireceğiniz zamanınızın büyük bir bölümünü dosya sistemleri üzerinde harcayacaksınız.

4.3.1 mount İşlemi

Bir dosya sisteminin kullanılabilmesi için boş bir dizin altına yerleştirilmesi gerekir. Ancak bu gerçekleştirilirse sözkonusu dosya sistemini oluşturan dosyalar üzerinde işlem yapılabilir. Bu işlem mount komutu yardımıyla yapılır.

`mount` komutunun şu şekilde kullanılır :

```
mount -t <tip> <nereye> <aygit>
```

tip : Dosya sistemi çeşidi. Örnek dosya sistem çeşitleri arasında ext2, minix, msdos sayılabilir. Dosya sistemleri hakkında daha geniş bilgi için Çekirdek konfigürasyonu konusuna bakın. Genellikle mount programı ne tür bir diske eriştiğini otomatik olarak anlayacaktır.

nereye : mount edilecek dosya sistemi hangi dizin altına yerleştirileceğini belirtir.

aygıt : Üzerinde dosya sisteminin bulunduğu sabit disk, CD-ROM benzeri aygıt.

Linux'a bağlı olan tüm donanımlara ait bir giriş, `/dev` dizini altında bulunur. Bu dizin altındaki dosyalar özel olarak tanımlanmıştır ve bunlara kısaca "düğüm" adı verilir. Örneğin `/dev/hda2`, birinci IDE sabit diskin ikinci bölümünü; `/dev/tty1` ise birinci sanal konsolu gösterir.

Çeşitli sabit disklere ait `/dev` düğümleri, *Linux Kurulumu* bölümünde detaylı olarak anlatılmıştı. mount programı bu düğümlere ihtiyaç duyacaktır. Bu yüzden bir dosya sistemini erişilebilir hale getirebilmek için hangi düğüm ismine sahip olduğunu bilmelisiniz. Örnek olarak CD-ROM için `/dev/cdrom`, birinci disket sürücü için `/dev/fd0` gibi.

CD-ROM sürücüsünü `/mnt` altına mount etmek için

```
# mount -t iso9660 /dev/cdrom /mnt
```

yazın. Eğer çekirdekte CD-ROM dosya sistemi desteği varsa ve sistem açılırken çekirdek CD-ROM'u tanımışsa `/mnt` dizini altına CD'deki dosya sistemi yerleştirilir. Buna benzer şekilde MS-DOS disket içeren disket sürücüyü ve üzerinde ext2 dosya sistemi bulunan ikinci IDE sabit diskin birinci bölümünü sırasıyla `/mnt/disket` ve `/mnt/disk` dizinlerine bindirmek için

```
# mount -t msdos /dev/fd0 /mnt/disket
```

ve

```
# mount -t ext2 /dev/hdb1 /mnt/disk
```

yazılabilir. Bir mount işleminin tersini yapmak ve dosya sistemini erişilemez kılmak için `umount` komutu kullanılır. `/disk` altındaki `/dev/hdb1` sabit disk bölümünü umount etmek için

```
# umount /disk
```

veya

```
# umount /dev/hdb1
```

kullanılır. Her iki yazım şekli de kabul edilir. Son olarak o an sistemde bindirilmiş halde bulunan tüm dosya sistemlerini görmek için `mount` komutunu parametresiz yazın. Bu komut hakkında daha detaylı bilgi ve aldığı diğer parametreler için man sayfasına bakın.

`mount` ve `umount` komutları ile bir dizin yapısını disk üzerinde konumlandırırken veya ayırırken üzerinde işlem yapılan dizinde bulunmamanız, ya da o dizinde başka bir süreç çalışıyor olmaması gereklidir. Aksi takdirde aşağıdaki hata mesajını alırsınız:

```
# pwd
/disk
# umount /disk
umount: /dev/hdb1: device is busy
```

Yapmanız gereken, ayıracağınız dizinden dışarı çıkıp tekrar denemek. Diğer bir seçenek de `fuser` komutunu kullanmak. Bu komut, ayırmak istediğiniz dizini kullanan tüm süreçleri ekranda gösterir. Böylelikle rahatlıkla yukarıdaki örnekteki gibi `umount` işlemini gerçekleştirebilirsiniz. `fuser` komutunun -km seçeneğiyle o dizini kullanan süreçleri öldürülebilir.

```
# cd /
# fuser /disk
/disk:                  78c
# fuser -km /disk
# umount /disk
```

`fuser` komutunu kullanırken dikkatli olun, zira sistemi kilitlenme noktasına getirebilirsiniz.

4.3.2 Dosya Sistemi Bilgileri

Linux açılırken `mount` komutu yardımıyla root dosya sistemine ekleyeceği her değişik dosya sistemini `/etc/fstab` dosyasından okur ve işleme koyar. Aşağıda örnek bir fstab dosyası yeralıyor.

```
/dev/hda2       /           ext2        defaults  1 1
/swap           none        swap        defaults  1 1
/dev/cdrom      /cdrom      iso9660     defaults  1 1
none            /proc       proc        defaults  1 1
/dev/hda1       /dos        msdos       defaults  1 1
```

Çekirdek açılış mesajları içinde

```
VFS: Mounted root (ext2 filesystem) readonly.
```

gibi bir satır göreceksiniz. Sistem açılırken en önce `/` dosya sistemini mount eder. Bu sayede bu dosya sistemi altında yeralan ve hayati önem taşıyan yazılımlara (`fsck`, `mount` gibi) ulaşır. Daha sonra yerel dosya sistemleri ve en son da ağ üzerinden erişilen dosya sistemlerine (NFS) bağlanır. fstab dosyasında kullanıcının kendi dosyalarını yerleştiremediği, fakat sistem tarafından kullanılan *swap* ve *proc* dosya sistemleri için de mount bilgileri bulunur.

Yukarıdaki fstab dosyasında ext2 dosya sistemine sahip `/dev/hda2` sabit disk bölümü, / dizinini oluşturuyor. MS-DOS formatlı `/dev/hda1` bölümü de `/dos` dizinine erişilebilir olarak açılış anında eklenecektir.

Sistemdeki CD-ROM sürücü `/cdrom` dizini altına yerleştirilmiş olup root kullanıcı bunu istediği dizine koyabilir. Bu dosyaya, mount komutundaki gibi parametreler de eklenebilir. Dördüncü sırada yeralan defaults parametresinin yanına eklemek istediğiniz opsiyonları yazın. Bu parametrelerden sıkça kullanılan bir tanesi de "user" olup root dışındaki kullanıcılara mount hakkını verir. CD-ROM sürücüyü normal sistem kullanıcılarının mount veya umount yapabilmesi için fstab dosyasındaki ilgili satırı

```
/dev/cdrom      /cdrom      iso9660             defaults,user 1 1
```

satırıyla değiştirin. *defaults* parametresi ise , halihazırdaki dosya sistemini okunup yazılabildiğini, asenkron olduğunu, üzerindeki bilgileri bloklar halinde alıp verdiğini (buna diğer örnekler CD-ROM ve disket sürücülerdir), içerdiği programların çalıştırılabilir olduğunu ve normal kullanıcıların `mount`, `umount` yapamadığını gösterir. *user* parametresiyle kullanıcılara `mount` ve `umount` hakkı yukarıda verilmiştir.

4.3.3 Dosya Sistemi Yaratılması ve Kontrolü

Bazen MS-DOS altından oluşturulamayan dosya sistemlerini (ext2 gibi) oluşturmak gerekebilir. Linux altında bir dosya sistemini oluştururken aynı zamanda bunu kontrol eden programlar vardır. Bunların en çok kullanılanı, Slackware sürümünde gelen `mkfs/`'tir.

`mkfs` ile Linux native (ext2) dosya sistemi oluşturmak için `mkfs.ext2` programı kullanılır.

```
mkfs.ext2 <aygit> <blok-sayisi>
```

"aygıt" yerine üzerine dosya sistemi kurulacak olan cihazın `/dev` dizini altındaki düğüm dosyasının ismi yazılır. Blok sayısı yerine, `fdisk` yazılımından o bölümün kaç blok olduğunu öğrenip bu sayıyı girmelisiniz. Bir blok 1024 bayttan oluşur. Örneğin

```
# mke2fs /dev/hda3 163829
```

komutu, `/dev/hda3` disk bölümünde ext2 dosya sistemi kurar. Bu komutu kullanmadan önce iyi düşünün, zira burada yeralan tüm dosyalar silinir ve bu hatanın geri dönüşü olmaz. Aşağıdaki komut ile bir disket üzerinde ext2 dosya sistemi kuruluyor.

```
# mke2fs /dev/fd0 1440
```

`mke2fs` ile `mkfs.ext2` komutları aynıdır. Benzer şekilde, MS-DOS dosya sistemi yaratmak için `mkfs.msdos`, minix dosya sistemi yaratmak için `mkfs.minix` kullanılabilir. Blok sayısı girilmez ise `mke2fs` bunu otomatik olarak bulacaktır.

Herhangi bir sebepten dolayı diskte bozulma ve veri kaybını en aza indirmek için de yazılımlar vardır. `fsck` ile dosya sistemi kontrolü yapılabilir. `fsck` ile ext2, `dosfsck` (veya `fsck.msdos`) ile MS-DOS formatlı alanlar kontrol edilir.

```
linux:~# fsck /dev/hda2
Parallelizing fsck version 0.5b (14-Feb-95)
e2fsck 0.5b, 14-Feb-95 for EXT2 FS 0.5a, 95/03/19
/dev/hda2 is mounted.  Do you really want to continue (y/n)? yes

Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
Fix summary information<y>? yes

Block bitmap differences: -46486 -46487 -46488 -46489 -46490.  FIXED
Free blocks count wrong for group 5 (2803, counted=2808).  FIXED
Free blocks count wrong (16326, counted=16331).  FIXED
```

```
/dev/hda2: ***** FILE SYSTEM WAS MODIFIED *****
/dev/hda2: ***** REBOOT LINUX *****
/dev/hda2: 3494/29632 files, 42904/59235 blocks
```

Bir dosya sistemini kontrol etmeden önce okunan bölümü root dosya sisteminden ayırın ve kontrol işleminden sonra bilgisayarı kapatıp tekrar açın.

`debugfs` programı, dosya sistemi parametrelerini incelemek için kullanılır. Disk üzerine doğrudan erişım yaptığı için dikkatli kullanmalısınız. ext2fs yardımıyla kurtarılamayan bazı dosyalar (özellikle silinmiş dosyalar) `debugfs` yardımıyla kurtarılabilir.

4.4 Sembolik Bağlantılar

Bazı durumlarda, bir dosyayı oluşturup bu dosyanın başka bir dosyayı işaret etmesi istenebilir. Genellikle sistem yöneticileri tarafından kullanılan sembolik bağlantı yardımıyla bir dosya veya dizin, bunlara karşılık gelen başka bir dosya veya dizin olarak gösterilir.

Aşağıda bir sembolik bağlantı örneği yeralıyor. Sistem görevlisi, `/root` dizini altındayken `/usr/src/linux` dizinine geçmek istiyor. `/usr/src/linux` dizinine işaret eden bir bağlantı kurmak için `ln` komutu kullanılır. -s parametresi, bu bağlantının sembolik olacağına işaret eder.

```
ln -s <nereye-baglanti-yapilacak> <hangi-isimle-yapilacak>
```

```
linux:~# ls -l
total 116
drwxr-xr-x   2 root     root         1024 Feb 14 08:13 kernel
-rw-r--r--   1 root     root       115695 Sep 15  1994 lodlin15.zip
drwx------   2 root     root         1024 Jan  7  1980 mail
-rw-r--r--   1 root     root            0 Feb 15 08:14 typescript
linux:~# ln -s /usr/src/linux linux
linux:~# ls -l
total 116
drwxr-xr-x   2 root     root         1024 Feb 14 08:13 kernel
-rw-r--r--   1 root     root       115695 Sep 15  1994 lodlin15.zip
drwx------   2 root     root         1024 Jan  7  1980 mail
lrwxrwxrwx   1 root     root            9 Feb 15 08:14 linux -> /usr/src/linux
-rw-r--r--   1 root     root            0 Feb 15 08:14 typescript
```

Bu işlemden sonra kullanıcının `/usr/src/linux` dizinine geçmesi için `cd linux`" yazması yetecektir. Sembolik bağlantı dosyasını diğerlerinden ayıran, satırın en solundaki `l` karakteridir.

4.5 Dosya Arşivleme ve Sıkıştırma

Linux altında dosya arşivlenmesi ve sıkıştırılması neredeyse belirli ve standartlaşmış birkaç komut ile gerçekleştiriliyor. `tar` komutu yardımıyla istenilen dosyaları arşivleyebilir, bunların üzerine yenilerini ekleyebilir, istediğiniz zaman da tekrar açıp eski haline getirebilirsiniz. GNU `gzip` komutu ise dosyaların sıkıştırılması (zip) ve sıkıştırılmış dosyaların açılması (unzip) işlemlerini yapabiliyor. Fazla disk alanınız yoksa bu iki komut her zaman yardımınıza koşacaktır.

4.5.1 Dosya Arşivleme

`tar` (Tape ARchive) programı, bir veya birden fazla dosyayı tek bir forma sokar. Genellikle bir dizin ve bu dizinin altında yer alan tüm alt dizinleri biraraya getirmek için kullanılır. `tar`, dosyaların üzerinde sıkıştırma işlemini normalde uygulamaz. Fakat `tar`'ın GNU sürümü (Linux'ta da kullanılan) bu işlemi gzip kullanmadan da yapabiliyor.

`tar` komutuna bir örnek verelim :

```
$ tar -cf arsiv.tar tmp/
```

Bu komut, `-c` (create archive) parametresi yardımıyla `tmp` dizini altındaki herşeyi paketleyip `arsiv.tar` isimli dosyaya yollar. `-f` parametresi, hangi dosyaya yazılacağını belirtir. İstediğiniz zaman dosyayı tekrar açmak isterseniz :

```
$ tar -xf arsiv.tar
```

komutunu kullanın. `-x` (extract) parametresi, `-f` ile belirtilen dosyayı açacaktır.

`-r` opsiyonu ile daha önce yaratılmış bir arşive dosya eklenebilir. Böylece dosyayı açma ve tekrar arşiv oluşturma işleminden kurtulmuş olursunuz.

```
$ tar -rf arsiv.tar ekleme.txt
```

Dosyanın içeriğini önceden görebilmek için `-t` parametresini kullanın ve bu işlemi de alışkanlık haline getirin. Bazı durumlarda arşivlenmiş yazılım, kendine ait bir dizin açmak yerine içerdiği dosyaları bulunduğu yere yazabilir.

```
tar -tf arsiv.tar
```

Hangi dosyaların açıldığını ekranda görebilmek için `-v` parametresini ekleyin:

```
# tar -zcvf tmp.tgz /tmp
tar: Removing leading / from absolute path names in the archive.
tmp/
tmp/tar-error
```

```
tmp/.X11-unix/
tmp/rc.inet1.OLD
tmp/networks.OLD
tmp/linux/
tmp/linux/fss.txt
tmp/lilo.conf
tmp/PKGTOOL.REMOVED
```

Aşağıdaki komut, arşivlenen dosyaları diskete kaydetmeye yarıyor. Bunun için temiz bir disketi yuvasına yerleştirin ve aşağıdaki satırı yazın.

```
# tar -cf /dev/fd0 tmp/
```

`tmp` dizini altındaki herşeyi diskete kaydetmiş olduk. Açmak için bilinen yöntemi kullanırız :

```
# tar -xf /dev/fd0
```

4.5.2 Dosya Sıkıştırma ve Açma

Dosya sıkıştırma amacıyla sıkça kullanan iki yazılım vardır : `gzip` ve `compress`. Her iki yazılım da *Slackware Linux* dağıtımında bulunuyor. GNU'nun dağıtımı olan `gzip`, `tar` ile birlikte kullanılan ve dosya sıkıştırma ve arşivlemede neredeyse standart haline gelmiş bir program. Bir dosyayı sıkıştırmak için parametre girmeden gzip komutunun ardından dosya ismini yazın.

```
$ gzip elvis
$ ls -al
elvis.gz
```

`gzip` ile sıkıştırılmış dosyaların sonu .gz ile biter. Bunları açmak için `gunzip` komutunu kullanın:

```
$ gunzip elvis.gz
```

`compress` ve `uncompress` de sırayla bir dosyayı sıkıştırmak ve açmak için kullanılır. Aslında Linux'ta `uncompress` adında bir yazılım yoktur, bu dosya ismi `compress` dosyasına bağlantılıdır. `compress` ile sıkıştırılan dosyaların sonu .Z ile biter.

```
$ ls -al web.html
-rw-r--r--   1 gorkem   users       41450 Jan 27 13:40 web.html
$ compress web.html
$ ls -al web.html.Z
-rw-r--r--   1 gorkem   users       18906 Jan 27 13:40 web.html.Z
```

Yukarıdaki `web.html` dosyasını sıkıştırdığımızda dosyanın boyutu 41450 bayttan 18906 bayta indi. Açmak için:

```
$ uncompress web.html.Z
```

Bir dosyanın uzantısından ne tür bir dosya olduğunu anlayamazsanız file komutu imdadınıza koşar. Linux'ta belirli uzantılı dosyaların başı bilinen bir harf veya harf grubu ile başlar. file komutu dosyanın başındaki karakterleri kontrol ederek ve bunları bir listeyle (`/etc/magic`) karşılaştırarak dosyanın ne tür olduğunu söyler.

```
# file tmp.gz
tmp.tgz: gzip compressed data - deflate method , last modified: Sat Feb 15 08:21:50 1997 os: Unix
# file tmp.tar.Z
tmp.tar.Z: compressed data 16 bits
```

4.5.3 Birlikte Kullanım

ftp adreslerinde Linux için yeralan yazılımlar genellikle `tar` ve `gzip`, nadiren de `tar` ve `compress` ile sıkıştırılıp arşivelenerek saklanırlar. Çünkü tar komutu tek başına arşivi sıkıştırmaz, bu işlem için `gzip` veya `compress` kullanılır. Bunların açılabilmesi için `tar` komutu ve birkaç parametre yeterli olur.

```
$ ls
netscape-4.0.linux-elf.tar.gz
```

Yukarıdaki dosya önce tar ile arşivlenmiş, ardından `gzip` ile sıkıştırılmış. Tek adımda bu iki dosyayı açmak için tar dosyasına x ve f parametrelerinin dışında z parametresini de ekleyin :

```
$ tar -zxf netscape-4.0.linux-elf.tar.gz
$ ls
netscape-4.0.linux-elf
```

Eğer dosya sıkıştırılırken compress komutu kullanılmışsa z yerine Z parametresini yazın.

```
$ ls
folder.tar.Z
$ tar -Zxf folder.tar.Z
$ ls
folder
```

Benzer şekile, bir dosyayı aynı anda hem tar ile arşivlemek, hem de sıkıştırmak istersek c ve f parametreleri dışında compress ile Z, gzip ile z parametrelerini girmek yeterlidir. Aşağıda sırayla gn-gopher dizisinin önce tar ve gzip ile, ardından tar ve compress ile arşivlenmesi görülüyor.

```
$ ls
gn-gopher/
$ tar -zcf gopher.tgz gn-gopher/
$ tar -Zcf gopher.tar.Z gn-gopher/
```

4.6 Dizin Tarama

Linux işletim sistemini ilk kurduğunuz anda, yüzlerce dizin altında binlerce dosyanız olacaktır. `find` komutu, bu dosyaların arasında tarama yapabilmek için geliştirilmiştir. Bu komutla sadece dosya isimlerine değil, dosyanın sahibi, erişim hakları, son erişim tarihi gibi verilere de ulaşmak mümkündür. `find` komutunun en sık kullanış şekli şudur :

```
find <dizin-ismi> -name <ne-aratiliyor>
```

dizin-ismi yerine taranacak olan dosyanın bulunduğu dizin ismi yazılır. -name parametresinden sonra ise aranan dosyanın ismini girin. Burada "*" ve "?" gibi joker karakterlerini de kullanabilirsiniz. (Bu gibi özel karakterler hakkında daha geniş bilgi *Bash kabuğu* konusu altında bulunabilir)

Aşağıda, find komutunun kullanımına ilişkin birkaç örnek yeralıyor.

```
$ find . -name "*.html"
(bulundugunuz dizinden itibaren sonu .html ile biten dosyalari arar)
$ find /home/halil -name "screen"
(/home/halil dizininden itibaren screen isimli dosyalari arar)
```

Bu komut, `-type d` parametresi yardımıyla sadece dizin isimleri arasında arama yapar.

```
$ find /usr/ -name "linux" -type d
```

`find` komutuna bir alternatif olan `locate` komutu ile dosyalar çok kısa bir sürede taranabiliyor. `update` komutu, her sabah belirli bir saatte sistemdeki tüm dosya ve dizin isimlerini bir veri tabanına yükler. Daha sonra bu dosyada yeralan belirli bir dosya veya dizin ismini `locate` ile arayabilirsiniz:

```
# locate pico
/usr/bin/pico
/usr/man/man1/pico.1.gz
```

Yukarıdaki komut ile içinde pico kelimesi geçen dosyaları bulduk.

4.7 mtools MSDOS Arabirimi

Linux altından, MS-DOS formatlı disketlere ulaşmayı kolaylaştıran paketin ismi `mtools` olarak biliniyor. Bu isim, paketin tüm komutlarının ilk harfinin "m" olmasından kaynaklanmış. Paketle birlikte gelen çeşitli yardımcı programlar sayesinde MS-DOS formatlı diskette yeralan dosyalar listelenebilir, silinebilir, disket formatlanabilir, hatta disketin kimliği değiştirilebilir.

Son çıkan `mtools` paketi, 2MB formatlı disketleri de tanıyabiliyor. Aşağıda, `mtools` paketinden çıkan bazı komutlar ve örnek kullanımları gösteriliyor.

mdir : disketin içeriğini görüntüler.

```
linux:~# mdir a:
 Volume in drive A is SYSTEM_BT
 Directory for A:/

IO       SYS       40566    9-30-93   6:20a
MSDOS    SYS       38138    9-30-93   6:20a
AUTOEXEC BAT          91    3-29-95  10:23a
NDD              <DIR>      3-29-95  10:26a
CHKLIST  MS           81   11-07-96   7:00p
VGA_45           <DIR>      3-29-95  10:28a
YARDIM           <DIR>      3-29-95  10:17a
UTIL             <DIR>      3-29-95  10:17a
VIRUS            <DIR>      3-29-95  10:17a
       9 File(s)     291328 bytes free
linux:~#
```

mcopy : Bir dosyayı Linux'tan diskete, disketten Linux'a aktarır.

```
linux:~# mcopy a:/autoexec.bat .
Copying AUTOEXEC.BAT
```

mdel : DOS dosyasını siler

mformat : Disketi formatlar.

mmd : Disket üzerinde dizin oluşturur.

MSDOS `mtools` arabirimi aynı zamanda VFAT tipi uzun dosya isimlerini de ekranda gösterebilir. Eğer dosya ismi, tek bir DOS ismi içine sığmıyorsa ve 8+3 kuralını bozuyorsa uygun bir kısa isim yaratılır.

Bunun yanında `mtools`'un 2 Mb'lık disketleri okuyabilme özelliği de vardır. mtools paketinin konfigürasyon dosyası `/etc/mtools.conf`'tur. Bu dosyanın kullanımı için mtools(1) man dosyasından yararlanabilirsiniz.

Chapter 5

Linux Sistem Yönetimi

Bir işletim sistemi, marka ve modeli ne olursa olsun bakım isteyen bir araçtır. Yapılacak küçük bir hata, veya hatayı gösteren bir işaretin gözardı edilmesi, zaten değerli olan zamanın boşa gitmesine neden olur. Linux da açık bir işletim sistemi olduğundan, sorumluluk sahibi sistem yöneticisi tarafından açık tutulduğu süre içinde verimli ve düzgün çalışması için gerekli özen gösterilmelidir. İnternet üzerinde üzerinde 500 kullanıcı olan , veya `<ftp.linux.org>` adresi gibi her hafta onbinlerce ziyaretçisi olup ağır sorumluluklar yüklenen Linux makinaları vardır. Bunların bir saat boyunca bile kapalı kalması, o anda bağlanmayı bekleyen pek çok kullanıcıyı mağdur durumda bırakacaktır.

Performansı etkileyen unsurların başında, önlemek pek de elinizde olmasa da elektrik kesintileri ve makinanın kapalı olduğu zamana denk gelen çekirdek değişimleridir. Bu tür donanımsal etkileri bir tarafa bırakırsak sistem yöneticisinin uyması gereken ve İnternet etikleri dediğimiz davranış kuralları vardır. Bu kurallara öncelikle sistem sorumlusu uymalı ve kullanıcıları da uymaya zorlamalıdır. Açık işletim sistemlerinin en büyük sorunlarından biri olan güvenlik konusuna sistem yöneticisi belirli günler zamanının bir bölümünü ayırmalıdır. Kullanıcılardan gelebilecek her türlü saldırı, sisteme onarılması çok zor olabilecek zararlar verebilir. Bunun için düzenli olarak dosyaları kontrol edin ve şüpheli durumlarda araştırma yapmaktan çekinmeyin. Bu konuda daha fazla bilgiyi *Linux İşletim Sisteminde Güvenlik* başlığı altında bulabilirsiniz.

Yukarıdaki tabloya bakarak işinizin zor olduğu kanısına kapılmayın, zira tüm bu denetleme mekanizmasını basitleştirebilirsiniz. Yapılacak tüm rutin işleri bir programa yazıp günün veya haftanın belirli saatlerinde bu programı çalıştırın ve olabilecek tüm hata mesajlarını bir dosyaya yazın. Bu dosyayı boş bir vakitte inceleyerek programın ne kadar düzgün çalıştığını gözlemleyebilirsiniz. Bunun yanında sistem yönetimini zevkli kılacak uygulama programları da vardır. Bu uygulama yazılımlarından bazıları o kadar gelişmiştir ki yapamadıkları tek iş yemek pişirmektir.

Yabancı kaynaklı yayınlardan *Essential System Management (Frisch)* ve *Essential System Administration (O'Reilly)* bu yönde tavsiye edilebilecek kitaplar. Piyasada bulunan Türkçe kitapların hiçbirisinde sistem yönetimine ciddi olarak değinilmediği için Türkçe yayın bulmak gayet zor.

5.1 Yetkili Kullanıcı ve Diğer Kullanıcılarla İlişkiler

Sistemin en büyük sorumlusu *root* şifresini bilen kullanıcıdır. Yukarıda da belirtildiği gibi bu şifrenin sorumluluğu büyüktür. Sistem yöneticisi, makinadaki kullanıcıların isteklerini ve şikayetlerini yönelttikleri tek merci durumundadır. Burada karşılıklı güven sözkonusudur, bunu da sistem görevlisi herkese hissettirmelidir. root kullanıcısının, başkalarından gözünüz gibi sakladığınız e-posta ve dosyaları sizden habersiz okuduğunu öğrenseydiniz tepkiniz ne olurdu ?

root kullanıcısının sahibinin haberi olmaksızın sistem kullanıcılarına ait dosyaların yerini ve izinlerini değiştirmesi de uygun değildir. Zorda kalsanız bile, başkalarına ait hiçbir dosyayı silmeyin veya içeriğini değiştirmeyin. Belki de sıkışık bir durumdan kurtulmanın başka bir yolu vardır ?

Temel olarak zaman zaman kullanıcıları bilgilendirmek hoş bir davranış olarak karşılanır ve onların Linux'a olan ilgilerini perçinleyebilir. Örneğin `/etc/motd` ve `/etc/issue` dosyalarına periyodik olarak çekirdek değişimlerini ve eklenen uygulama programlarını haber veren birkaç satır iliştirebilirsiniz. Sistemde açılan her kullanıcı hesabına *Linux Çokça Sorulan Sorular (Linux ÇSS)* `<http://www.linux.org.tr/linuxtrfaq.html>`, çeşitli UNIX belgeleri ve İnternet hakkında bilgilendirici dökümanlarının yeraldığı bir dizi e-posta da yollanabilir. Bu konuda yaratıcılığınızı konuşturun.

5.2 Sistem Kuralları

Kullanıcı hesabı açmadan önce birkaç basit kuralı tanımlayıp kullanıcılardan bunlara uymalarını istemelisiniz. En basitinden birkaç kural, bir hesabın birden fazla kişi ile kullanılmaması, sistemin güvenliğini zedeleyecek `crack, cops, satan` gibi programların diğer kullanıcıların zararına çalıştırılmaması ve sistem kaynaklarının sorumsuzca tüketilmemesi olarak sayılabilir.

Siz, en iyisi yukarıda yeralan kuralları birkaç madde halinde yazın ve sistemdeki herkesin bu maddelere uyacağını taahhüt eden imzasını alın. Tatsız durumların çıkmaması için de önlem almayı ihmal etmeyin.

5.3 Kullanıcı İşlemleri

Aşağıda yeralan konularda, kullanıcı hesaplarının açılması ve kapatılması anlatılıyor.

5.3.1 Kullanıcı Hesabı Açmak (adduser)

Kullanıcı hesabı açmak iki yoldan yapılabilir. Birincisi, `adduser` komutu yardımıyla, diğeri de bu komutun yaptığı işlemleri sırasıyla elden yapmak. root olarak sisteme girin ve komutu çalıştırın :

```
# adduser

Login name for new user (8 characters or less) []: mehmet
```

```
User id for mehmet [ defaults to next available]:

Initial group for mehmet [users]:

Additional groups for mehmet []:

mehmet's home directory [/home/mehmet]:

mehmet's shell [/bin/bash]:

mehmet's account expiry date (MM/DD/YY) []:

OK, I'm about to make a new account. Here's what you entered so far:

New login name: mehmet
New UID: [Next available]
Initial group: users
Additional groups: [none]
Home directory: /home/mehmet
Shell: /bin/bash
Expiry date: [no expiration]

This is it... if you want to bail out, hit Control-C.  Otherwise, press
ENTER to go ahead and make the account.


Making new account...

Changing the user information for mehmet
Enter the new value, or press return for the default

        Full Name []: linux kullanicisi
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:

Changing password for mehmet
Enter the new password (minimum of 6, maximum of 8 characters)
Please use a combination of upper and lower case letters and numbers.
New password:
Bad password: too simple.
Warning: weak password (enter it again to use it anyway).
```

```
New password:
Re-enter new password:
Password changed.
Done...
```

Kullanıcının isminin 3 karakterden az, 8 karakterden fazla olmamasına özen gösterin. Sistem sizden kullanıcının gerçek ismini, ev dizinini, (genellikle /home içinde olur), grubunu isteyecektir. `adduser` komutu, eğer özel olarak değiştirmek istemezseniz her kullanıcının grup numarasını GID = Group ID) 100 olarak tanımlayacaktır. Bu gruba karşılık gelen isim (Group Name) *users* dır. Kullanıcı kimliği (UID = User ID) ise genellikle 500'den sonra gelen ve kullanılmayan ilk sayı olarak atanır. Eğer UID olarak 0 girerseniz sistemde sizinle beraber bir süper-kullanıcı daha yaratmış olursunuz :)

Sisteme girişte çalıştırılacak olan kabuk için seçimi ister kullanıcıya bırakın, isterseniz öntanımlı olarak bash verin. Bash'in sistemdeki tam patikasını (`/bin/bash`) yazmalısınız. Bu alana sadece kabuk ismi girmek şart değildir.

Yukarıda sistem bizi güçlü bir şifre seçmediğimiz için uyardı. Şifre olarak, herkesin kolayca tahmin edemeyeceği zorlukta, fakat kullanıcının bir yere yazmasını önleyici kolaylıkta bir kelime bulun. Kullanıcı, hesabının açılmasından hemen sonra sisteme girecekse ve şifresini değiştirecekse kısa ve kolay bir şifre yeterli olacaktır.

Her şeyin doğru olduğuna eminseniz, sorulan soruya "y" ile, değilseniz "n" ile cevap verin. Hemen ardından `/etc/skel` dizini altında yeralan tüm dosyalar bu kullanıcının dizinine kopyalanacaktır. Buraya, her kullanıcının ev dizinine önceden yüklenmesini istediğiniz sistem dosyalarını yerleştirebilirsiniz. Bu andan itibaren yeni kullanıcı sisteme girebilir.

Sisteme kullanıcı ekleme işlemini el yordamıyla yapmak isterseniz, `/etc/passwd` ve `/etc/shadow` dosyalarına birer satır eklemelisiniz. `/etc/passwd`'de aralarında birer ':' karakteri olmak kaydıyla sırayla hesap ismi, "x", grup kimliği, kullanıcı kimliği, kullanıcı isim ve soyadı, ev dizini ve kabuk programı yeralır. `/etc/passwd` içine bu satırı ekleyin:

```
ozgur:x:100:502:Misafir Sanatci:/home/ozgur:/bin/bash
```

`/etc/shadow` içine ise aşağıdaki satırı yerleştirin, tabi diğer satırlara dokunmadan.

```
ozgur::9990:0:99999:7::::
```

Bu satırda yeralan ikinci girdi mirkelam kullanıcısının şifresini belirtiyor. `/etc/shadow` dosyasına yazılacak olan şifreyi önce boş bırakıp, ardından `passwd ozgur` komutuyla bir şifre atayın.

`/etc/shadow` dosyasında kullanılabilecek parametreler için shadow(5) man dosyasına göz gezdirin.

Ardından `/home/ozgur` dizinini oluşturun. Dizinin grubunu da ayarlayın. Çok gerekli olmasa da gelen e-postalar için `/var/spool/mail` dizininde bir dosya tanımlayın ve `/etc/skel` içinde yeralan dosyaları kullanıcının dizinine kopyalayın.

```
# mkdir /home/ozgur
```

```
# chown -R ozgur.users /home/ozgur
# touch /var/spool/mail/ozgur
# chown ozgur.mail /var/spool/mail/ozgur
```

Yukarıda yazılan elle kullanıcı girme işlemi, shadow kullanan Linux makinaları için geçerlidir. Shadow için *Güvenlik* bölümüne göz atın.

5.3.2 Kullanıcı Grubu

Eğer Linux makinanız birkaç yoğun görev üstlenmişse (örneğin www veya ftp sunucusu olarak kullanılıyorsa) her bir işlem için birden fazla grup tanımlanabilir. *users* grubuna bağlı kalmadan *wwwadm* veya *ftpadm* gibi grup isimleri belirtebilir, bu grupları da bir veya birkaç kullanıcıya ait olacak şekilde değiştirebilirsiniz.

users grubunun dışında birden fazla grup tanımlanması, sistemin denetlenmesini bir miktar güçleştirse de sistem güvenliğini artırır. Bazı kullanıcıların birden fazla grubunun olmasıyla bu kullanıcılar rahatlıkla sistemde *root* kullanıcısının üstlenmesi gereken işleri yapabilirler.

Grup kimliğinin değiştirilmesi veya yeni grubun eklenmesi için `/etc/group` dosyası kullanılır. Bu dosyaya grubun ismi, numarası, grup şifresi ve gruba ait kullanıcılar yazılır. Her kullanıcının en az bir grubu vardır.

`/etc/group` dosyasının iki satırında yeralan parametreler,

```
root::0:root
sys::3:root,bin,sys,adm
```

gibidir. Sırayla ":" karakteri ile ayrılmış her kolon,

1. Grup ismi
2. Grup şifresi
3. Grup numarası
4. Gruba ait kullanıcılar

şeklinde özetlenebilir. İlk sütun grup ismini belirtir. Grup ismi, `ls -l` komutunu işletince kullanıcı isminin hemen sağında görülen kelimedir. Grup şifresi, gruba ait olmayan kullanıcıların dosyalarına erişim sağlamak için kullanılır.

Grup şifresi nadiren kullanılır. Grup numarası, kullanıcının ait olduğu grubun karşılığıdır. Son sütunda ise gruba ait kullanıcılar, her biri birer virgülle ayrılmış olarak yazılır. Kullanıcı sisteme girdiği zaman otomatik olarak `/etc/passwd` dosyasında yeralan grup ismi aktif olur, bu yüzden her kullanıcının ismi `/etc/groups` dosyasına yazılmaz. Sistem ilk kurulduğu anda birkaç grup halihazırda zaten vardır. Örneğin root grubunun grup numarası 0'dır ve pekçok sistem dosyasının sahibidir, bin grubunun numarası is 2 olup `/usr/bin` ve `/bin` dizinlerindeki çoğu dosyanın grubudur.

Hangi grubun (veya tanımlanmışsa grupların) içinde olduğunuzu öğrenmek için `groups` komutunu kullanabilirsiniz.

```
$ groups gorkem
users
```

5.3.3 Kullanıcının Hesabının Silinmesi

Bir kullanıcıya ait hesabı silmek için `/etc/passwd` dosyasında bu kullanıcıya ait olan satırdaki şifre kolonunun başına * işareti koyun. Bir şifrelenmiş bir parolanın içinde * karakteri olamayacağından bu kullanıcı bir daha sisteme giremez.

Hesabı tamamen silmek için, kullanıcının ev dizinini ve e-postalarını yedekledikten sonra sistemde kullanıcıya ait olan dosyaları (`/tmp` altında) ve ev dizinini silin.

`/etc/passwd` ve `/etc/group` dosyasından kullanıcının isminin geçtiği satırları kaldırın.

Sistemde kullanıcıya ait olabilecek tüm dosyaları da silmeniz gerekebilir. Bunun için `find` kullanabilirsiniz.

```
$ find / -user gorkem -ls -exec rm {} \;
```

Bu komutu `/etc/passwd` dosyasından kullanıcı ismine ait olan satırı kaldırmadan önce yapmalısınız.

5.4 Sistemin Yedeklenmesi

Makinanızın belirli aralıklarla yedeklenmesi, sistem güvenliğinin bir parçası sayılabilir. Donanımdan meydana gelen bir hata sonucu veya yanlışlıkla girilen bir komut, sistemi açılamaz hale getirebilir. Ancak yedekleme ile bu tür kullanıcı hatalarının önüne geçebilirsiniz. Yedekleme genellikle tüm sistem üzerinde yapılmaz. Sadece önemli sistem bilgilerini içeren dizinler yedekleme aşamasına girmelidir.

Yedekleme üniteleri olarak teyp, disket veya başka diskler kullanılabilir. Teypler, büyük miktarda yedekleme işlemleri için ön plana çıksa da birkaç dosyadan oluşan paketlerin korunması söz konusu olacaksa disketler daha pratiktir. İnternet üzerinde ise bağımsız bir makina, kendi rutin görevi dışında yerel ağda bulunan makinaların yedeklerini belirli gün ve saatlerde almak üzere programlanabilir.

Sistemde fazla miktarda kullanıcı varsa yedekleme işlemi ciddi bir düzene oturtulmalıdır. Sistemdeki hayati dosyalar örneğin her hafta, üzerinde kullanıcıların bulunmadığı akşam saatlerinde yedeklemeye sokulabilir.

Yedekleme için birkaç yol önerilebilir. Kullanışlı olması için yedeklemeyi yapan program, yedeklenen dosyalar arasındaki belirli bir dosyayı kolayca açabilmelidir. Bunun yanında, yedek bilginin de bozulması ihtimali vardır. Yedeğin içindeki tek baytın bozulması, bazı durumlarda tüm yedeğin çöp yığını haline gelmesine yol açabilir. Doğru yedekleme programının kullanılması bu anda önem kazanır.

Yedeklenecek olan dizinlerden en önemlileri, sistem konfigürasyon dosyalarının yeraldığı `/etc`, kullanıcı dizinlerinin yeraldığı /home, bazı X Window dosyalarının bulunduğu /usr/X11/lib/X11 dizini altındaki bazı dosya ve dizinler, `/usr` ve `/usr/local` dizinde yeralan bazı dizin ve dosyalar ile `/root` dizinidir. Şart olmasa da yedekleme için yeriniz bolsa `/tmp` dizinini de koyabilirsiniz.

Elektronik postaları (`/var/spool/mail`) da unutmamak gerekir. Yoksa, çok meşhur olan "e-meyılım cyber-space'de kayboldu" cümlesini sıkça duymaya başlarsınız.

Aşağıdaki komut, `/usr` ve `/home` dizinlerini diskete yedekler.

```
$ tar cvMf /dev/fd0 /usr /home
```

`tar` komutununa verilen M parametresi yardımıyla bir disket dolduğu zaman başka bir diskete devam etmenizi sağlar. Dizinleri sıkıştırarak atmak için z seçeneğini vermek yeterli olur. Sıkıştırarak yedekleme disket ya da teyp'te yer kazandıracaktır, ancak hata olduğunda verilerin kurtarılması daha zor olacaktır. Açmak için

```
$ tar xf /dev/fd0
```

komutu yeterlidir. Sıkıştırılarak diskete atılan yedekler için açarken de "z" kullanın. Disketleri sıra ile numaralandırmayı ihmal etmeyin, yoksa bir disketten fazla yer kaplayan arşivleri açamazdınız.

Linux'ta gerek SCSI, gerek floppy teyp desteği vardır. Özellikle çok veri yedekleyecekseniz teypler kaçınılmaz olur. Teyplere yedek alırken de yukarıdaki floppy orneklerindeki gibi `tar` kullanabilirsiniz. `tar`'ın yanı sıra sıkça kullanılan yedekleme komutları arasında `dump ve restore`, `cpio` ve `dd` gösterilebilir. Linux'ta ilk SCSI teyp sürücüye `/dev/st0`, ikinciye `/dev/st1` ... kullanarak, PC'lerde sıkça kullanılan "floppy teyp"lere de `/dev/ftape` üzerinden erişebilirsiniz. Piyasada bulacağınız ortalama floppy teyplerin kapasiteleri 120M, 4mm SCSI teyplerin 4G, 8mm SCSI teyplerin ise 7G kadar olacaktır. Teybin kapasitesi kullandığınız sürücünün yeteneğine ve teybinizin uzunluğuna göre değişir. Örnek olarak, bütün sistemin tar kullanarak ilk SCSI teybe yedeklemek için,

```
# tar cvf /dev/st0 /
```

Bir TCP/IP ağınız varsa, ve birden fazla Linux ya da UNIX makinası kullanıyorsanız, tek bir teyp sürücü ile bütün makinalarınızın, hatta Windows makinalarınızın bile (Linux'un yardımıyla) yedeğini alabilirsiniz. Bu iş için *amanda* gibi daha özelleşmiş ve yedekleme işlemini otomatikleştiren bir program kullanabileceğiniz gibi, `GNU tar`'dan da yararlanabilirsiniz. Örneğin `tar` ile, *cclub.metu.edu.tr* makinasından, *linux.org.tr* makinasındaki floppy teybe yedek almak için,

```
cclub:~# tar cvf linux.org.tr:/dev/ftape /home
```

yukarıdaki komutun çalışması için linux.org.tr makinasının `root`'unun `.rhosts` dosyasında cclub makinasının `root`'una izin verilmiş olması gerekir. Bu işlem güvenlik için sakıncalı olduğundan, bu iş için özel bir kullanıcı açmak gibi bir yönteme başvurulabilir. Örneğin linux.org.tr'da `yedek` adlı bir kullanıcı açtığımızı düşünelim. Bu durumda yukarıdaki komutu,

```
cclub:~# tar cvf yedek@linux.org.tr:/dev/ftape /home
```

şeklinde yazabiliriz.

5.5 Sistemin Güncellenmesi

Tekrar Linux yüklenecek olan makina üzerinde halihazırda Linux bulunmasının yararları olduğu gibi bazı sakıncaları vardır. İlk olarak yararlarına değinmek gerekirse, yapılacak aslında sadece işletim sisteminin bulunduğu bölümü değiştirmektir. Takas alanının ayrılması, kullanıcı alanının ayrılması gibi işlemler bu durumda önceden yapılmış oluyorlar. Ancak yeni işletim sisteminin yüklenmesinin bazı zararları da var.

Herşeyden önce çalıştığı bilinen bir işletim sistemi yenisi ile değiştiriliyor. Gerçekten bunu gerektirecek bir sebep olmadıkça bu tür güncellemelerden kaçınılması veya en aza indirilmesi gerekir. Yeni bir sistem, yeni bilinmeyenler ve az da olsa bile çalışmama riski getirecektir.

Eğer sistem üzerinde belirli hizmetler veriliyorsa o hizmetlere ilişkin verilerin bu güncelleme esnasında yok olmamasına dikkat edilmeli. Mesela makina bir web sunucu olarak kullanılıyorsa web dosyalarının bir şekilde yeni sisteme aktarılması gerekecektir .

Benzer bir şekilde makinaya ait tanımların (`/etc`) ve standart dağıtım dışında yüklenmiş paketlerin önceden yedeklenmesi gerekir. Yoksa herşeye yeniden başlamak gerekebilir.

Güncelleme için iki yöntem akla gelebilir, doğrudan yeni sistemi eskisinin üzerine bindirmek veya eskisini silerek yeniden yüklemek. Slackware sürümü için en sağlıklısı eski sistemi silmektir. Çünkü değişik dağıtımlarda dosya dağılımı farklılıklar göstere bilmektedir. Bunun sonucunda bazı dosyalar birden fazla yerde gözükebilmektedirler. Bir diğer sakınca ise setup yazılımının bazen bu durumda şaşırabilmesi ve hata vermesi. Bazı ticari Linux sürümleri (*Red-Hat, Debian, Caldera gibi*) eski sürümü silmeye gerek bırakmadan sistemi güncelleme özelliğine sahipler.

Güncellemeniz gerekebilecek dizinler, `/var/spool/mail`, kullanıcı dizinleri (root dahil), `/usr` altında gerekli dizinler,`/var/adm/` ve `/etc` olarak sayılabilir. Bazen `/etc` içinde yer alan dosyaların yeri veya şekli değişmiş olabilir. Onun için en güvenlisi teker teker dosyaları inceleyip kopyalamaktır. Belirli dosyaları yeni sistemi kurduktan sonra hemen kopyalayabilirsiniz :

```
/etc/group
/etc/printcap
/etc/hosts ...
```

Daha önce shadow kullanmayan bir sistemden shadow sistemine geçiyorsanız, `/etc/passwd` ve `/etc/shadow` dosyalarında gerekli değişiklikleri yapmadan yeni yerine kopyalamayın.

Ayrıca ne olur ne olmaz diye çalıştığını bildiğiniz bir çekirdeği de yedeğe almayı ihmal etmeyin.

5.6 Çekirdeğin Derlenmesi

Bu bölümde, çekirdeğin konfigürasyonu ve derlenmesi için gerekli bilgiler vardır. Çekirdek derleme uzun ve zahmetli bir işlemdir. Derleme aşamasını en kısa zamanda ve en uygun yoldan gerçekleştirmek gerekir. Derleme işlemi, kullanıcının önüne kafasını karıştıracak kadar çok seçenek sunar.

Nedir bu çekirdek? Genel anlamda çekirdek, yazılımın ve donanımın arasında bekleyerek iletişimi sağlayan bir arabirimdir. Çalışan tüm programların ve süreçlerin hafıza ile olan ilişkilerini düzenler. İşlemcinin gücünün tüm bu programlar arasında adil bir şekilde dağıtımını üzerine alır. Bu işlemleri kullanıcının haberi olmadan her an durmaksızın yapar.

Çekirdek kaynak dosyaları diskte hatırı sayılır bir alan tutar. Örneğin sürüm 2.0.20 yaklaşık 6Mb disk alanı kaplar. Açıldığı zaman ise toplam kapladığı alan 24 Mb olur. Derleme aşamasından sonra bu rakam 30 Mb'yi geçer. Her eklenen aygıt sürücü desteği bir miktar daha yer alır.

Kullandığınız sisteme göre çekirdeğin derlenme süresi değişkenlik gösterir. 12 Mb 486DX4/100 makinasında, 5 dosya sistemi desteği, ağ desteği ve ses kartı sürücüsü ile yaklaşık 30 dakika sürer. Aynı konfigürasyonu içeren çekirdeğin derlenmesi ise daha güçsüz bir makina olan 8Mb 386DX/40 altında 1.5 saati bulur. Derlerken gidip çayınızı için, biraz televizyon seyredin. Makinaniz 386 ise bir banyo alıp dışarı çıkın, alışveriş yapın. Çok ümitsiz bir durumdaysanız, yapılacak en iyi şey en yakın bilgisayar satıcısına uğrayarak yeni fiyatları sormak :-)

Çekirdek kaynak kodu , *ODTÜ ftp arşivindeki* `<ftp://ftp.metu.edu.tr/pub/linux/tsx/sources/system>` adresinden veya tsx-11.mit.edu adresinin yansısını tutan herhangi bir adresten alınabilir. Bu dizin altındaki Linux sürümlerinin bir tanesini (v1.2, v2.0, v2.1 gibi) seçtikten sonra altındaki dosyayı alabilirsiniz. Dosyanın ismi linux-x.y.z.tar.gz yapısındadır.

Aşağıdaki örnekte ODTÜ ftp arşivinden 2.0.20 sürümünün alınışı gösterilmektedir:

```
$ ftp ftp.metu.edu.tr
Connected to ephesus.cc.metu.edu.tr.
220-
220-Welcome to METU Ftp Site !
220-
220-All logins and transfers are logged with your hostname.
220-If you do not agree with this policy, please cancel now.
220-
220 ephesus FTP server (Version wu-2.4(5) Wed Jan 10 18:06:56 EET 1996) ready.
Name (ftp.metu.edu.tr:gorkem): ftp
331 Guest login ok, send your complete e-mail address as password.
Password:
230-##############################################################################
230-##    Welcome to the Middle East Technical University Ftp Archive       ##
230-##############################################################################
230-
```

```
230-
230-Hello ftp@compclup.ceng.metu.edu.tr !
230-You are the 19. anonymous user.
230-Maximum number of users allowed is 100.
230-Local Time is Tue Nov  5 12:19:41 1996.
230-
230-Mirrored sites are placed under the directory /pub
230-This archive is maintained by ftpload@ftp.metu.edu.tr
230-You can reach this archive via FSP at port 21.
230-
230-You can reach public directories via "cd dirname"
230-command independent of your current location;
230-e.g. "cd linux" instead of "cd /pub/linux"
230-
230 Guest login ok, access restrictions apply.
ftp> cd /pub/linux/tsx/sources/system/
250-Please read the file README
250-  it was last modified on Mon Sep 16 11:36:00 1996 - 50 days ago
250 CWD command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 66
-rw-r--r--  1 ftpload  ftp-adm     36981 Sep 16 12:36 CREDITS
drwxr-xr-x  3 ftpload  ftp-adm       512 Oct 13 15:59 Historic
-rw-r--r--  1 ftpload  ftp-adm     12056 Sep 16 12:36 README
drwxr-xr-x  2 ftpload  ftp-adm       512 Oct 14 11:47 SillySounds
drwxr-xr-x  2 ftpload  ftp-adm       512 Oct 13 18:17 testing
drwxr-xr-x  2 ftpload  ftp-adm       512 Oct 14 11:47 v1.0
drwxr-xr-x  2 ftpload  ftp-adm      2048 Oct 31 21:55 v1.1
drwxr-xr-x  2 ftpload  ftp-adm      1024 Jun 25 11:34 v1.2
drwxr-xr-x  2 ftpload  ftp-adm      6656 Sep 27 04:24 v1.3
drwxr-xr-x  2 ftpload  ftp-adm      1536 Nov  2 00:05 v2.0
drwxr-xr-x  2 ftpload  ftp-adm       512 Nov  2 21:33 v2.1
226 Transfer complete.
ftp> cd v2.0
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> dir linux*
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 141009
-rw-r--r--  1 ftpload  ftp-adm   5859566 Jul  3 11:22 linux-2.0.1.tar.gz
```

```
-rw-r--r--  1 ftpload  ftp-adm   5940281 Jul 27 06:39 linux-2.0.10.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5952403 Aug  5 03:01 linux-2.0.11.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5954278 Aug  9 12:53 linux-2.0.12.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5981780 Aug 16 06:19 linux-2.0.13.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5984960 Aug 20 11:52 linux-2.0.14.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5986368 Aug 25 06:20 linux-2.0.15.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5990228 Aug 31 14:03 linux-2.0.16.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5991606 Sep  2 06:37 linux-2.0.17.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5993351 Sep  5 10:38 linux-2.0.18.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5997342 Sep 11 10:21 linux-2.0.19.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5861789 Jul  5 12:09 linux-2.0.2.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5997693 Sep 13 08:53 linux-2.0.20.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5999162 Sep 20 09:51 linux-2.0.21.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   6009488 Oct  8 13:02 linux-2.0.22.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   6010694 Oct 18 08:20 linux-2.0.23.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5931967 Jul  6 07:53 linux-2.0.3.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5926911 Jul  8 08:48 linux-2.0.4.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5927899 Jul 10 08:23 linux-2.0.5.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5929807 Jul 12 07:22 linux-2.0.6.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5931378 Jul 15 07:38 linux-2.0.7.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5936467 Jul 20 02:52 linux-2.0.8.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5939510 Jul 26 04:09 linux-2.0.9.tar.gz
-rw-r--r--  1 ftpload  ftp-adm   5843677 Jun  9 23:36 linux-2.0.tar.gz
226 Transfer complete.
ftp> get linux-2.0.20.tar.gz
200 PORT command successful.
150 Opening BINARY mode data connection for linux-2.0.20.tar.gz (5997693 bytes).
226 Transfer complete.
5997693 bytes received in 136 secs (43 Kbytes/sec)
ftp> quit
221 Goodbye.
```

Sisteme root olarak giriş yapın ve `/usr/src` dizinine geçin. Linux'un kurulması aşamasında çekirdekleri de kurmuş iseniz bu dizinde 'linux' isminde ve eski çekirdeğin kaynak kodlarını kapsayan bir alt dizin göreceksiniz. Genellikle her kullanıcının yaptığı, bu dizinin ismini o an çalışan çekirdeğin sürümüne göre değiştirmektir. Çekirdek sürümünü bulmak için ise `uname -r` komutunu kullanabilirsiniz.

```
$ uname -r
  2.0.23
```

Bu komuttan çekirdeğin 2.0.23 sürümü olduğunu anlayabiliriz.

Çekirdek kodunu içeren paketi açmadan önce `/usr/src` dizinine kopyalayın, ve açmadan önce bu dizinde 'linux' isimli bir dizinin olmadığına emin olun. Şimdi `/usr/src` içinde iken sıkıştırılmış çekirdek kodunu 'tar' komutu yardımıyla açın.

```
# tar -zxf linux-x.y.z.tar.gz
```

Komutun işletimi bitince 'linux' dizini oluşur. İlk defa derleme işlemi yapacaksanız aşağıdaki komutları sırasıyla yazın.

```
# cd /usr/include
# rm -rf asm linux scsi
# ln -s /usr/src/linux/include/asm-i386 asm
# ln -s /usr/src/linux/include/linux linux
# ln -s /usr/src/linux/include/scsi scsi
# cd /usr/src/linux
# make mrproper
```

5.6.1 Çekirdek Konfigürasyonu

`/usr/src/linux` dizini altinda iken `make config` yazın. Bu komut, bir konfigürasyon dosyası yaratmak amacıyla sorular sorar.

Karşınıza çıkan sorulara 'y' (evet) veya 'n' (hayır) ile cevap vermeniz gerekir. Bazı aygıt sürücülerde 'm' seçeneği de olabilir. Bunun anlamı, sistem o programı *modül* olarak derleyecek ve çekirdeğe dahil etmeyecek demektir. Kullanıcı derleme aşamasından sonra istediği zaman bu modülü çekirdeğe dahil edebilir, işini bitirdikten sonra da çıkarabilir.

2.0.x ve daha yukarı sürümlerde '?' tuşuna basmakla daha ayrıntılı bilgi alabilirsiniz. Aşağıda bazı konfigürasyon parametreleri verilmiştir.

Matematik işlemci emülasyonu (Kernel math emulation)

Kullandığınız makinanın işlemcisi yardımcı matematik işlemcisine sahip değilse (486SX, 386SX veya daha düşük makinalar) bu seçeneğe evet cevabını vermelisiniz.

Normal (MFM/RLL) disk ve IDE disk/cdrom desteği (Normal disk and IDE disk/cdrom support)

Hemen herkesin kullandığı PC sabit disklerinin standart olması sebebiyle bu seçenege 'y' cevabı verilebilir. Daha ileride SCSI sabit disk ve cdrom desteği de sorulacaktır. Ardından gelecek olan 'old disks-only' ve 'new IDE' sürücülerinden birini seçerken, aralarındaki farkın ilkinde tek arabirim üzerinden iki adet diski desteklemesi, diğerinde ise ikinci arabirim ile birlikte IDE/ATAPI cdrom'ları desteklemesidir.

Ağ desteği (Networking support)

Makinanız herhangi bir ağ üzerinde ise (İnternet gibi), veya İnternet'e bağlanmak için SLIP, PPP gibi programları kullanmak istiyorsanız bu bölüme 'y' cevabı vermelisiniz. Fakat, ağ üzerinde olmasanız bile her geçen gün yeni yeni uygulama programlarının ve paketlerin ağ desteğine ihtiyacı olduğunu düşünürsek (X Window gibi) bu bölüme 'y' cevabı vermeniz şart görünüyor.

En fazla 16Mb hafıza (Limit memory to low 16Mb)

Bazı hatalı 386 DMA denetleyicileri hafızanın 16Mb'tan yüksek bölümlerine ulaşamazlar. Çok nadir görülen bu durum makinanızda varsa bu seçeneği 'y' ile geçin.

System V IPC (System V IPC):

IPC (Interprocess Communication) desteği verir. Bazı Perl programları ve oyunlar (DOOM gibi) bu seçenek olmadan çalışamazlar. Bu durumda ne yaptığınızı iyi bilmiyorsanız bu seçeneğe 'y' -evet- cevabı verin.

İşlemci tipi (386, 486, Pentium, PPro):

Burada, çekirdeğinizi hangi tür makina için derliyorsanız, o makinanın tipini giriniz.

SCSI desteği (SCSI Support):

SCSI aygıtlarınız var ise bu seçenek için 'evet' deyin. Daha sonra da karşınıza gelecek SCSI aygıtlar arasından size uygun olanları alın.

Ağ kartı desteği (Network Device Support):

Ağa bağlanmak için bir kart kullanacaksanız veya SLIP, PPP bağlantıları, ya da paralel port üzerinden yapacağınız her türlü işlemler için bu seçeneği işaretleyin. Ardından ekrana gelen kart tiplerinden uygun olanlarını seçin.

Dosya sistemleri (Filesystems):

Burada, aşağıdaki dosya sistemlerinden kullanmak istediklerinizi seçebilirsiniz.

- minix: Yeni dağıtımlar , artik tarih olmaya başlayan bu dosya sistemini desteklememekte. Bazı kurtarma disketleri yaratmak için daha yaygın, fakat daha az kararlı olan ext2 dosya sistemine tercih edilirler.
- Extended file system (ext): Bu da artık kullanımı olmayan bir dosya sistemi. Desteğini ekleyip eklememek size kalmış.
- Second extended file system (ext2): Linux'un kullandığı en önemli dosya sistemi. Alınması şart.
- xiafs: Artık kullanılmıyor.
- msdos: Sabit diskinizde DOS bölümü varsa, ya da DOS disketi kullanmak istiyorsanız, alın.
- umsdos: MS-DOS dosya sistemine bazı yenilikler (uzun dosya isimleri gibi) getirilerek oluşturulan bir teknoloji. MS-DOS desteği almadıysanız buna da gerek yok.
- proc: Bu aslında gerçek bir dosya sistemi değil, çekirdek görüntüsüdür. 'y' cevabını vermezseniz ileride başınız çok ağrır.
- NFS: Linux makinaniz bir ağ üzerinde kurulu iken ağ üzerindeki diğer makinaların dosya sistemlerini kullanabilmenizi sağlar.
- ISO9660: CD-ROM sürücünüzü Linux altında kullanabilmek için 'y' cevabı verin.
- OS/2 HPFS: OS/2 dosya sistemlerini okuyabilmek için 'y' cevabı verin.

- System V: Bir UN*X sürümü olan System V makinalarının dosya sistemleri üzerinde çalışabilmeyi sağlar.

Karakter aygıtlar (Character devices):

Bu bölümde yazıcı, fare ve bazı teyp yedekleme sürücüleri gibi karakter aygıtları (iletişimini karakter karakter sağlayan aygıtlar) için sürücüleri ekleyebilirsiniz.

Ses kartı (Sound Card Support):

Linux'un ses kapasitesini "duyabilmek" için bu desteği ekleyin.

Çekirdek işlemleri (Kernel Hacking):

Çekirdekten doğan hataların bulunmasını kolaylaştıran bu seçenek, programcılara yönelik.

Tüm bu işlemlerin ardından `make config` komutunun görevi bitmiş olur.

5.6.2 Temizlik

Fazla dosyaları temizlemek, gerekli olan dosyaların da yerinde olup olmadığını kontrol etmek için , sırasıyla

```
# make dep
# make clean
```

yazın. Ardından tüm nesne dosyaları ve eski sürümden arta kalan dosyalar silinecektir. Bu adımı es geçmemelisiniz.

Temizlik işleminden sonra,

```
# make zImage                      , veya
# make zdisk
```

yazın. Eğer bir kişisel bilgisayarda çalışıyorsanız, ilk seçenekte çekirdek derlenip `/usr/src/linux/arch/i386/boot` dizini altına `zImage` adıyla yerleştirilecektir. Buraya yerleştirilmeden önce yerden kazanmak için çekirdek sıkıştırılır. Diğer durumda ise yeni çekirdek sabit disk yerine `A:` disket sürücüsüne yazılır. Disket üzerine yazmanın bir avantajı, kararlı olmayan ve sistemi açmayan çekirdek ile karşılaşılması halinde, sistemi sabit disk üzerindeki eski çekirdekten açabilme imkanı bırakmasıdır.

5.6.3 Çekirdeğin Kurulması

LILO, bilgisayarın açılışı anında çekirdek ile ilgili düzenlemeleri okur, buna göre hangi çekirdekten açması gerektiğini anlar. Konfigürasyon dosyası biraz karmaşık olmasına rağmen kullanım aşamasında büyük yararlar sağlar. `/etc/lilo.conf` dosyasının içinde gerekli olan kurulum bilgileri vardır. Bu dosyanın bir bölüm aşağıda görülüyor.

```
image = /vmlinuz
  label = linux
  root = /dev/hda1

other = /dev/hda1
  label = dos
  table = /dev/hda
```

`image =` ibaresi kurulu olan çekirdeğe işaret eder. Çoğu kullanıcı bu kısma `vmlinuz` yazar. `label =` bölümünde hangi işletim sistemi veya çekirdek sürümünün açılışta kullanılacağı yazılır. Hemen altındaki `root` bölümünde ise `label =` kısmında seçilen işletim sisteminin hiyerarşik sırada en üst dizin yapısı yazılır. Eski çekirdeğin bir yedeğini alın ve yeni derlediğiniz çekirdeği `image =` satırında belirtilen yere kopyalayın.

```
# cp zImage /vmlinuz
```

Şimdi LILO programını çalıştırın. Komut satırında `lilo` yazmak yeterlidir. '*' karakteri, çalıştırılacak ilk çekirdeği veya sistemi gösterir.

```
# lilo
Added linux *
Added dos
```

LILO ve konfigürasyonu hakkında , daha fazla bilgi *LILO-NASIL çevirisinden* `<http://www.linux.org.tr/tr-ldp/lilo/>` alınabilir.

Yeni bir çekirdeği derledikten sonra LILO üzerinden ekleme yapmak ve açılış anında kullanmak için `/etc/lilo.conf` dosyasındaki

```
image = /vmlinuz
  label = Linux
  root = /dev/hda1
```

satırlarının kopyasını çıkarın, `image` , `label` ve `root` yerine sırasıyla yeni derlenen çekirdeğin sabit disk üzerindeki yerini, ismini ve Linux işletim sisteminizin kök dizinini yazın. Bunlara ek olarak, açılış anında bekleme sağlamak ve seçim yapabilmek için konfigürasyon dosyasının başına `delay = xxx` ekleyebilirsiniz. *xxx* sayısı, gecikme zamanı saniyenin onda biri olarak verilir. Bu sayede açılış anında , örneğin *shift* tuşuna basılarak istenen çekirdeğin ismi (`label =` satırında yazılan isim) seçilir ve çalıştırılır.

5.6.4 Çekirdeğin Yamanması (patch)

Çekirdeğin her yeni sürümüyle birlikte bir önceki sürümünun yaması da gelir. Yamayı uygulamak, tüm çekirdeğin tekrar derlenmesinden daha kolaydır. Örnek olarak, elinizde 2.0.10 çekirdeği var ve ftp adresinde 2.0.11 sürümü bekliyor. Halihazırdaki çekirdeği 2.0.11 sürümüne yükseltelim:

Yamayı önce `/usr/src` dizinine kopyalayın. Bu dizin altında iken

```
# zcat patch-2.0.11.gz | patch -p0
```

veya eğer sıkıştırılmış değilse

```
# patch -p0 < patch-2.0.11
```

yazın. Bu komutun ardından satırlar ekrandan kaymaya başlayacak ve yama işlemi tamamlanacaktır. Satırlar bazen o kadar hızlı kayar ki neler yazdığını bile göremeyebilirsiniz. Yamanın başarılı bir şekilde tamamlandığından emin olmak amacıyla, `/usr/src/linux` dizini altında `.rej` uzantılı dosyalara bakın.

```
# find . -name '*.rej' -print
```

Komut, `.rej` uzantılı dosyalar bulursa yama işlemi tamamlanamamış demektir. Eğer yama işlemi tamamlanırsa, sırasıyla

```
# make clean
# make config
# make dep
```

komutlarını yazın. Bu komutlarla ilgili detaylı bilgi önceki konuda anlatılmıştır.

`patch` komutunun alabildiği bazı seçenekler vardır. `patch -s` komutu oluşabilecek tüm hata mesajlarını ekrana yazar. Diğer yama seçenekleri `man patch` yazarak görülebilir.

Birkaç yamanın ardından, .orig uzantılı dosyalar etrafta dolaşmaya başlar. Bu dosyalar gereksizdir, büyük yer kapladıkları için silinmeleri gerekir. Aşağıdaki komut yardımıyla `/usr/src/linux` dizini altında `.orig` uzantılı dosyaları silebilirsiniz.

```
# find /usr/src/linux  -name '*.orig' -exec rm -f {} ';'
```

5.6.5 Çıkabilecek Sorunlar

Eger rutin bir çekirdek güncellemesinden sonra yeni çekirdeğiniz gerçekten garip şeyler yapıyorsa, büyük ihtimalle yeni çekirdeğinizi derlemeden önce "make clean" komutunu kullanmayı unutmuşsunuzdur.

Eğer çekirdek çok fazla hafıza harcıyorsa, çok büyükse, veya yeni 786DX6/440 makinanızda bile derlenmesi sonsuza kadar sürüyorsa, büyük ihtimalle ayarlanmış bir çok gereksiz parça (aygıt sürücüleri, dosya sistemleri, vb.) vardır. Büyük çekirdeklerin en belirgin özelliği sürekli hafızadan diske ve diskten hafızaya takas yapmalarıdır.

Çekirdeğinizin ne kadar hafıza kullandığını makinanın toplam hafızasından `/proc/meminfo` dosyasındaki `total mem` değerini çıkararak veya `free` komutunun çıktısından öğrenebilirsiniz. Ayrıca `dmesg` komutunu kullanarak (veya çekirdek kayıt dosyasına bakarak) da bu değere ulaşabilirsiniz.

```
Memory: 15124k/16384k available (552k kernel code, 384k reserved, 324k data)
```

LILO'yu çalışıtırmayı unuttunuz, veya sistem hiç açılmıyorsa bu durumda yapacağınız en iyi şey disketten açılış yapmak ve daha sonra açılış yapabilen başka bir disket hazırlamaktır.

Eğer eski çekirdek 1.2.X ve daha düşükse ve bunu 2.0.x sürümüne yükseltmek istiyorsanız, 2.0.x çekirdek kurulumunda getirilen bazı değişiklikler sorun çıkartabilir. `/usr/src/linux/Documentation/Changes` dosyasında yapmanız gereken değişklik listesi vardır. `gcc`, `libc` gibi birkaç paketi yeniden kurmanız gerekebilir.

Aynı anda birkaç çekirdek kaynak kodunu sistemde bulundurmayı düşünüyorsanız, izlenecek en güvenilir yol, her birine `/usr/src/linux-x.y.z` ismini vermektir (*x.y.z* halihazırdaki çekirdek sürümünü gösterir). Ardından sembolik bir bağlantı yardımıyla istediğiniz kaynak dizinini seçebilirsiniz, örneğin

```
# ln -sf linux-2.0.2 /usr/src/linux
```

komutu, `linux-2.0.2` dizinini aktif hale getirir. Artık kaynak dosyalarının bulunduğu yer `linux-2.0.2` dizinidir.

5.7 Modüller (Yüklenebilir Çekirdek Parçaları)

Linux işletim sistemi çekirdeğinin boyu genellikle 400-600Kb arasındadır. Hantal çekirdeklerde bu sayı 800Kb'lara kadar çıkabilir. Çekirdeğin boyunun büyümesi, sistemin performansını olumsuz yönde etkiler. Özellikle bilgisiz kullanıcıların gerekli gereksiz tüm desteği çekirdeğe yüklemesi sıkça görülür. Bunun önüne geçmek de deneyim ister.

Linux'a özgü özelliklerden birisi modüllerdir (yüklenebilir çekirdek parçaları). Modüller sayesinde nadiren gerekli duyulan bir çekirdek parçası sadece gerektiği zaman çekirdeğe eklenir. Örneğin Java programlarını her zaman çalıştırmak istemiyorsanız, bu desteği modül olarak derleyip bir kenara koyarsınız. Ne zaman bir Java programı çalıştırılacaksa bu modül hafızaya yüklenecektir.

Bir modül, çekirdeğin derleme aşamasında 'M' parametresiyle tanımlanır. Her desteğin modülü henüz geliştirilmemiş olmasına rağmen bu sayı gün geçtikçe artmaktadır. İlginizi çekebilecek modüller arasında disket desteği, yazıcı desteği, minix dosya sistemi desteği veya SCSI aygıt sürücü desteği sayılabilir.

Bir modülü derledikten sonra modüle ait dosya `/lib/modules` dizini altında yeralır. Burada istenilen bir modül

```
# insmod modul-ismi
```

komutuyla çekirdeğe eklenir. Modülü hafızadan çıkarmak için

```
# rmmod modul-ismi
```

yazın. Hafızadaki modüllerin bir listesini görebilmek için de lsmod komutu kullanılabilir.

`kerneld` komutu, arkaplanda çalışarak gerekli olan modülleri anında yükler. Bu sayede çekirdeğe eklemek için `insmod` komutuna gerek kalmaz. Ayrıca belirli bir süre içinde kullanılmayan modülleri de çekirdekten atar.

5.8 Yeni Yazılımların Yüklenmesi

Linux işletim sistemi, benzerlerine nazaran çok büyük bir hızla genişliyor. Her an yeni bir program, yeni bir yazılım piyasaya çıkıyor ve bunları takip etmek de gittikçe zorlaşıyor. Bir ftp adresini kontrol ettikten iki gün sonra tekrar bakıyorsunuz ki üç-beş yeni yazılım gelivermiş.

Tabi ki her an her programı makinanıza indirip açmak zorunda değilsiniz. Buna ne zamanınız, ne de sabit diskiniz yeter (Linux'un sunsite arşivi 20GB'tan fazla yer kaplıyor!). Öyleyse tespih çeker gibi program indirmek veya varolan yazılımları güncellemek komik olur. Sadece gereken zamanda gereken dosyaları makinanıza kurmalı veya güncellemelisiniz.

Bu işlem çekirdek için de geçerlidir. Bazı çekirdek fanatikleri çekirdeği çıktığı anda makinalarına indirir ve derlerler. Linux'a çok önemli bir fonksiyon kazandırmayacaksa her an çekirdek derlemek hem zaman kaybı olur, hem de derleme esnasında sistemi yavaşlatarak diğer kullanıcıları kızdırır.

Kurmak istediğiniz ve ftp adresinden bulup getirilen dosya çok büyük ihtimalle `tar` ve `gzip` ile arşivlenip sıkıştırılmıştır. Bu dosyanın sonu genellikle `tar.gz`, `tgz` veya `tar.Z` ile biter. Dosya kaynak koduyla veya çalıştırılabilir program olarak dağıtılıyor olabilir . Her iki durumda da yapılacak işlemler birbirine benzer.

Öncelikle yazılımın açıldığı zaman hangi dizine kendini yerleştireceğine bakmak gerekir. Bunu tar komutunun `-t` seçeneği ile kontrol edin. compress ve gzip ile sıkıştırılmış dosyalar için,

```
# tar -ztvf dosya.tar.gz
```

komutu kullanılabilir. Çok nadir durumlarda dosya, kendine ait bir dizin yaratmaz, onlarca programı ve konfigürasyon dosyasını bulunduğunuz dizine yazar. Karmaşıklık yaratan bu durumdan ancak arşivi ayrı bir dizin altında açmakla kurtulabilirsiniz.

Linux'çular yeni bir yazılımı denemek istediklerinde arşivi genellikle `/usr/src` veya `/usr/local/src` dizini altına kopyalarlar ve burada açarak biraz sonra anlatılacak konfigürasyon işlemini uygularlar. Derleme gerektirmeyen bazı programların ise doğrudan kök dizinde açılmaları gerekebilir. Bu programlar açıldıkları anda çalışmaya hazırdırlar.

`/usr/src` veya `/usr/local/src` dizini altına dosyayı açtıktan sonra bol miktarda dosya ve dizinle karşılaşabilirsiniz. Buna en iyi örnek Linux çekirdeğidir. Burada yeralan bazı anahtar dosyalar yazılımın kurulması için hayati önem arzeder.

Öncelikle `README`, `README.linux` ve varsa `INSTALL` dosyalarının gözden geçirilmesi gerekir. Bunlar sırayla bir kullanıcı olarak dosyaları nasıl kuracağınızı ayrıntılarıyla söyler. Kurma aşamasında dikkat edilecek dosyalardan ilki `Makefile` veya `makefile` dosyasıdır. Bu dosyanın içinde (nadiren) küçük birkaç değişiklik yapmak gerekebilir. Diğer kontol edilmesi gereken dosya ise varsa `config.h` dosyasıdır. Bunu da sistem gereksinimize göre değiştirmelisiniz.

Bazen başta derleyici olmak üzere sistemdeki yazılımı baştan aşağı kontrol eden programlar olabilir (`configure` gibi). Bunların da komut satırından çalıştırılmaları gerekir.

Varsa `Makefile` dosyasına göre yazılımı kurmak için `make` komutunu kullanın. make komutu kuruluma `make all`, `make install` gibi bazı parametreler alabilir. `make all`, halihazırda yeralan tüm yazılımları kurarken `make install`, bu yazılımı konfigürasyon dosyalarının gösterdiği dizinlere yerleştirirler.

Çoğu büyük paketler, man dosyalarıyla birlikte gelirler. Bunları da diğer man dosyalarının yanına (`/usr/man`) koymak için ayrı bir `make install.man` türü komut kullanmak gerekebilir.

Her durumda, mutlaka `README` dosyasını okuyun. Biraz sıkıcı gelse de sizi zaman kaybından kurtaracaktır.

Slackware dağıtımını kullanıyorsanız, yüklenmiş yazılımları silmek için 2 yolunuz var. Birincisi, root olarak sisteme girip, istemediğiniz programları tek tek komut satırından silebilirsiniz. Ama bu size önerebileceğimiz bir yol değil ve çok fazla sakıncası var. Örneğin, programı silip, ona olan bağlantıları unutabilirsiniz. Ya da tam tersi olabilir. İkinci yol ve en geçerlisi, Slackware dağıtımı ile gelen *pkgtool* programı kullanmanız. `pkgtool` seçenekleri arasındaki *remove package* işinize yarayacaktır.

5.9 Sistem Açılış Dosyaları

Linux makinası açılırken onlarca dosyayı okur. Birkaç dakika içinde çekirdeği hafızaya yükler, belirli dizinlerin altındaki programları arkaplanda çalıştırır. Sistem uyanırken hatırı sayılır nicelikte konfigürasyon dosyası elden geçer. Açık bir işletim sistemini diğerlerinden ayıran bir özellik de kullanıcının büyük bir kolaylıkla bu konfigürasyon dosyalarını değiştirebilmesidir. Sistem açılış dosyalarında , özellikle ağ temelli dosyalarda değişiklik yapabilmek için TCP/IP konusunda basit temelleri anlayabilmiş olmanız gerekir.

Açılış dosyalarında, tek bir kullanıcı için veya sistemde hesabı olan herkese yönelik olarak değişiklik yapmak mümkündür. Pek çok kullanıcı bu açılış dosyalarıyla oynayarak Linux'u daha yakından öğrenme imkanına sahip olur.

İnternet servisi veren bir makina bu servisini daemon'lar aracılığıyla yapar. Bir daemon, her makina tarafından bilinen bir port numarasını açar ve dinlemeye koyulur. Eğer sorumlu olduğu port adresine istek gelirse bu isteğe cevap vererek servise başlar.

Bash kabuğunun anlatıldığı konu içinde sisteme girerken okunan dosyaları anlatmıştım. Sistemde yeralan önemli dizinler de dosya sistemi yapısı konusunda detaylı olarak belirtilmişti. Bu bölümde daemonlardan kısaca bahsedip açılış anında çalıştırılan programlara ve bunların konfigürasyon dosyalarına kısa bir giriş yapılacaktır.

5.9.1 inetd ve /etc/inetd.conf

inetd, en ağır görevleri üstlenen ve en önemli İnternet servislerindendir. Bu program sistem açılırken arkaplanda çalışmaya bırakılır. inetd'nin görevi belirli portları dinleyerek bu portlara yapılan bağlantıları denetlemektir. Bir bağlantı yapıldığı anda inetd bu portla ilgilenecek olan programı çalıştırır. Örnek olarak Linux'a ftp isteği geldiği anda inetd, ftp isteğine cevap verecek olan *wu.ftpd* programını uyandırır ve geri kalanını bu programa bırakır.

inetd olmasaydı her standart port numarası için bu portu dinleyecek olan bir program arkaplanda çalışıyor olacaktı. Bu da sisteme normalden daha fazla yükün binmesi demektir. inetd'nin kullandığı konfigürasyon dosyası `/etc/inetd.conf`'tur. Bu dosyadan bir bölüm aşağıda yeralmaktadır :

```
# "kill -HUP <pid of inetd>".
# The inetd will re-read this file whenever it gets that signal.
#
# <service_name> <sock_type> <proto> <flags> <user> <server_path> <args>
#
time    stream  tcp     nowait  root    internal
time    dgram   udp     wait    root    internal
ftp     stream  tcp     nowait  root    /usr/sbin/tcpd  wu.ftpd
telnet  stream  tcp     nowait  root    /usr/sbin/tcpd  in.telnetd
#nntp   stream  tcp     nowait  root    /usr/sbin/tcpd  in.nntpd
shell   stream  tcp     nowait  root    /usr/sbin/tcpd  in.rshd -L
login   stream  tcp     nowait  root    /usr/sbin/tcpd  in.rlogind
ntalk   dgram   udp     wait    root    /usr/sbin/tcpd  in.talkd
pop3    stream  tcp     nowait  root    /usr/sbin/tcpd  in.pop3d
imap2   stream  tcp     nowait  root    /usr/sbin/tcpd  imapd
finger  stream  tcp     nowait  nobody  /usr/sbin/tcpd  in.fingerd -w
systat  stream  tcp     nowait  nobody  /usr/sbin/tcpd  /bin/ps -auwwx
netstat stream  tcp     nowait  root    /usr/sbin/tcpd  /bin/netstat    -a
# End of inetd.conf.
```

`/etc` altında bir konfigürasyon dosyasını değiştirmek isterseniz, o dosyayı okuyan ve arkaplanda çalışıyor durumda bulunan süreçleri bundan haberdar etmelisiniz. Çünkü bu dizindeki bazı dosyalar sadece açılışta okunur. inetd, yukarıdaki `inetd.conf` dosyasını sistem açılırken okuyacaktır. Değişiklik yaptıktan sonra inetd'nin bu değişiklikten tekrar haberinin olması için bu sürecin `PID` numarasını `ps -aux` komutu ile bir kenara not edin ve şunları yazın.

```
# kill -HUP <PID-inetd>
```

PID-inetd yerine not ettiğiniz süreç numarasını girmelisiniz. Bu sayede inetd kendisine ait olan inetd.conf dosyasını tekrar okuyacak ve değişiklikler işleme konacaktır.

Her bilgisayar tarafından bilinen servislerin isimleri ve hangi portları (portları) kullandıkları `/etc/services` dosyasında tutulur.

5.9.2 syslogd ve /etc/syslog.conf

syslogd programı Linux'taki çeşitli olayların kayıtlarını tutar. Bunlar genellikle hata mesajları veya çekirdek mesajları gibi kayıtlardır. syslogd sistem açılırken arkaplanda işlemeye bırakılır.

`/etc/syslog.conf`, syslogd programının konfigürasyonu için kullanılır. Normal olarak sistem kayıtları `/usr/adm/messages` dosyasının sonuna eklenir. Bu dosyanın yapısı biraz karışık olmasına rağmen eğer sistem yöneticiyseniz syslogd sizin en büyük yardımcınız olabilir.

Örnek bir `syslog.conf` dosyası aşağıda yeralıyor.

```
*.info;*.notice                                    /usr/adm/messages
*.debug                                            /usr/adm/debug
*.warn                                             /usr/adm/syslog
*.emerg;*.alert;*.crit                             /dev/tty8
mail.*                                             /dev/tty9
kern.*                                             /dev/tty10
```

Bu dosyanın en solunda yeralan ve nokta ile ayrılmış olan iki kelimeden ilki, kaydı yapılan program grubudur. Bu kısma kern (çekirdek mesajları), mail (sendmail mesajları), login (sisteme giriş veya sistemden çıkış mesajları), auth, authpriv veya security (güvenlik mesajları), cron (cron mesajları), daemon (deamon'lardan gelen mesajlar),lpr (yazıcı mesajları), news (haber grubu mesajları), user (kullanıcı mesajları) veya uucp (uucp mesajları) yazılabilir. Noktanın sağındaki kelime ise kaydı yapılan program grubunun mesaj önem sırasını belirler. Bu kısımda önem sırasına göre debug, info, notice, warning, err, crit, alert ve emerg bulunur.

Dosyanın sağında yeralan kısımda ise bu mesajların hangi dosyaya yazılacağı vardır. Yukarıda verilen örnekte, syslogd'den gelen mesajlar sanal konsollardan kullanılmayanlara yönlendirilmiştir. Böylece sistem yöneticisi F8, F9 ve F10 tuşlarını kullanarak sistem kayıtlarına anında ulaşabilir.

Başka bir makinaya kayıt tutturmak da mümkündür. Uzak makinanın ismi `syslog.conf` dosyasının sağ bölümüne yazılır. Böylece diğer makina yerel olarak sistem kayıtlarını tutacaktır.

```
mail.err                     @ordek.cclub.metu.edu.tr
```

Bazı kritik ve sistemin düzgün çalışmadığını haber verebilecek mesajların konsola yönlendirilmesinde fayda vardır.

```
kern.crit                    /dev/console
```

Sisteme bağlanmış olan kullanıcılar hakkında bilgiyi ise

```
/var/adm/wtmp
```

dosyası tutar. Bu dosya ASCII olmadığından doğrudan okunamaz, `last` komutu ile içeriği incelenebilir. Buna ek olarak çeşitli paketler çalıştırılıyorsa onların yarattığı kayıt dosyaları da incelenmelidir. Örneğin ftp çalışıyorsa `/var/adm/xferlog` kimin ftp ile hangi dosyaları aktardığını gösterir. Bu tür dosyalar içerisinde elde edilebilecek ipuçları ile sistemin çalışma performansı artırılabilir, kendisini pek belli etmeyen hatalar bulunabilir.

5.9.3 init ve /etc/inittab

init, sistemde ilk çalışan programdır. Neredeyse sistemin tüm açılış yükünü elinde tutar. init sayesinde açılış anında hangi programların çalışacağı belirlenir. Bu programları da /etc/inittab dosyasından öğrenir.

Burada /etc/inittab dosyasından alınan birkaç örnek satır yeralıyor. Her satırın sonunda çalıştırılacak olan dosyalar var.

```
# Default runlevel.
id:3:initdefault:

# System initialization (runs when system boots).
si:S:sysinit:/etc/rc.d/rc.S

# Script to run when going single user (runlevel 1).
su:1S:wait:/etc/rc.d/rc.K

# Script to run when going multi user.
rc:23456:wait:/etc/rc.d/rc.M

# What to do at the "Three Finger Salute".
ca::ctrlaltdel:/sbin/shutdown -t5 -rfn now

# Runlevel 0 halts the system.
l0:0:wait:/etc/rc.d/rc.0

# Runlevel 6 reboots the system.
l6:6:wait:/etc/rc.d/rc.6

# What to do when power fails (shutdown to single user).
pf::powerfail:/sbin/shutdown -f +5 "THE POWER IS FAILING"

# If power is back before shutdown, cancel the running shutdown.
pg:0123456:powerokwait:/sbin/shutdown -c "THE POWER IS BACK"

# If power comes back in single user mode, return to multi user mode.
ps:S:powerokwait:/sbin/init 5

c1:1235:respawn:/sbin/agetty 38400 tty1 linux
c2:1235:respawn:/sbin/agetty 38400 tty2 linux
c3:1235:respawn:/sbin/agetty 38400 tty3 linux
c4:1235:respawn:/sbin/agetty 38400 tty4 linux
c5:1235:respawn:/sbin/agetty 38400 tty5 linux
c6:12345:respawn:/sbin/agetty 38400 tty6 linux
```

```
# Serial lines
#s1:12345:respawn:/sbin/agetty 19200 ttyS0 vt100
#s2:12345:respawn:/sbin/agetty 19200 ttyS1 vt100

# Dialup lines
#d1:12345:respawn:/sbin/agetty -mt60 38400,19200,9600,2400,1200 ttyS0 vt100
#d2:12345:respawn:/sbin/agetty -mt60 38400,19200,9600,2400,1200 ttyS1 vt100

x1:4:wait:/etc/rc.d/rc.4
```

Bir Linux makinasının birkaç tane çalışma düzeni vardır. Bunlardan birkaçı 3 (çok kullanıcılı konum), 1 (tek kullanıcılı konum) ve 4 (X Window konumu)'tür. Her konum, bir sayı veya bu sayılara karşılık gelen tek harften oluşur. Bir konum altında çalışırken diğerine geçebilmek için `init` veya `telinit` komutu kullanılır.

`/etc/inittab`'da yeralan dosya isimleri değiştirilebilir veya kullanıcının isteği doğrultusunda her çalışma modu için farklı dosyalar çalıştırılabilir.

```
# init 1
Switching to runlevel 1
Sending all processes the TERM signal
Waiting for processes to terminate........
Turning off quota..
Single user mode.
```

`init` komutundan sonra açmak istediğiniz çalışma düzenine ait olan sayıyı yazabilirsiniz. Örneğin init 0 komutu sistemi kapatacaktır. Yukarıdaki komut çalıştıktan sonra `/etc/inittab` dosyasında bu konuma ait olan `/etc/rc.d/rc.K` dosyası çalıştırılır. Bu dosya çalıştıktan sonra dışarıdan yapılan bağlantılara izin verilmez.

`/etc/inittab` dosyasında yapılacak her değişikliğin ardından

```
# init q
```

komutunu vererek init programının inittab dosyasını tekrar okuyup değişikliklerden haberdar olmmasını sağlamalısınız. O an sistemin hangi konumda çalıştığını bulabilmek için runlevel komutunu kullanın :

```
$ runlevel
N 3
```

Sistem, yukarıdaki bilgilere göre 3 numaralı konumda (çok kullanıcılı konum) çalışıyor.

5.9.4 /etc/rc.d/* Dosyaları

İşte sistemin kalbinin attığı yer. Ne yaparsanız yapın ama bu dosyaları gözünüz gibi koruyun. Sistem açılırken arkaplanda çalışmaya başlayan programlar burada yazılıdır. Tüm bu dosyalar Slackware dağıtımına özgüdür.

rc.0

Bu dosya, 0 (halt) çalışma düzenine geçerken okunur. Sistemin shutdown veya halt komutları ile kapatılmak istenirse bu program çalıştırılır. Önce sistemdeki tüm süreçler öldürülür, ardından sistemin kapandığını belirten bazı satırları kayıt dosyalarına atar. Tüm dosya sistemlerini umount işleminden geçirdikten sonra da sistemi dondurur.

rc.K

Bu dosya, init tarafından 1 numaralı çalışma düzenine (tek kullanıcılı konum) geçilirken çalıştırılır. Tüm süreçler öldürülür ve sistem sadece konsoldan giriş yapılacak şekilde tekrar açılır.

rc.cdrom

Bu dosya, bulduğu CDROM sürücüsünü /cdrom dizini altına mount yardımıyla ekler.

rc.local

Çok kullanıcılı açılış anında çalıştırılan bu dosyaya istediğiniz programları ekleyebilirsiniz.

rc.4

4 numaralı çalışma düzenine geçilirken bu dosya çağırılır. Tüm sanal konsollara vt100 terminalleri yerine grafik tabanlı bir program kullanılarak girilir.

rc.M

init tarafından çok kullanıcılı konuma geçmek için kullanılır.

rc.S

Sistem açılırken bu dosya mutlaka çalıştırılır. Takas alanı açılır, dosya sisteminde hata olması halinde bunu gidermeye çalışır. Bazı kontrollerden sonra kontrolü diğer rc programlarına bırakır.

5.10 Çekirdek Mesajları

Çekirdek, hafızaya yüklenip çalıştırıldıktan sonra kendisi ve sistemdeki donanım hakkındaki her türlü bilgiyi ekrana yazar. Her türlü bilginin içine hata mesajları da girer. Çekirdeğin hata verip vermemesi açılış anında ekrana bakmakla anlaşılabilir. Bu nedenle sistem açılırken gözünüzü monitörden ayırmamalısınız. Aşağıda bir çekirdeğin açılış kaydı yeralıyor :

```
Console: colour EGA+ 80x25, 1 virtual console (max 63)
bios32_init : BIOS32 Service Directory structure at 0x000faf10
bios32_init : BIOS32 Service Directory entry at 0xfb390
pcibios_init : PCI BIOS revision 2.10 entry at 0xfb3c0
Calibrating delay loop.. ok - 30.22 BogoMips
Serial driver version 4.11 with no serial options enabled
tty00 at 0x03f8 (irq = 4) is a 16550A
tty01 at 0x02f8 (irq = 3) is a 16550A
lp1 at 0x0378, using polling driver
ftape: allocated 3 buffers aligned at: 00210000
hda: QUANTUM ELS170A, 162MB w/32KB Cache, CHS=1011/15/22, MaxMult=8
ide0: primary interface on irq 14
Floppy drive(s): fd0 is 1.44M
FDC 0 is a post-1991 82077
lance.c: PCI bios is present, checking for devices...
Memory: 6612k/8192k available (760k kernel code, 384k reserved, 436k data)
This processor honours the WP bit even when in supervisor  mode. Good.
Swansea University Computer Society NET3.019
Swansea University Computer Society TCP/IP for NET3.019
IP Protocols: ICMP, UDP, TCP
PPP: version 0.2.7 (4 channels) NEW_TTY_DRIVERS OPTIMIZE_FLAGS
TCP compression code copyright 1989 Regents of the University of California
PPP line discipline registered.
SLIP: version 0.8.3-NET3.019-NEWTTY (4 channels) (6 bit encapsulation enabled)
CSLIP: code copyright 1989 Regents of the University of California
eth0: D-Link DE-600 pocket adapter: at I/O 0x378.
Checking 386/387 coupling... Ok, fpu using exception 16 error reporting.
Checking 'hlt' instruction... Ok.
Linux version 1.2.13 (root@bigkitty) (gcc version 2.7.0) #1 Wed Aug 23 01:06:43 CDT 1995
Partition check:
  hda: multiple mode turned off
  hda: hda1 hda2
VFS: Mounted root (ext2 filesystem) readonly.
Adding Swap: 8188k swap-space
```

Yukarıdaki mesajda swap alanı belirlenmesi, sabit diskin tanınması, disketin tanınması, aygıt sürücülerinin, ethernet kartını tanıması ve ekranın özelliklerinin yazılması gibi bilgiler vardır. Bu bilgilere sistem açıldıktan sonra `/var/adm/messages` dosyasından da erişilebilir.

5.11 Linux Dosya Sistemi Yapısı

Bu bölümde Linux'un kullandığı dosya sistemi hakkında bilgi verilecektir. Dosya sistemini oluşturan dizinler, kapsadıkları dosyalar ve bunların Linux işletim sistemindeki görevleri de kısaca belirtilecektir.

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 yeralan dosya, diğer sürümde `/bin` altına yerleştirilebiyor.

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 makinalar da sunucu üzerindeki diski paylaşır. Buna örnek olarak kullanıcı ev dizinlerinin yeraldığı `/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 yeralmalıdır.

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 konfigürasyon dosyaları. Bu dizinde çalıştırılabilir dosyalar bulunmamalıdır.

- skel : Buradaki dosyalar, kullanıcı hesabı açıldığında kullanıcının ev dizinine kopyalanır.
- rc.d : Bu dizinin içinde, init sürecinin başvurduğu konfigürasyon dosyaları vardır. Bunlara "rc dosysları" 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 konfigürasyon 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 kullanılabilecek kabuk isimleri burada tutulur.
- login.access : login komutu için konfigürasyon dosyası. Sisteme girişi kullanıcı bazında sınırlamak için kullanılir.

/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 yeralıyorlardı. 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 Window 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 yeralması gerekmeyen çalıştırılabilir sistem görevlisi dosyaları
- /var : Sürekli değişen sistem bilgileri burada tutulur. İstisnalar dışında diğer makinalarla paylaştırılmaz.
- adm: Sistem yönetimini ilgilendiren kayıtlar
- preserve : Sistemin göçmesinden sonra zarar görmesi mümkün dosyaların kaydedildiği yer.
- spool : Sonra işlenecek olan veriler buraya atılır (e-posta gibi)

Eğer sistem yöneticisi iseniz düzenli olarak dosya ve dizin yapısını kontrol etmelisiniz. Örneğinn `/tmp` dizini normal kullanıcıların kullanımına açık olduğu için çok kısa bir sürede dolabilir. Burayı periyodik olarak silmek için find komutu işe yarayabilir :

```
# find /tmp -atime +7 -exec rm -rf  {} \;
```

Yukarıdaki komut, `tmp` dizini altında yeralan tüm dosyaları kontrol edecek ve 7 günden daha eski dosyaları silecektir. Benzer şekilde, kullanıcıların dosyalarının yer aldığı `/home` dizini de çok kısa bir sürede dolmaya mahkumdur. Sistemde ne kadar büyük sabit disk olursa olsun bir gün gelecek burası dolacaktır. Periyodik olarak buraya da göz gezdirmek gerekir. Aşağıda yeralan kısa program, `/home` dizininde yeralan kullanıcı dizinlerideki disk kullanımını büyükten küçüğe doğru `disk` isimli dosyaya yollayacaktır. Bu dosyanın incelenmesiyle hangi kullanıcıları "cezalandıracağınızı" görebilirsiniz :)

```
# du -skx /home/* | sort -rn > disk
```

`du` (disk usage), sistem kullanıcılarının sıkça başvurduğu bir komuttur. Kullanılabilecek parametreler için *du(1)* man sayfasına bakın.

UNIX, (ve dolayısıyla Linux) iki tip aygıt tanır. Bunlardan ilki karakter aygıtları, başka bir deyişle veri alışverişini bayt bazında yapan cihazlar, diğeri de blok aygıtları, yani veri alışverişini blok cinsinden yapan cihazlardır. Karakter aygıtlarına örnek olarak seri bağlanan cihazlar ve teypler, disk aygıtlarına örnek olarak da diskler verilebilir. Bu cihazlardan bir bilgi yazılıp okunduğunda dosya sistemi altında bunları tanımlayan dosyalar (`/dev/*`) kullanılmış olur. Örnek olarak bir yazıcıya bilgi göndermek için,

```
$ cat dosya > /dev/lp1
```

komutunu kullanabilirsiniz. Dosya, yazıcının anlayabileceği bir yapıda olmalıdır (örneğin düz metin).

Her aygıt, bir dosya olarak dosya sisteminde kendine yer bulduğuna göre hangi aygıt dosyalarının bulunduğunu görmek mümkün olabilir.

```
$ ls -l /dev/cua0
crw-rw-rw- 1 root uucp 5, 64 Nov 30 1993 /dev/cua0
```

İlk kolondaki ilk karakter (c), bu dosyanın bir karakter aygıtına ait olduğunu belirtiyor. Normal dosyalar için buradaki karakter '-' olmalıydı. Benzer şekilde blok aygıtlar için ise 'b' kullanılr.

Aygıt sürücü desteği çekirdekte olmasa bile `/dev` dizini altında genellikle tüm aygıt dosyaları bulunur. Bu demektir ki `/dev/sda` dosyasına sahip olduğunuz halde bir SCSI sabit diski sistemde takılı olmayabilir. Tüm aygıt sürücülerinin bulunması, programların kurulumunu ve yeni donanımın eklenmesini kolaylaştırır.

5.12 Konsolda Türkçe Yazmak

Linux sanal terminallerinde bir tuşa basıldığı zaman bilgisayarın ekranda herhangi bir dildeki alfabeye ait harfi göstermesi kolayca sağlanabilir. Bunun yanında 12 fonksiyon tuşunun herbirine belirli bir dizisel değer atanabilir ve bu sayede bu tuşlara basıldığında bir komutun veya programın çalışması mümkün olur.

Aslında bunları yapabilmek için önümüzdeki iki temel sorunu aşmamız gerekiyor. Birincisi standart olmayan karakterlerin (örneğin Türkçe) ekranda görüntülenebilmesi, ikincisi de bilgisayara standart dışı bir klavye olduğunun belirtilmesidir.

Dünya üzerindeki belli başlı dillerin hepsi yıllar önce bir standart geliştirilmesi amacıyla sınıflandırıldı. Avrupa'da Slav dilleri hariç hemen tüm dillerin karakterleri ISO (Uluslararası Standartlar Organizasyonu) tarafından tek bir sınıfa yerleştirildi ve buna ISO8859-1 (latin 1) dendi. O vakit izlenen yanlış politikalar nedeniyle Türkçe bu sınıfa dahil edilmedi ve sadece Türkçe için ISO8859-9 (latin 5) isminde ayrı bir sınıf açıldı.

Türkçe yazabilme sorununu hemen her işletim sisteminde hissediyor olmalısınız. Linux altında Türkçe desteğini sağlamak için birkaç işlem yapılmalıdır. Bu işlemlerin başında Linux'a Türkçe destekli klavye ve font tanıtmak geliyor.

`/usr/lib/kbd/keytables` dizini altında yeralan dosyalar, çeşitli ülke dillerine ait klavye dağılımını içeriyor. Buradaki bilgileri kullanarak klavye üzerinde yeralan tuşlara basılması halinde yazılı tuş dışında bir karakterin ekranda gösterimi sağlanabilir.

Sistem açıldığı zaman -eğer kurulum aşamasında bir font beğenip sürekli onu kullanmıyorsanız- `defkeymap.map` dosyasında yeralan klavye dağılımını kullanacaktır. Farklı bir klavye düzeni kullanmak için .map dosyalarını yüzeysel olarak anlamanız gerekiyor. `/usr/lib/kbd/keytables` dizini altındaki herhangi bir dosyada 300'den fazla satır ve üç ana tanımlamayla karşılaşacaksınız. Birinci grup, "keycode", "alt", "control" gibi bir satırla, ikinci grup "string" ile, üçüncü grup ise "compose" kelimesiyle başlar.

Dosyanın yapısını daha iyi kavrayabilmek için üzerinde bir örnekle açıklamaya çalışalım. Control-Alt-Del tuşlarına basılınca bilgisayar kendini "reset"ler. Eğer yanlışlıkla bu üç tuşa basmışsanız, klavyeden girilecek bir `shutdown -c` komutunu yazmak ve sistemi kapanmaktan kurtarmak için sadece 3-4 saniyeniz vardır. Fakat `shutdown -c` komutunu bir fonksiyon tuşuna bağlar ve bu tuşa basarsanız sistem eski haline dönecektir. Örnek olarak Control-F1 tuşunu bu işe ayıralım.

Öncelikle `/usr/lib/kbd/keytables` dosyasında yeralan `defkeymap.map` dosyasındaki F1 fonksiyon tuşunun hangi tuş tanımına denk geldiğini bulmalıyız. Dosyadaki F1 ile ilgili satırlar şunlardır :

```
keycode  59 = F1               F13              Console_13
        control keycode  59 = F25
        shift   control keycode  59 = F37
        alt     keycode  59 = Console_1
        control alt     keycode  59 = Console_1
```

Linux, her satıra bir numara atar. Yukarıdaki satırda, F1 tuşunun numarasının 59 olduğu kolayca görülüyor. 59 numaralı tuşa basılınca, F1 ile tanımlanan dizi ekrana yazılacaktır. Benzer şekilde F1'e control ile birlikte basılması halinde ise F25 dizisini ekrana basar. Önceden tanımlanan değerlerden *Console_1*, 1 numaralı sanal konsolun ekranda gösterilmesini işaret eder. Tüm bu dizilerin içeriğini görebiliriz. Aynı dosyanın sonlarında şu satır yeralıyor :

```
string F1 = "\033[[A"
```

Bunun anlamı, F1'e basınca `ESC[[A` dizisini ekrana gönder demektir. sayısı, Escape tuşunun oktal (sekizli) sistemde gösterimidir.

Bu satırı veya F1'den itibaren seçeceğimiz bir fonksiyon tuşu ismini istediğimiz şekilde değiştirebiliriz. Yanlız dikkat etmemiz gereken bir husus, bu tuşun başka uygulamalar tarafından kullanılabiliyor olmasıdır. Örnek olarak F61 ve karşılık gelen dizi değeri dosyada yeralmıyor. Dosyaya küçük bir ekleme yapalım:

```
string F61 = "shutdown -c\n"
```

Dosyayı farklı bir isimde (örneğin `bas.map`) kaydedip çıkın. Yeni klavye dağılımını denemek için

```
# loadkeys bas.map
```

komutunu kullanın. Artık yeni klavye standardı sisteme yerleşmiştir. Sistemin çalıştığına emin olmak için Shift-Control-F1 tuşuna basın. `shutdown -c` komutu işletilmelidir.

Artık Türkçe yazabilmeye geldik. Bunun için standart dışı klavyeden gelen bir Türkçe karakteri ekranda gösterebilmek amacıyla bir font dosyası oluşturmamız gerekecek. Linux altında bu dosyalar `/usr/lib/kbd/consolefonts` dizini altında yeralıyor.

Linux'ta standart dışı bir font yüklemek için `setfont` komutu kullanılır. Türkçe fontları, *Linux Kullanıcıları Grubu ftp arşivinden* `<ftp://ftp.linux.org.tr>` temin edebilir veya iso09 ön adlı herhangi bir fontu kullanabilirsiniz. Aşağıdaki komut, 8 piksel eni, 16 piksel boyu olan Türkçe font yükleyecektir.

```
setfont tr8x16.map
```

Türkce klavye düzenlerini içeren dosyaları ftp ile yine aynı adresten alabilirsiniz. Burada `tr.map` ve `trq.map` adlı iki dosya bulacaksınız. Bunlardan `tr.map`, normal klavye düzeninin yanında 'alt' tuşları ile "c, g, i, s, o, u" tuşlarına basıldığında karşılık gelen özel Türkçe karakterler görüntülenir. `trq.map` yaygın olarak kullanılmakta olan Q-Türkçe klavye düzenini içerir.

Bash kabuğu altında Türkçe yazabilmek için aşağıdaki üç komutu `.inputrc` dosyanıza eklemeniz gerekir.

```
set meta-flag on
set output-meta on
set convert-meta off
```

`less` komutununu Türkçe karakterleri göstermesi için de aşağıdaki gibi LESSCHARSET değişkenini latin5 yapmak gerekiyor.

```
LESSCHARSET=latin5
```

`loadkeys` ve `setfont` komutlarını sistem açılırken aktif olmaları için bunları `/etc/rc.d/rc.local` dosyasına yerleştirebilirsiniz. Fontu ve klavye düzenini ayarladıktan sonra sistemde çıkın ve tekrar girin.

Artık Türkçe yazabilirsiniz.

5.13 Belirli Zamanlarda Komut İşletilmesi

crontab dosyası yardımıyla sistem üzerinde olmadığınız zamanlarda Linux'tan bazı belirli işleri yapmasını istemek mümkündür. Örneğin gereksiz trafik sıkışıklığı yaratamak için akşam geç saatlerde sistemi yedeklemek için evden İnternet'e bağlanmak yerine crontab ile bu işlemi otomatikleştirmek mümkün olur.

Bir crontab dosyasında toplam 6 alan bulunur. Bunlardan ilk beşi komutun ne zaman işleneceğini gösterirken son kısımda bu saatler içinde çalıştırılacak olan komutu gösterir. Zamanı belirten sütunlardan ilki dakikayı (0 ile 59 arası), ikincisi saati (0 ile 23 arası), üçüncüsü günü (1-31 arası), dördüncüsü ayı (1-12 arası), son kısım ise haftanın gününü (0 pazarı göstermek üzere 0-6 arası) verir. Yine crontab'ın güzel özelliklerinden birisi de bu alanların her birisinde zaman aralıklarını yazmaya izin vermesidir. Mesela haftanın üçüncü ve beşinci günleri arası, saat 4 ve 9 arası gibi. Eğer bir zaman alanına * girilirse, bunun anlamı "her zaman" olur. Örneğin saat yerine * konulması "her saat", ay yerine * konulması "her ay" anlamına gelecektir.

Önce aşağıda yeralan küçük bir crontab dosyası incelenecek, ardından crontab işlemine nasıl başlanılacağı kısaca anlatılacaktır.

```
40 06 * * * updatedb
```

Yukarıdaki örnekte birinci ve ikinci sayılar 06:40 saatini gösteriyor. Diğer zaman kısımları * ile doldurulmuş olduklarından her gün 06:40'ta `updatedb` isimli program çalışacaktır.

Komut yerine kullanılacak olan program ismi parametre alabilir. Bu örnekteki crontab dosyası pazartesi günleri saat 1'de, `/tmp` dizinindeki üç günden daha eski dosyaları siler.

```
0 1 * * mon find /tmp -atime 3 -exec rm -f {} \;
```

Bu örnekte kullanılan virgül, her ayın 9, 19 ve 29'unu gösterir. Aynı alanda birden fazla sayı kullanacaksanız arada boşluk bırakmamaya dikkat edin.

```
0 1 9,19,29 * * find /tmp -atime 3 -exec rm -rf {} \;
```

Ay isimlerinde *ocak, şubat, mart, nisan, mayıs, haziran, temmuz, ağustos, eylül, ekim, kasım, aralık* için sırasıyla *january, february, march, april, may, june, july, august, september, october, november, december* kullanabilirsiniz. Benzer şekilde *pazar, pazartesi, salı, çarşamba, perşembe, cuma, cumartesi* ve *pazar yerine sunday, monday, tuesday, wednesday, thursday, friday ve saturday* kullanın. Aşağıdaki crontab girdisi yardımıyla iki ayda bir çalıştırılan komutla önce teyp başa alınıyor, ardından `/home` dizini bu teybe yedekleniyor :

```
0 2 1 */2 * mt -f /dev/rft0 rewind; tar -cf /dev/rft0 /home
```

*/2 ile belirtilen zaman dilimini ikiye bölebilirsiniz. 2 yerine uygun herhangi bir tamsayı yazılabilir.

Bu girdileri bir crontab dosyasına nasıl yazacağız? Bunun için root kullanıcısı olarak girin ve aşağıdaki satırları yazın :

```
# crontab -e
```

Ekrana bilgisayar ilk kurulduğu anda ön tanımlı olarak girilen crontab satırları gelecektir. Buraya yukarıda belirtilen yapıda istediğiniz kadar crontab girdisi yapabilirsiniz. Eğer daha önceden *VISUAL* çevresel değişkenini uygun bir editöre ayarlamamışsanız, vi altında çalışmanız gerekecektir. Başka bir editörle çalışabilmek için bu değişkeni, editörün patikasını gösterecek şekilde değiştirin. İleride de kısaca bahsedilecek olan tin (haber grubu okuyucu yazılımı) de VISUAL değişkenine göre metin editörünün patikasını ayarlayacaktır.

```
# export VISUAL=/usr/bin/pico
```

Crontab dosyasının normal olarak işletilmesinden crond sorumludur. Sistem açılırken `/etc/rc.d` dizini içindeki dosyalardan bir tanesi crond programını çalıştırır ve bu andan itibaren `crontab -e` komutuyla yarattığınız dosyadaki emirler zamanı geldikçe yerine getirilir.

Chapter 6

Bash Kabuğu

UNIX için geliştirilen en önemli ve en çok kullanılan kabukların başında *tcsh*, *bash* (Bourne Again Shell) ve *ksh* (Korn Shell) gelir. Bash, Bourne Shell'in biraz daha geliştirilmiş sürümüdür. Tcsh ise C-shell üzerine eklenen yeniliklerle genişletilmiştir. Unix altında, bash kabuğu diğer benzerlerine nazaran daha fazla özelliğe ve işleve sahiptir. Linux işletim sistemi için de derlenen bash ise hem tcsh, hem de ksh'dan özellikler taşır.

Bir Slackware dağıtımında tüm bu kabuklar kullanıcının hizmetine sunulmuş olup bash kabuğu sistem ilk açıldığında tanımlıdır. Kullanıcı sisteme girdiği zaman bu kabuk aktif olur, bundan sonra kullanıcı herhangi bir kabuk altında çalışmalarına devam edebilir. Bunun için sadece ismini komut satırında yazması yeterlidir.

6.1 Bash Özellikleri

Bash'in kullanıcıya zaman kazandıran en önemli özelliklerinden birisi de dosya isimlerini tamamlamasıdır. Komut satırında tamamlanmamış bir komut veya dosya ismi yazdıktan sonra TAB tuşuna basılırsa satır tamamlanacaktır. Eğer komut satırındaki karakter kümesiyle başlayan birden fazla komut varsa bir sinyal sesi duyulacak ve sizden yeteri kadar karakteri yazmanızı bekleyecektir.

```
$ ls
postgres        mandel.doc      lilo-howto
$ vi post <TAB>
$ vi postgres
```

Komut satırındaki karakterler dosyayı veya komutu tanıtmaya yetmiyorsa, <TAB> tuşunun iki kez ard arda basılmasıyla ekrana mümkün olan tüm dosya isimleri getirilebilir.

```
$ ta <TAB> <TAB>
tac   tail  talk  tar
```

```
$ talk
```

Bash, komut satırında iken satırın kolayca değiştirilebilmesini sağlar. Böylece komut çalıştırılmadan önce birkaç tuş darbesiyle üzerinde değişiklik yapılabilir. Klavye üzerindeki alt ve üst yön tuşları, daha önce yazdığınız komutları görmenizi ve arasında seçim yapmanızı sağlar. Sağ ve sol yön tuşları ile seçtiğiniz komutun üzerinde değişiklikleri uygulabilirsiniz.

6.2 Takma Adlar (alias)

`alias` komutu ile bir komut veya komut kümesinin yerine bir isim tahsis edilebilir. İşleyişi bir makroya benzeyen bu komut yardımıyla uzun komutlar, daha kısa komutlarla tanımlanabilir. Bir `alias` komutu , anahtar kelimeyle başlar, ardından bir eşittir (=) işareti ve yerine kullanılacağı komut yazılır. Arada boşluk bırakılmaz.

```
$ alias dir='ls -al'
$ dir
total 668
-rw-r--r--   1 gorkem   users        1016 Dec  7 13:51 .profile
-rw-r--r--   1 gorkem   users         277 Nov 26 13:02 .signature
drwxr-xr-x   4 gorkem   users        1024 Dec  3 18:24 .tin/
drwxr-xr-x   2 gorkem   users        1024 Nov 12 10:58 .wine/
-rw-r--r--   1 gorkem   users         231 Nov 23 17:40 .xinitrc
drwxr-xr-x   2 gorkem   users        1024 Oct  3 16:17 Mail/
-rw-r--r--   1 gorkem   users        26721 Dec  3 14:55 NIS-HOWTO
drwxr-xr-x   2 gorkem   users        1024 Nov 17 14:09 News/
```

Daha karmaşık alias 'lar da tanımlanabilir:

```
$ alias yedek="cd /var/log; tar -zcvf yedek.tgz cron debug lastlog; cd -"
```

6.3 Özel Kabuk Tanımları

Bazı özellikleriyle bash, benzeri kabuklar arasından kolayca sıyrılır. Bunlardan birisi de tanımlardır. Tanımlamalar set komutu yardımıyla yapılır, tüm tanımlar küçük harflerle yazılırlar, bu özellikleri yardımıyla değişkenlerden ayrılırlar.

```
$ set -o tanim          # tanimi isletir
$ set +o tanim          # tanimi kaldirir
```

Aşağıda en çok kullanılan iki tanım yeralmaktadır:

noclobber : Bu tanımın seçilir kılınmasından sonra bir komutun çıktısı bir dosyaya gönderildiğinde, dosyanın eski içeriği aynı kalır. Bu tanım sayesinde kazara bir dosyanın üzerine başka bilgilerin yazılması önlenebilir.

```
$ ls -al dosya
-rw-r--r--   1 gorkem   users       1409 Nov 29 13:58 dosya
$ set -o noclobber
$ ls > dosya
bash: dosya: Cannot clobber existing file
```

ignoreeof : ignoreeof, kabuktan *control-d* tuş takımı ile çıkılmasını engeller. Yanlışlıkla basılan *control-d* ile sistemden anından çıkmak mümkün olduğu için kullanışlı bir tanımdır.

```
$ set -o ignoreeof
$ (control-d 'ye basin)
Use "logout" to leave the shell.
$ set +o ignoreeof
$ (control-d 'ye basin, sistemden hemen cikar)
$ ignoreeof=4
```

Yukarıdaki son örnekte 4 kere control-d tuşuna basılırsa kabuktan çıkılır.

6.4 Özel Kabuk Değişkenleri

Sisteme girdiğiniz zaman Linux size bir kabuk tahsis eder ve kabuk üzerinde değişkenler tanımlayabilmenize izin verir. Kabuk içinde bir kabuk programı (script) çalıştırılınca sistem tarafından bir alt kabuk daha yaratılır. Bu andan itibaren iki kabuk çalışır, birisi sisteme girdiğiniz anda tahsis edilen, diğeri de programın çalıştırılabilmesi için sistem tarafından çağırılan. Program bittiği anda alt kabuğun işlevi sona erer ve sistem tarafından öldürülür.

Bir kabuk altında tanımlanan değişken o kabuğa özgüdür. Bir kabuk altında erişilebilen değişkene diğer bir kabuk erişemez. Fakat bash kabuğu altında bir çevre değişkeni yaratılırsa, sistem tarafından her alt kabuğa bu çevre değişkeni kopyalanır. Böylece bir değişken yaratılır, ama her alt kabuk tarafından kullanılabilir. bash kabuğunda yaratılan çevre değişkenlerinin normal değişkenlerden tek farkı, bu değişkenin diğer alt kabuklar tarafından tanınmasıdır.

Bir çevre değişkeni belirtmek için `export` komutu kullanılır. Aşağıdaki örnekte *dosyam* isimli çevre değişkeni tanımlanıyor.

```
$ export dosyam="internet.97.txt"
$ echo $dosyam
internet.97.txt
```

Bu esnadan sonra tüm kabuklar altında bu değişken tanımlı olacaktır.

`env` komutu yardımıyla sistemde öntanımı yapılan veya sonradan tanımlanan tüm değişkenler ekrana listelenir.

```
$ env
```

```
HOME=/home/gorkem
SHELL=/bin/bash
LS_OPTIONS=--8bit --color=tty -F -b -T 0
PS1=\h:\w\$
PS2=>
LOGNAME=gorkem
OSTYPE=Linux
```

Sisteme girdiğiniz anda tanımlanan bazı komutlar ve açıklamaları aşağıda verilmiştir:

HISTFILE : Tüm yazdığınız komutlar `.bash_history` adlı bir dosya içinde tutulur. Her kullanıcının kendi ev dizinleri içinde kullanıcıya özgü bu dosyadan vardır. `$HISTFILE` değişkeni ile dosyanın ismi değiştirilebilir.

```
$ echo $HISTFILE
/home/gorkem/.bash_history
$ HISTFILE="komutlar"
$ echo $HISTFILE
komutlar
```

PATH : PATH değişkeninde bir komut yazıldığı anda sistem tarafından aranacak olan patika listesi görüntülenir. Örnek olarak `who` komutu `/usr/bin/` dizini altındadır ve bu bilgi PATH değişkeninde saklı tutulur. Kullanıcının yaptığı programları çalıştırabilmesi için PATH değişkeninde bulunduğu dizini de tanımlaması gerekir.

PATH değişkeninde her dizin iki nokta üstüste işareti ile birbirinden ayrılır. Örnek olarak `/usr/local/bin:/usr/bin` dizisi sırasıyla `/usr/local/bin` ve `/usr/bin` patikalarına karşılık gelir.

Kullanıcı PATH değişkenine yeni girdiler ekleyebilir. Aşağıdaki örnekte bunun nasıl uygulandığı görülüyor. Kullanıcı kısaca PATH değişkeninin sonuna kendisinin istediği patikayı veya patikaları ekler.

```
$ echo $PATH                       # PATH degiskenini ekrana bas
$ export PATH=$PATH:/usr/net       # buna /usr/net patikasini ekle
```

SHELL : O an kullanılan kabuğun patika ismini verir. Kabuk programları genellikle `/bin` dizini altında tutulurlar. Her kabuğun patika ismi sistemdeki `/etc/shells` dosyasında bulunmalıdır.

```
$ echo $SHELL
/bin/bash
```

HOME : Kullanıcının ev dizinini gösteren patikayı ekrana basar. Her kullanıcının ev dizini, sistemde hesap açılırken sistem görevlisi tarafından belirlenir. Linux Slackware dağıtımı altında bu dizin öntanımlı olarak `/home` 'dur. Aşağıdaki örnek kullanıcının ev dizinine ait patikayı ekrana basıyor.

```
$ echo $HOME
/home/gorkem
```

LOGNAME : Sistemdeki kullanıcı hesabınının ismini tutar. Her kullanıcı için farklı bir değerde olur.

```
$ echo $LOGNAME
gorkem
```

TERM : Kullanıcının halen üzerinde çalıştığı terminal tipini görüntüler. `vi` ve `pico` gibi editörler çalıştırıldıkları anda TERM çevresel değişkenine bakarlar. Bu sayede ekran ile ilgili bilgileri önceden düzenleyebilirler. Aşağıdaki komut önce halihazırdaki TERM değişkenini görüntülüyor, ardından değişkene başka bir değer atıyor.

```
$ echo $TERM
linux
$ export TERM=vt100
$ echo $TERM
vt100
```

PS1 ve PS2 : Ekranda komut satırının başına gelecek olan karakterleri belirlerler. Komut isteminin en başındaki yazı bu iki değişken yardımıyla değiştirilebilir. PS1, birinci satırın karakterlerini tutarken satıra sığmayan komutlar bir satır aşağı geçeceklerinden ikinci satırın karakterlerini de PS2 belirler. Aşağıdaki satır, bu iki değişkenin öntanımlı olan "$" işaretini bir kelimeyle değiştiriyorlar.

```
$ export PS1='-> '
-> export PS2='>'
->
```

Önceden tanımlanan kodlar yardımıyla komut satırı değiştirilebilir. Her kodun başında \ karakteri vardır. Bu kodların herbiri örneğin o anki saati, makinanın ismini veya bulunduğunuz dizini ekrana yazabilirler. Örneğin \h ve \w kodları sırasıyla çalışılan makinanın ismini ve çalışma dizinini gösterirler. Bu örnekte PS1 değişkeni bu iki kod yardımıyla değiştiriliyor.

```
$ export PS1='\h:\w\$ '
linux:~$                        # yeni komut satiri goruntusu
linux:~$ cd /tmp
linux:/tmp$                     # calisma dizinini de gosteriyor
```

Aşağıda, sıkça kullanılan kodların bir listesi görülüyor.

- **\t** , saat

- **\d** , tarih
- **\s** , kabuk ismi
- **\w** , çalışma dizini
- **\u** , kullanıcı ismi
- **\h** , makina ismi

6.5 Sisteme Giriş Dosyaları

Sisteme girerken her giriş anında çalıştırılan birtakım dosyalar bulunur. Bash kabuğu ile doğrudan ilgili olan giriş dosyaları arasında

```
/etc/profile
~/.bash_profile
~/.bash_login
~/.profile
~/.bashrc
```

sayılabilir. Kullanıcı bu dosyaların kendine ait olanlarını dilediği gibi değiştirebilir. Sisteme girerken tanımlamak istediği değişkenleri tanımlar, çalıştıracaği programları yazar. `/etc/profile` dosyasını tüm kullanıcılar ortak kullanır. Her kullanıcı sisteme girdiğinde sistem tarafından bu dosya çalıştırılır. Bu dosyada kullanıcıların ihtiyacına yönelik özel sistem değişkenleri bulunur , ek olarak birtakım kontrol ve kayıt işlemleri yapılır.

Örnek bir `/etc/profile` dosyası şekildeki gibidir.

```
if [ "$TERM" = "" -o "$TERM" = "unknown" ]; then
 TERM=linux
fi
#PS1='`hostname`:`pwd`# '
if [ "$SHELL" = "/bin/pdksh" -o "$SHELL" = "/bin/ksh" ]; then
 PS1="! $ "
elif [ "$SHELL" = "/bin/zsh" ]; then
 PS1="%m:%~%# "
elif [ "$SHELL" = "/bin/ash" ]; then
 PS1="$ "
else
 PS1='\h:\w\$ '
fi
PS2='> '
ignoreeof=2
export PATH DISPLAY LESS TERM PS1 PS2 ignoreeof
```

```
umask 022
# set up the color-ls environment variables:
if [ "$SHELL" = "/bin/zsh" ]; then
  eval ‘dircolors -z‘
elif [ "$SHELL" = "/bin/ash" ]; then
  eval ‘dircolors -s‘
else
  eval ‘dircolors -b‘
fi
```

`.bashrc` dosyası, her bash kabuğu veya alt kabuk çalıştırıldığı anda okunur. Her kabuk programı çalıştırılınca bir alt kabuğa ihtiyaç olduğundan bu durumda `.bashrc` dosyası da okunacak ve içerdiği değişkenler proram çalışmadan önce tanımlanacaktır.

Sistemden çıkarken varsa çalıştırılan dosyanın ismi `.bash_logout`'tur. Bu dosya ile kullanıcı sistemi terkederken fazladan işlemler yapabilir. Aşağıdaki örnekteki `.bash_logout` dosyasında kullanıcı sistemden her çıktığında ekran temizlenecek ve en üste 'Hoscakal' yazacaktır.

```
clear
echo "Hoscakal!"
```

6.6 Kabuk Programlamaya Giriş

Her kabuğun kendine özgü programlama dili yapısı vardır. Bash kabuğu ise güçlü programlama özellikleriyle karmaşık programların rahatça yazılmasına izin verir. Mantıksal operatörler, döngüler, değişkenler ve modern programlama dillerinde bulunan pek çok özellik bash kabuğunda da vardır ve işleyiş tarzları da hemen hemen aynıdır.

Genellikle, bir programı oluşturacak olan komutlar bir dosyaya yazılırlar ve ardından bu dosya çalıştırılır. Herhangi bir editör yardımıyla yazılan program, daha sonra kabuk altında çalıştırılır. Bir kabuk programı diğerlerini çalıştırabilir. Bu düzende kabuk programlarını daha karmaşık komutların biraraya gelmiş ve büyümüş haline benzetebiliriz.

6.6.1 Kabuk Programları

Kabuk programları, bir veya birden fazla linux komutunu tutan dosyalardır. Bu dosya yaratıldıktan sonra doğrudan dosyanın ismi girilerek veya dosya isminden önce '.' karakteri getirerek çalıştırılabilir. Bir kabuk programı, çalıştırma bitini 1 yapmak suretiyle "çalıştırılabilir" hale getirilir. `chmod` komutu yardımıyla bir programı çalıştırılabilir yapmak için ,

```
$ chmod +x komut-ismi
```

yazılabilir. Bundan sonra programın ismi yazılıp enter tuşuna basıldığı zaman bir program Linux komutuymuş gibi çalışacaktır.

```
$ cat calistir
echo -n "Tarih : "
date
$ chmod +x calistir
$ calistir
Tarih : Sun Dec  8 07:11:51 EET 1996
```

Yukarıdaki örnekte "calistir" isimli iki satırlık bir kabuk programının önce içeriği ekrana yazıldı, ardından çalıştırılacak duruma getirildi ve çalıştırıldı.

Kabuk programları yazarken dosyanın işlevini ve her satırdaki komutun veya komut kümesinin ne amaçla kullanıldığını gösteren açıklama satırları kullanmak işe yarar. Bir açıklama eklemek için satır başına (veya boş satıra) `#` işareti eklenir ve ardından istenilen cümle girilir. `#` işaretinden sonraki tüm satır kabuk tarafından gözardı edilir. Aşağıdaki programda komut öncesinde yeralan açıklama satırı, komut hakkında bilgi veriyor.

```
# gunzip komutu dosya acmak icin kullanilir.
gunzip sistem.gz
```

Yorum satırı, komutun sonuna da eklenebilir.

```
ps -aux         # sistem surecleri hakkinda ayrintili bilgi..
```

Bir kabuk altında çalışırken başka bir kabuk için yazılmış bir programı çalıştırmak mümkündür. Örneğin tcsh altındasınız ve daha evvel bash kullanarak yazdığınız bir programı çalıştırmak istiyorsunuz. Önce bash yazarak kabuk değiştirmeli, ardından programı çalıştırmalı, ve tekrar tcsh'a dönmelisiniz. Tüm bunları otomatik olarak yaptırabilirsiniz. Programın en başına `#!` karakterini, ardından programın çalışacağı kabuğun patikasını yazın. Örneğin `#!/bin/bash` komutunu programın en üstüne eklerseniz bu program bash kabuğu altında çalışacaktır.

6.6.2 Değişkenlerin Kullanımı

Bir değişkene değer atandığı anda sistem tarafından tanınır. Değişkenler alfabetik veya nümerik karakterlerden oluşabilirler fakat bir değişken sayısal bir değer ile başlayamaz. Bunların dışında değişken isminin içinde "_" karakteri de bulunabilir. Bir değişkene değer ataması "=" işareti yardımıyla yapılır.

```
$ mesaj="aksama yemege geliyorum"
```

İçeriği olan bir değişkene başına "$" işareti konularak ulaşılır. Aşağıda, `echo` komutu yardımıyla bir değişkenin içeriği ekrana basılıyor.

```
$ echo $mesaj
aksama yemege geliyorum
$ echo yarin $mesaj
yarin aksama yemege geliyorum
```

Aynı mesajı değişken kullanmadan da görüntüleyebiliriz.

```
$ echo "Aksama yemege geliyorum"
Aksama yemege geliyorum
```

6.6.3 Giriş/Çıkış İşlemleri

Bir kabuk programı çalışırken kullanıcıdan klavye yardımıyla bilgi girmesi sağlanabilir. Bu tür işlemler için tanımlanan `read` komutu klavyeyi okur ve aldığı bilgiyi bir değişkene atar. Aşağıdaki komutları içeren program yardımıyla klavyeden okunan değer ekrana yazılıyor. `echo` komutundan sonra birden fazla değişken grubu veya hem değişken, hem de dizi kullanılabilir.

```
echo Bir sayi giriniz..
read sayi
echo Girilen sayi : $sayi
```

Bazı durumlarda girilen değer özel karakterleri içerebilir. Bu durumda istenmeyen bazı sonuçların doğması kaçınılmaz olur. Aşağıdaki örneği bir dosya içine yazın ve dosyayı çalıştırdıktan sonra "*" tuşuna basın.

```
echo Bir karakter giriniz
read a
echo Girdiginiz karakter : $a
```

6.6.4 Aritmetik İşlemler

bash kabuğunda matematiksel işlemlere büyük sınırlamalar getirilmiştir. Tamsayı değişkeni dışında matematiksel değişken kullanmak için bu işlemler için geliştirilmiş ve kolaylıklar sağlayan awk veya bc kullanabilirsiniz.

Aritmetik işlemler için `eval` komutunu veya bash kabuğu altında yerleşik (builtin) komut olan `let` komutunu kullanabilirsiniz. Aşağıda let komutunun kullanımı görülüyor.

```
let "degisken=aritmetik islem"
```

Bu örnekte iki sayı çarpılıp çıkan sonuç başka bir değişkene yazılıyor.

```
$ let "carpim=2*7"
$ echo $carpim
```

Aritmetik değişken tanımlamanın diğer bir yolu da `typeset` komutu kullanmaktır.

```
$ typeset -i sonuc      (sonuc degiskeni bir dogal sayi icerecek)
$ a=100 ; b=56          (iki komutu ayirmak icin ; kullanilabilir)
$ sonuc=a*b
$ echo $sonuc
5600
```

6.6.5 if-else Kalıbı ve Kontrol İşlemleri

Hemen her programlama dilinde olan `if` kalıbı bir Linux komutunun çalışmasını kontrol (`test`) eder. `if` komutu yerleşik bir komuttur ve her `if`, bir `fi` komutuyla bitmelidir. `if` komutunun ardından gelen Linux komutu çalıştırılır ve komutun çıkış durumu (exit status) gözönüne alınarak ardından gelen `then` deyimiyle birlikte devamı işletilir. Genellikle komutun iki türlü çıkış durumu olacağından `else` komutunun ardından gelen komut zinciri, diğer çıkış durumunda çalıştırılır. Aşağıda if-else-then komutunun örnek sözdizimi görülüyor.

```
if linux komutu
    then
        komut1
        komut2
        ...
    else
        komut1
        komut2
        ...
fi
```

`if` komutu genellikle kendine `test` komutu ile birlikte kullanım bulur. Bu komut yardımıyla mantıksal işlemler yapılabilir, sayılar ve hatta diziler karşılaştırılabilir. Anahtar sözcük olan `test`'ten sonra opsiyonlar ve/veya karşılaştırılacak olan değerler yazılır. Her opsiyon bir mantıksal işleme karşılık gelir. Örneğin `-lt` opsiyonu ilk girilen aritmetik değişkenin ikinci değerden küçük olup olmadığını denetler. Benzer şekilde = opsiyonu da iki karakter kümesinin eşitliğini kontrol eder. Aşağıda `test` komutunun örnek kullanımı yeralıyor.

```
$ test 5 -eq 3
$ a="linux"
$ test $a="linux"
```

komutun işletilmesinin ardından kabuğa bir değer dönderilir. Bu değer komut başarılı olarak işletilmişse 0, değilse 1'dir. Son çalıştırılan tüm Linux komutlarının çıkış değeri $? değişkeninde tutulur. test komutunun çıkış değeri de bu yolla öğrenilebilir.

```
$ sayi=4
$ test $sayi -eq 4
$ echo $?
```

```
0
$ test $sayi -lt 2
$ echo $?
1
```

`test` komutu yerine parantezler de kullanılabilir. Yukarıdaki iki örnek, parantez kullanılarak şu şekilde yazılabilir:

```
$ [ $sayi -eq 4 ]
$ [ $sayi -lt 12 ]
```

Dikkat edilmesi gereken bir nokta, köşeli parantez kullanırken araya boşlukların eklenmesidir. Parantezler başlı başına bir komut olarak görüldüklerinden sağında ve solunda en az bir boşluk bırakılmalıdır. `test` komutunda sıkça kullanılan diğer seçenekler şunlardır :

Aritmetik karşılaştırmalar	
-gt	büyük
-lt	küçük
-ge	büyük eşit
-le	küçük eşit
-eq	eşit
-ne	eşit değil
Dizisel karşılaştırma	
-z	boş dizi
-n	tanımlı dizi
=	eşit diziler
!=	farklı diziler
Dosya karşılaştırması	
-f	dosya var
-s	dosya boş değil
-r	dosya okunabilir
-w	dosyaya yazılabilir
-x	çalıştırılabilir dosya
-h	sembolik bağlantı
-c	karakter aygıt
-b	blok aygıt
Mantıksal karşılaştırmalar	
-a	VE
-o	VEYA
!	DEĞİL

`if` komutunun `test` ile birlikte kullanılabildiğini daha önce belirtmiştik. Aşağıda bununla ilgili küçük bir örnek yeralıyor.

```
#!/bin/bash
echo "0 ile 20 arasinda bir sayi secin"
read sec
if [ $sec -lt 10 ]
   then
        echo "Secilen sayi tek basamakli"
   else
        echo "Secilen sayi cift basamakli"
fi
```

Her `if` komutu bir `fi` ile son bulmalıdır.

6.6.6 case Kalıbı

Birkaç alternatif arasından seçim yapmak için kullanılan bir komut olan `case`, bir eşleştirme gördüğü anda belirli bir komut kümesini işleme sokar. case yapısı `case` komutu ile başlar, eşleştirilecek olan anahtar sözcük yazılır ve seçenekler alt alta, her seçeneğe ait olan komutlarla birlikte belirtilir. Tüm yapı `esac` komutu ile son bulur.

```
case anahtar-sozcuk in
        secenek1)
                komutlar
                ;;
        secenek2)
                komutlar
                ;;
        *)
                komutlar
                ;;
esac
```

Seçenekler arasında özel karakterler *(*, [] , ? gibi)* kullanılabilir. Hiçbir eşleme yapılmadığı zaman *) seçeneği değerlendirilecek ve buna bağlı olan komutlar işletilecektir. * kullanımı isteğe bağlıdır. Aşağıda `case` komutuna ilişkin kısa bir örnek veriliyor.

```
#!/bin/bash

clear
echo    "1. ekrani temizle"
echo    "2. sistemdekileri goruntule"
echo    "3. dizindeki dosyalari goster"

echo -n "Secenegi giriniz : "
read secenek
```

```
case $secenek in
        1)
          clear
          ;;
        2)
          w
          ;;
        3)
          ls -al
          ;;
        *)
          echo Hatali secenek
esac
```

6.7 Döngüler

Diğer hemen tüm programlama dillerinin en büyük gücü olan döngü işlemlerine kabuk altında da izin veriliyor. Burada programcı tarafından en çok kullanılan 2 döngü tipi anlatılacaktır: `while` ve `for`. `while` komutu her döngüde bir denetleme mekanizmasını harekete geçirirken `for` döngüsü daha değişik bir mantıkla çalışır.

6.7.1 while-do Döngüsü

Döngü bloğu `while` anahtar kelimesiyle başlar, ardından gelen koşul sağlandığı sürece döngü işletilir. Önce koşulun sağlanıp sağlanmadığına bakılır. Döngüden çıkabilmek için mutlaka döngü içindeki koşul ifadesinin değerini yanlış yapacak bir durum oluşmalıdır, aksi halde sonsuz döngü oluşur.

```
while kosul ifadesi
   do
        komutlar
   done
```

`if` komutuyla birlikte kullanılan `test` komutu, `while` döngüsünde koşul ifadesi olarak da yer alabilir. Aşağıda 1'den 100'e kadar sayan ve ekrana basan bir döngü görülüyor.

```
#!/bin/bash
deger=0
while [ $deger -lt 100 ]
   do
        deger=$((deger+1))
        echo $deger
```

```
   done
```

Yukarıda kullanılan ((ve)) karakterleri arasına matematiksel bir işlem getirilebilir. Bu özellik bash kabuğuna özgüdür.

6.7.2 for-do döngüsü

Bir liste dahilindeki tüm değerlere sırayla erişimi sağlar. for komutundan sonra yeralan liste sırayla kullanılır ve herbirisi için döngü çalıştırılır. Listenin sonuna gelindiğinde ise döngüden çıkılır.

```
for degisken1 in deger1 deger2 ... degerX
   do
        komutlar
   done
```

Aşağıdaki örnek bu döngüyü kullanarak ekrana bir dizi kelime yazıyor. Döngü boyunca akasya, elma ve visne kelimeleri "agac" değişkenine kopyalanıyor ve her döngüde bu değişkenin içerdiği bilgiler ekrana yazılıyor.

```
for agac in akasya elma visne
   do
        echo $agac
   done
```

for-do döngüsü, dosya isimleri üzerinde yapılan işlemlerde de büyük kolaylıklar sağlar. Bunun için özel karakterlerden yararlanmak da olasıdır. Örnek olarak * karakteri o anki çalışma dizini içindeki tüm dosyaları seçer.

```
for a in *.pl ; do
        file $a
        done
```

6.8 Örnek Kabuk Programı

Sistem görevlisinin en çok kullandığım komutlardan birisi `ps -aux | grep -i xxx` komutudur. Bu satır yardımıyla çalışan *xxx* isimli program hakkında daha detaylı bilgi elde edilebilir. `ps` komutu detaylı bir süreç listesini ekrana verirken çıktı doğrudan grep komutuna yönlendirilir ve sadece istediğimiz bilgi monitörde görünür. Fakat her zaman aynı uzun satırı tekrar tekrar yazmaktansa bu satırı bir dosyaya gönderip, dosya adını komut satırından çalıştırmak zamandan tasarruf sağlayacaktır. Biz de öyle yapalım ve aşağıdaki satırları "p" isimli dosyaya yazalım.

```
#!/bin/bash
```

```
if [ $# = 1 ]
  then
  ps -ax | grep -i $1
else
  ps -ax
fi
```

Ardından dosyayı PATH değişkeninin işaret ettiği dizinlerden `/usr/local/bin` altına yerleştirip çalıştırılabilir olması için `chmod +x /usr/local/bin/p` komutunu uygulayın. Kendi yazdığınız dosyaları `/usr/local` veya `~/bin` dizini altına kopyalamanız bunların derli toplu olarak tek bir dizinde her an erişilebilir şekilde durmaları açısından önem taşır.

`grep` komutu, bir dosyadaki veya standart girişteki satırlar içinden belirli bir kelime, harf veya harf grubunu bulmaya yarar. Aşağıdaki birkaç örnek `grep`'i özetlemeye yeter.

```
$ grep while sort.c
sort.c dosyasinda while kelimesini iceren tum satirlari ekrana yazar.
$ w | grep demir
sistemde demir isimli kullanici var mi?
$ ls -al | grep "^d"
```

Son örnek, `ls -al` çıktısında satır başında "d" varsa ekrana getirir. Sadece dizin isimlerinin başında "d" harfi bulunur. Şapka isareti, satır başlangıcını gösterir.

Dosyaya biraz daha yakından bakalım. İlk satır, bu dosyanın `/bin/bash` programı tarafından çalıştırılacağını gösterir. İkinci satır yardımıyla komut satırı üzerinde kaç tane opsiyon olduğu bulunur. $# çevresel değişkeni her biri TAB veya boşluk karakteri ile ayrılmış komut satırı opsiyonları sayısını verir. Aşağıdaki komutta toplam 3 opsiyon vardır.

```
$ ls --8bit  -F -b
```

`$#` komutuna benzer şekilde $1, $2, $3 ... değişkenleri de opsiyonları verir. Yukarıdaki komutta $1, $2, $2 değişkenleri sırasıyla

```
--8bit
-F
-b
```

değerlerini alırlar. Komut satırından çalıştırılan komut ise $0 değişkenine atılır. Yukarıdaki örnekte $0 değişkeni `ls` değerini tutacaktır.

`p` dosyasına tek opsiyon yollayacağız. Bu opsiyon da hakkında detaylı bilgi alacağımız süreç olacaktır. Dosya içinde $1 değişkenini `ps` komutuna yerleştirmek için :

```
ps -ax | grep -i $1;
```

yazılır. Bundan sonra komut satırında, örneğin:

```
$ p bash
```

girilirse $1 değişkeni `bash` e eşit olacak ve bu da `ps -ax | grep -i bash` eşdeğer komutunu çalıştıracaktır. Komut olarak sadece `p` girilirse $# değişkeni 0'a eşit olacağından dosyada yeralan ve hiç bir filtreleme yapmayan `ps -ax` komutu çalışır.

Linux'u yüklerken *HOWTO* dökümanlarını da eklemeyi ihmal etmediniz, değil mi ? Aşağıdaki kısa program, bu dosyalara en hızlı şekilde ulaşmanıza yarıyor. Tüm HOWTO'lar kurulum aşamasında `/usr/doc/faq/howto` dizini altına kopyalanır ve sıkıştırılır.

Bu programı herhangi bir isimle kaydedin. Ardından komut satırından programın ismi ile birlikte ekranda görüntülemek istediğiniz *HOWTO* dökümanının ismini yazın (örneğin PPP). Bir parametre girilmez ise `/usr/doc/faq/howto` dizinindeki dosyalar ekrana gelir.

```
#!/bin/bash
if [ "$1" = "" ]; then
    ls /usr/doc/faq/howto | less
else
    gunzip -c /usr/doc/faq/howto/$1-HOWTO.gz | less
fi
```

Chapter 7

Sed ve Vi

7.1 SED (Stream editor)

Metin dosyaları üzerinde sürekli işlem yapanlar için *sed* ve *vi* hemen imdada yetişir. Bu iki programın yapabildiklerini etkileşimli olarak bir metin editörü üzerinden gerçekleştirebilmek de mümkündür. *sed* ve *vi* tam anlamıyla zaman kaybını önleyen kullanıcı dostu iki uygulamadır. Yapısal olarak diğer programlama dillerinden biraz farklı olduklarından öğrenmesi biraz zaman alır. Buna rağmen eğer metin editörlüğü işinizin bir parçası ise *sed* ve *vi*'ı öğrenip bunlardan verimli bir şekilde yararlanmanız uygun olacaktır.

7.1.1 Sed ve Özellikleri

Sed bir dosyadan veya standart girdiden(klavyeden) bilgi okur ve standart çıktıya (ekrana) okuduğu bilgileri kullanıcının belirlediği düzene sokarak yazar. Bu veriyi de kullanıcı genellikle bir dosyaya yerleştirir.

Sed komut satırında iken veya bir kabuk programı içinden kullanılabilir.

7.1.2 Örnekler

Sed kullanırken en çok başvuracağınız işleç '*s*' işlecidir. Bir karakteri veya karakter kümesini başka bir diziye çevirmeye yarar. Buna basit bir örnek :

```
sed 's/hapisane/hapishane/g' dosya
```

Komutun sonundaki '*g*' işleci, sed'e dosyanın tamamını araştırmasını söyler. Eğer bunu kaldırırsanız, sed her satırın başında bulduğu ilk değiştirmeyi yapacak ve satırın geri kalanına dokunmayacaktır.

Aşağıdaki karakterler sed tarafından kullanılan özel karakterler kapsamına girer. Bu karakterlerin herbirisinin anlamı olduğundan dikkatli kullanılmaları gerekir.

```
.*[]^$\&
```

Yukarıdaki karakterleri gözönüne almadan yapılan aşağıdaki gibi bir kullanım hataya yol açar.

```
sed 's/[J.S. Bach]/[Bach, J.S]/' dosya
```

Bunun yerine her özel karakterin başına '\' karakteri getirilerek özel anlamlarının yitirilmesi sağlanabilir (saklama işlemi). Yukarıda çalışmayan örneği yeniden düzenleyerek yazalım:

```
sed 's/\[J\.S\. Bach\]/[Bach, J.S ]/' dosya
```

Aynı anda birden çok değiştirme yapabilmek için ise her argüman için `-e` kullanılır.

```
sed -e 's/computer/bilgisayar/g' -e 's/server/sunucu/g' dosya
```

İçinde '/' karakteri olan bir diziyi değiştirmek için ne yapmalı? Bu, özel karakter sınıfına girdiği için karakterin özel anlamını yitirmesi sağlanır.

```
sed 's/\/usr\/bin/\/bin/g' dosya
```

7.1.3 Düzgün Deyimler (Regular Expressions)

Sed, UNIX üzerinde metin işleyen birçok program gibi düzgün deyimlerden yararlanır. Aşağıdaki bu duruma birkaç örnekle yer veriliyor. '^' karakteri satır başlangıcını gösterir. Bu komut,

```
sed 's/^Pts /Pazartesi/' dosya
```

bir satırın başında "Pts" varsa bunu "Pazartesi" haline getirir. Dikkat ederseniz komut dahilinde '*g*' işleci kullanılmamıştır. Çünkü her satırda sadece bir satır başı olabilir.

$ karakteri satır sonunu belirtir. Aşağıdaki örnekte,

```
sed 's/ $//' dosya
```

Sed komutu satır sonundaki tüm boşluk karakterlerini siler. Her satır sonuna "SATIR_SONU" karakter kümesini getirmek için

```
sed 's/$/SATIR_SONU/' dosya
```

yazılabilir. Boş bir satırı bulmak için satır başı ve satır sonu belirteçleri birlikte kullanılır.

```
sed 's/^$/bu daha once bos bir satirdi/' dosya
```

'.' (nokta) karakteri *herhangi bir karakter* anlamına gelir. Örneğin üzerinde işlem yapacağımız dosyada bütün satırlardaki ilk iki karakteri silmek için:

```
sed 's/^..//' dosya
```

Köşeli parantezler birden fazla sayıda karakter içinde seçim yapar.

```
sed 's/[Oo]pen[Ww]in/openwin/g' dosya
```

Alfabedeki iki karakterin arasındaki tüm karakterleri seçmek için köşeli parantezlerin içinde '-' karakteri kullanılabilir. Bu karakter, sağındaki ve solundaki de dahil olmak üzere aradaki tüm harfleri seçer. Örnek dosyadaki tüm büyük harfleri silelim.

```
sed 's/[A-Z]//g' dosya
```

Bir karakter kümesinin dışında kalan karakterleri seçmek için ise ' ^' kullanılır. Örnek dosyada, rakamların ve küçük harflerin dışındaki tüm harfleri silmek istiyoruz.

```
sed 's/[^1-9a-z]//g' dosya
```

'*' karakteri, *kendisinden bir önce gelen karakterin veya küme içine alınmış karakter gruplarının* **sıfır** *veya daha fazla tekrarı* anlamına gelir.

```
sed 's/^ *//' dosya
```

komutu her satır başındaki boşluk karakterlerini siler. Aşağıdaki sed komutu her satırdaki son kelimeyi (boşluk karakterlerinden oluşmayan karakterler grubunu) siler.

```
sed 's/[^ ]*$//' dosya
```

unutulmaması gereken bir nokta, '*' karakteri önceki grubun *sıfır* kere tekrarını da gözönüne alır, bu durumda aşağıdaki satırda görülen ve bir tamsayı değerini "TAMSAYI" sözcüğü haline getirmek için kullanılan komut Linux tarafından yanlış işletilecektir.

```
sed 's/ [0-9]* / TAMSAYI /g' dosya
```

Bu satır aynı zamanda her boşluk karakterini "TAMSAYI" ya çevirir. Bunu engellemek için, en azından bir haneyi arama kapsamına sokmak gerekir:

```
sed 's/ [0-9][0-9]* / TAMSAYI /g' dosya
```

'.* harf kombinasyonu *herhangi bir sayıda karakter grubu* anlamındadır. Aşağıdaki örnekte, "ku" ve "ik" arasında ne olursa olsun bulunan satırı değiştirecek ve bunların yerine "kuru erik" koyacaktır.

```
sed 's/ku.*ik/kuru erik/g' dosya
```

Buraya kadarki bölümde eşleştirdiğimiz harf veya harf gruplarını ya başka bir forma soktuk ya da tamamen sildik. Ancak sed'in kelime işleme kapasitesi sadece yaptıklarımızla sınırlı değildir. '*(*' ve '*)*' operatörleri yardımıyla eşleştirildiği anda hafızaya kaydedilen kelime grupları daha sonra tekrar çağırılabilir. Parantezler özel anlamlı karakterlerin anlamlarını yitirmesini sağlamak amacıyla daha önceki kullanıma uygun konulmuşlardır. İlk kullanılan '*(\\)*' operatörü '*1*' adlı alana, ikinci kullanılan '*(\\)*' operatörü '*2*' adlı alana eşleştirilir (kayıt edilir). Aşağıdaki sed komutu,

```
sed 's/^\([A-Za-z][A-Za-z]*\) \([A-Za-z][A-Za-z]*\)/\2 \1/' dosya
```

bir dosyadaki "isim soyisim" sırasını "soyisim isim" sırasına çevirmekte kullanılabilir. Örnek olarak içeriği aşağıdaki gibi olan bir dosya üzerinde bu komut çalıştırılırsa

```
Mehmet Kırk
Biray Çift
Gürkan Ballı
```

ekranda aşağıdaki çıktı görünür:

```
Kırk Mehmet
Çift Biray
Ballı Gürkan
```

Bazı durumlarda tüm dosya üzerinde değil de bazı özellikleri sağlayan satırlar üzerinde işlem yapmak isteyebilirsiniz. Bunun için 'adresleme' yapmanız gerekir. Adresleme işlemi, '*s* işlecinden hemen önce yapılır. Bu komut ile dosyanın sadece 1 ve 20. satırları arasında işlem yapılacaktır.

```
sed '1,20s/Karadeniz/Akdeniz/g' dosya
```

"Makina" kelimesiyle başlayan tüm satırlardaki "Endüstri" kelimesi yerine "Elektronik" kelimesini yerleştirelim.

```
sed '/^Makina/s/Maden /Elektronik /g' dosya
```

"Makina" kelimesiyle *başlamayan* tüm satırlara benzer işlemi uygulayalım. '*!*' karakterinin yerine dikkat edin.

```
sed '/^Makina/!s/Maden /Elektronik /g' dosya
```

7.1.4 Sed Kaynak Dosyası

Eğer sed kaynak satırınız bir iki satıra sığmayacak kadar büyük ise komutları tek dosya altında toplayabilirsiniz. Bu örnekte `degistir.sed` isimli dosyaya üç satırlık sed komutu yazalım.

```
s/tiren/tren/g
s/Pazar/Pazartesi/g
s/1996/1997/g
```

Ardından sed komutunu `-f` parametresiyle çalıştıralım.

```
sed -f degistir.sed dosya
```

Bunların yerine kendi kendine çalışan bir sed programı yazmaya ne dersiniz ?

```
#!/usr/bin/sed -f
# degistir.sed dosyasi
s/tiren/tren/g
s/Pazar/Pazartesi/g
s/1996/1997/g
```

Çalıştırmak için dosya izni verdikten sonra

```
chmod u+x degistir.sed
```

Komut satırı üzerinden rahatlıkla çalıştırabiliriz

```
$ degistir.sed dosya
```

7.2 vi Editörü

Bir linux sistemde her kullanıcının zevkine hitap eden çeşitli metin editörler vardır. Bunların arasında en çok kullanılan ve standart olan ed ve vi, UNIX sistemler için özel olarak tasarlanmış editörlerdir. vi kadar çok kullanılmasa da ed, pico ve joe da büyük bir taraftar kitlesine sahiptir. vi ilk başta karmaşık görünse de hızı ve verimi ile her kullanıcının işini büyük ölçüde kolaylaştıracak bir editördür. En çok kullanılan komut takımlarını öğrendikten sonra vi ile aynı dili konuşuyor olduğunuzu göreceksiniz.

7.2.1 vi'a Başlangıç

PC editörlerin büyük çoğunluğunda klavye, editör komutlarını almak ve basılan tuşları ekrana göndermek için kullanılan iki gruba ayrılmıştır. Kullanıcı aynı anda hem komut işletip hem de yazı yazabilir. Bu tip editörler geniş bir klavye haritasına sahip oldukları için, örneğin fonksiyon tuşlarını veya klavyenin en sağındaki nümerik tuşları da kullanıyor olabilirler. vi' ın tasarım aşamasında bu durumdan kaçınılmış, standart dışı klavyesi olan bilgisayarların da var olabileceği düşünülerek klavye üzerinde kullanılması gereken tuşların sayısını mümkün olduğunca indirilmesine çalışılmıştır. Burada karşımıza vi editörünün çalışma aşamasında üç ayrı işlev geliyor.

Birincisi, bilgisayara komutların girdisi sırasında kullanılan *komut modu*, ikincisi yazı yazarken kullanılan *yazı modu* ve *satır modu*.

Komut modunda klavye üzerinde görevi olan tüm tuşlar, bilgisayar komut vermek için kullanılıyor. Yazı modunda ise diğer editörlerdekine benzer şekilde yazı yazmak mümkün oluyor. Klavye modunu değiştirdiğinizde klavye tuşlarının işlevleri de hemen değişiyor. vi editörünü ilk çalıştırdığınız anda komut moduna girersiniz. Bu anda her tuşa ait bir komut çalıştırılmaya hazırdır (örneğin '*j*' bir karakter aşağı, '*k*' bir karakter yukarı gider). Bu anda kullanıcı yazı moduna geçmek isterse '*i*' tuşuna basabilir.

Yazı moduna iken klavyeden girilen her karakter ekranda görünür. Bundan sonrası daktilo kullanmaya benzer. Tekrar komut moduna geçmek için *ESC* tuşu yardımcı olur. Bir dosya yazarken veya düzeltmede bulunurken her iki yazım stili arasında sürekli gidip geldiğinizi farkedeksiniz.

Bunlara ek olarak dosya işlemleri veya eşleme (substitution) yaparken iki mod da kullanılmaz. Linux kullanıcısı vi üzerinde çalışırken en alt satırda vi mesajlarını görür ve gerektiği zaman satır modunda komutları girer. Satır moduna geçmek için '*:*' (iki nokta üstüste) karakteri kullanılır. Bu tuşa basıldığında ekranın en altında bir satır açılır ve vi sizden bir komut girmenizi bekler. vi komutunu yazdıp enter'a bastıktan sonra tekrar editör komut moduna geçersiniz.

7.2.2 Dosya İşlemleri

Sisteme girip ekrana `vi telefon` yazdığınızda aşağıdakine benzer bir metin üzerinde arama ve değiştirme görüntüsü monitörde belirecek ve program çalıştırılacaktır.

```
_
~
~
~
~
~
~
~
~
~
~
~
~
 "telefon" [NEW FILE]  1 line, 1 char
```

vi'a ilk girdiğiniz anda komut modunda olursunuz. Yazı moduna geçmek için '*i*' veya '*a*' tuşuna basın. Bu andan itibaren zamanınızın büyük bir kısmını geçireceğiniz yazı moduna atlarsınız. Bıkana kadar yazıp önce *ESC* tuşuna, ardından iki kere '*Z*' tuşuna basın. Bu komut dizisi önce dosyayı kaydedecek, sonra da vi editörden çıkmanızı ve kabuğa dönmenizi sağlayacaktır.

vi' a yeni başlayanların yaptıkları en sık hatalardan biri yazı modundan komut moduna geçerken *ESC* tuşuna basmamalarıdır. '*w*' komutu '*ZZ*' komutuna benzer çalışır, hafızada halihazırda bulunan dosyayı, editörden çıkmadan diske kaydeder. Eğer isimsiz bir dosyada üzerinde çalışıyorsanız (örneğin vi telefon yerine sadece vi yazarak editöre girmişseniz) dosyayı sabit diske yazmak için '*w*' satır komutunu kullanmalısınız. Bunun için daha önce de belirtildiği gibi

```
:w telefon
```

yazıp satır moduna geçerek komutu girmelisiniz.

vi editörden çıkmak için '*:q*' yazın.

7.2.3 vi' da Yazma

Şimdi vi yardımıyla bir dosya üzerinde denemeler yapalım. Aşağıdaki satırları kabuk satırında iken yazın ve *ENTER*'a basın.

```
metu:~$ vi deneme
```

Ekran silinecek ve yukarıdakine benzer bir görüntüyle karşılaşacaksınız. Yazı moduna geçmek için '*i*' tuşuna basın. Hemen ardından bir zamanlar İnternet üzerinde görüp kaydettiğim birkaç satırı yazın.

```
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
~
~
~
~
~
~
```

Ekranın en sonuna geldiğinizde vi, alttaki satır başına geçmeyecektir. Bunun için *ENTER* tuşuna basarak yazmaya devam edin.

```
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan tembelliğe alışmışız anlaşılan,
~
~
~
~
~
```

Satır sonuna geldiğiniz zaman komut satırına geçmek için *ESC* tuşuna basın. Klavyenin sağındaki yön tuşlarını kullanarak dosya içinde gezinebilirsiniz. vi'ın sevdiğim özelliklerinden birisi de kullanıcıyı 10 parmak yazmaya zorlamasıdır. Yön tuşlarının görevini *h, j, k* ve *l* tuşları da yapabilir, böylece kullanıcı sağ elini klavye üzerinde sürekli gezdirmek zorunda kalmaz.

- **j**, imleci bir satır aşağı indir

- **k**, imleci bir satır yukarı gönder
- **h**, imleci bir karakter sola al
- **l**, imleci bir karakter sağa al

Alternatif olarak *ENTER* tuşu komut modunda '*j*' tuşu yerine de kullanılabilir. Kursörü yön tuşları yardımıyla "Fazla" kelimesiyle başlayan satıra kadar taşıyın. '*i*' tuşuna basıp tekrar yazı moduna geçerek metin üzerinde biraz değişiklik yapın:

```
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı_tembelliğe alışmışız anlaşılan..
~
~
~
~
~
```

"dolayı" kelimesini ekledikten sonra bir noktayı işaret etmek istiyorum. Kullandığım Linux makinası Türkçe destekli olduğu için Türkçe karakterleri okuyup yazmam mümkün, aslında Linux işletim sistemi hemen her ülkenin alfabesini desteklediğinden herhangi bir dilde (örneğin Alman alfabesi karakterlerini kullanarak) yazmak için birkaç komut yeterli oluyor.

'*i*' tuşu imlecin bulunduğu yerden yazmasına rağmen, '*a*' komutu bir karakter sağdan başlayarak yazmanızı sağlar. '*o*' komutu ise imlecin bulunduğu yerin altında bir satır boşluk bırakır ve boş satırın başına geçerek yazı moduna atlar. *ESC* tuşuna basarak komut modundayken en alt satıra inin ve '*o*' tuşuna basın. Burada bir satır daha ekleyelim:

```
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
Sayfaları karıştırmak zor geliyor...
~
~
~
~
```

Eğer ekrandaki bilgi birkaç sayfadan ibaretse *Control-F* tuşunu bir sayfa ileri gitmek için, *Control-B* tuşunu ise metni bir sayfa geriye almak için kullanabilirsiniz.

vi her satırı kendi içinde numaralandırır. Belirli bir satıra gitmek için tuşlar dışında satırın numarası kullanarak da gidilebilir. '*G*' komutu yardımıyla imlecin gideceği satırı belirtebilirsiniz. Bunun için komut modunda iken satır numarasını girin (ekranda görünmeyecektir) ve hemen ardından '*G*' tuşuna basın. Eğer satır numarası girilmezse imleci dosyanın sonunda konumlandırılır.

7.2.4 Silme ve Kopyalama

Yazı modunda iken *back space* tuşu yardımıyla, komut modunda iken ise '*x*' komutu ile istediğiniz kadar harfi silmeniz mümkündür. '*dd*' komutu ise imleci üzerinde bulunduğu satırı siler ve alttaki tüm satırlar bir satır yukarı alınır. PC üzerindeki editörlerdeki satır sonuna gelince imlecin otomatik olarak bir satır aşağı kayması, `:set wm=n` satır komutu yardımıyla gerçekleştirilir. Komuttaki n sayısı, sağdan kaç boşluk bırakılacağını söyler. Ekranın sağından 4 satır boşluk bırakmak için (bu durumda satır uzunluğu 76 karaktere inecektir) `:set wm=4` yazın.

'*dd*' veya '*x*' komutundan sonra silinen harf ve harf grupları hafızaya alınır ve istenildiği zaman '*p*' tuşu yardımıyla imlecin bulunduğu yerden itibaren ekrana basılır. '*P*' yardımıyla hafızadaki metni imlecin bir satır üzerine kopyalayabilirsiniz.

Kursörün bulunduğu yerden satır sonuna kadar silmek için '*d$*' , metin sonuna kadar silmek için '*dG*' kullanılabilir. ($ ve G harflerinin işlevlerine dikkat edin!)

Son kaldığımız yerden devam edelim. Son satırı '*dd*' ile silelim.

```
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
~
~
~
~
~
```

Ardından kestiğimiz satırı *P* komutuyla en üste yapıştıralım .

```
Sayfaları karıştırmak zor geliyor...
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
~
~
~
~
```

Tek bir karakteri başka bir karakterle değiştirmek için önce komut modunda karakteri silip yazı moduna geçtikten sonra gerekli değişikliği yapmaya gerek yoktur. '*r*' komutu tek bir harfi değiştirmek için kullanılır. Bu tuşa basınca imlecin üzerinde bulunduğu karakter silinir ve editör kullanıcıdan yerine gelecek karakteri bekler. Bundan sonra vi tekrar komut moduna döner.

Bir kelimeyi silip üzerine yazmak için de '*cw*' komutu kullanılabilir. Kelimenin başına gelin ve sırayla önce '*c*' sonra '*w*' tuşlarına basın. Editör yazmayı bitirmenizi bekleyecektir. *ESC* tuşuna basıncaya kadar yazdıklarınız ekranda kalacak ve (varsa) kelimeden arta kalan harf(ler) silinecektir.

Her ölümlü gibi siz de hata yapabilirsiniz. Hata yaptıktan sonra eski konuma dönmek için '*u*' komutunu kullanın. Bu komut, üzerinde değişiklikler yaptığınız metni son çalıştırılan komuttan önceki duruma getirir. Yeni konuma tekrar geri dönmek için *Control-R* 'ye basın.

7.2.5 Komut Tekrarı

20 satırı silmek için 20 kere dd tuşuna basmak gereksizdir. Bunun için '*dd*' komutundan önce klavyeden 20 rakamını girmek yeterli olur. Girilen rakamlar '*G*' komutundaki örneğe benzer şekilde ekranda görünmezler. Kullanılan rakamlar, kendinden sonra gelecek olan komutun kaç kere

işletileceğini belirtir. Örnek olarak '*8x*' komutu sekiz karakteri siler, '*12j*' komutu 12 satır aşağı iner. Üzerinde çalıştığımız dosyada birkaç uygulama yapalım. Kursörü ilk satırdaki "geliyor" kelimesinin başına alın ve '*7x*' komutunu girin. Son durum yaklaşık şu şekilde olmalıdır:

```
Sayfaları karıştırmak zor ...
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
~
~
~
```

vi'ın komut tekrar özelliği sayesinde ekrana yazdıklarınızı ardarda kopyalayabilirsiniz. Örnek dosya üzerinde birinci satırın başına gidin ve '*9i*' yazın. Gireceğiniz yazı modunda ekrana ne gönderirseniz *ESC* tuşundan sonra 9 kopyası basılacaktır:

```
bip bip bip bip bip bip bip bip bipSayfaları karıştırmak zor ...
Arada sırada kitapta bir şeyler ararken /'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
~
~
~
~
```

7.2.6 Kopyalama

Silinen karakter veya karakter gruplarının hafızaya kaydedildiğini söylemiştik. Bu hafıza geçici bir tampon vazifesi görür. '*yy*' komutu üzerinde bulunduğu satırı kopyalar. Eğer komuttan önce sayı girilirse o kadar satır kopyalanır. Ardından kullanıcı metin üzerinde istediği yere gelip '*p*' tuşuna basarak hafızadaki satırları imlecin bulunduğu satırın altından itibaren basar. '*yy*' komutu yerine '*Y*' de tercih edilebilir.

7.2.7 Arama ve Eşleştirme

vi'da çalışırken arama yapmak isteyebilirsiniz. Eğer üzerinde çalıştığınız metin uzun ise belirli bir kelime veya kelime gruplarını aramak çok zaman alabilir. Komut modunda iken '/' karakterine basarak arama/tarama moduna geçin. Ekranın en alt kısmında bir satır açılacaktır. Aramak istediğiniz kelimeyi yazın ve *ENTER* tuşuna basın. vi editörü imlecin bulunduğu yerden itibaren girilen anahtar sözcüğü metin içinde tarayacaktır. Eğer aranan kelime bulunmuşsa satır bunun üzerine konumlandırır.

'/' komutu dosyanın sonuna doğru arama yaparken '*?*' komutu başa doğru gider. Aynı kelimeyi tekrar aratmak için '*n*' tuşuna basın. Arama işlemi, '/' komutu yardımıyla yapılmışsa ileri doğru, '*?*' komutu yardımıyla yapılmışsa geriye doğru tekrar başlayacaktır.

Aslında arama yaparken komut modunda girilen her karakter grubu bir düzgün deyimdir. Örnek olarak komut modunda '*d$*' yazdığımızda $ harfinin satır sonunu belirttiğini, '*dG*' komutundaki

'*G*' harfinin de metin sonunu işaret ettiğinden imlecin bulunduğu yerden itibaren dosya sonuna kadar sildiğini görmüştük. Metin içindeki büyük harfleri aramak için

```
/[A-Z]
```

kullanılabilir. Burada gösterebildiğimiz özellikleri vi'ın sadece küçük bir parçasıdır. vi kullanıcılarının işine yarayabilecek birkaç arama-eşleştirme komutuna göz atalım.

```
:[a,b]s/deyim/eşlenecek deyim/işleç
```

Yukarıdaki komut metin içinde a ve b satırları arasındaki "deyim"i, "eşlenecek deyim"e çevirir. Özel tanımlı karakterleri de kullanabilirsiniz. Aşağıdaki komut da 1. ve 10. satırları arasında bulduğu tüm "nümerik" kelimelerini "sayısal" hale getirir.

```
:1,10s/nümerik/sayısal;
```

Satır sayısı yerine tüm dosya üzerinde eşleştirme yapmak için % sembolünü, satır sonunu belirtmek için $ sembolünü kullanabilirsiniz. Bu durumda satır sayısını belirten [a,b] tanımını boş bıraktığınızda değişiklik sadece imlecin bulunduğu satırda yapılacaktır. İşleçler de birkaç çeşit olabilir. '*c*' işleci, her eşleme öncesi kullanıcıdan izin alır. vi kullanırken çoğu durumda her satırda rastlanan ilk eşlemeyi yerine getirecek olan '*g*' işlecini kullandığınızı göreceksiniz. Daha az kullanılan '*i*' işleci ise aramanın büyük-küçük harf gözetimi yapmadan yerine getirmesini sağlar.

```
:%s/elektronik/s/nik//g
```

komutu, içinde "elektronik" kelimesi geçen her satırdaki "elektronik" kelimesini "elektron" haline getirir. Fakat "mekanik" kelimesine dokunmaz. Sed altında kullanılan '\(\)' operatörü vi için de geçerli bir tanımdır. Aşağıdaki komut da yukarıdaki uygulamaya benzer şekilde her "kalemlik" kelimesini "kalemler" haline getirir.

```
:g/\(kalem\)lik/s//\1ler/g
```

7.2.8 Diğer Dosyaların Metne Eklenmesi

Kursörün bulunduğu yerden itibaren `rehber.txt` dosyasını halihazırda çalıştığınız metine kopyalamak için

```
:r rehber.txt
```

komutunu kullanın.

7.2.9 Kabuk Komutlarının Çalıştırılması

vi altında iken kabuk komutlarını, hatta kabuğun kendisini bile çalıştırmak mümkündür. '*:!*' ile komut ismini girebilirsiniz. Aşağıdaki komut vi'dan çıkmadan `ls` komutunu çalıştıracak ve sonucu ekrana verecektir.

```
:!ls -la
```

Bunun yanısıra komut çıktısını ekran yerine üzerinde çalıştığımız metine aktarmak için aşağıdaki vi komutunu kullanın:

```
:r! ls -al
```

Metnin son halini görelim:

```
bip bip bip bip bip bip bip bip bipSayfaları karıştırmak zor ...
Arada sırada kitapta bir şeyler ararken / 'e basasım gelmiyor değil.
Fazla vi kullanmaktan dolayı tembelliğe alışmışız anlaşılan..
bash.txt
index.html
images/
~
```

7.2.10 vi Başlangıç Dosyaları

vi her açıldığında belirli bir dosyayı okur ve gerekli düzenlemeleri ve düzeltmeleri yapar. Bu işlemler için kullanılan set komutu birtakım parametreler alır. Örneğin `set wm=n` , *n* karakter kenar boşluğu bırakmak için, set nu ise her satırın numaralandırılması için kullanılır. Bu komutlar ister vi içinde , ister başlangıç dosyası içinde çalıştırılabilir.

vi her açılışta öntanımlı olarak önce bulunduğunuz dizin içindeki, ardından ev dizininizin içindeki `.exrc` dosyasına bakar. Böylece açılış anında konfigürasyon işlemlerini bu dosyayı kullanarak kolayca gerçekleştirebilirsiniz. Dosyaya her satırda bir set komutu gelecek şekilde istedinildiği kadar komut yerleştirilebilir. Aşağıda örnek bir .exrc dosyası görülüyor.

```
set ignorecase
set wrapmargin=2
set tabstop=6
```

Yukarıdaki seçenekler arasında `ignorecase`, vi editörünün arama yaparken küçük-büyük harf ayrımı gözetmemesini; `wrapmargin=2` sol kenardan iki satır boşluk bırakmasını; `tabstop=6` ise *TAB* tuşuna her basıldığında imlecin 6 karakter ileri gitmesini sağlar. Daha kısa yazmak isterseniz sırasıyla `ic,` `wm` ve `ts` kullanın.

Chapter 8

TCP/IP

Diğer açık sistemler gibi Linux'un da en büyük özelliklerinden biri ağ protokolleri ve uygulamalarının sistemin en doğal parçalarından biri olmasıdır. Dolayısıyla ağ yapısını bilmeyen bir yöneticinin, açık sistemler üzerindeki bilgisini daha da geliştirebilmesi mümkün değildir. Bu bölümde anlatılacaklar Linux üzerinde TCP/IP'ye bir giriş yapılmasını sağlayacaktır. Ayrıntılı bilgi TCP/IP servislerinin ve uygulamalarının anlatıldığı diğer bölümlerden ve aşağıda sözü edilen kitap ve dökümanlardan edinilebilir.

TCP/IP üzerine önerebileceğimiz bir kitap, ilk baskıları TÜBİTAK tarafından bastırılan, ve büyük bir okuyucu kitlesi bulmuş *Herkes İçin İnternet* (Kürşat Çağıltay) 'tir. Bunun dışında İngilizce Prentice Hall tarafından yayınlanan *Internetworking with TCP/IP Volume I (D.E. Comer)*, O'Reilly & Associates tarafından yayınlanan, *TCP*IP Network/ Administration (Craig Hunt) ve *Linux Network Administrator's Guide* (NAG - Olaf Kirsh). NAG, *Linux Documentation Project*'in bir parçası olduğundan elektronik kopyası İnternet üzerinden ücretsiz olarak edinilebilir.

Linux Documentation Project grubunun hazırladığı *NET-3-HOWTO*, TCP/IP temelleri ve ağ uygulamalarını örnekleriyle işlemiş. Bu dökümanı nasıl temin edebileceğinizi kitabın sonunda bulabilirsiniz. NET-3, TCP/IP ve buna bağlı protokollerin Linux için geliştirilen bölümüne verilen isimdir.

8.1 Temel TCP/IP

TCP/IP, ilk defa ABD'de ARPANet (Advanced Research Projects Agency Network) adı altında, askeri bir proje olarak geliştirildi. Önceleri askeri amaçlı düşünülen proje önce ünivesiteler tarafından kullanılmaya başlandı. Ardından ABD'nin dört bir yanında birbirinden bağımsız geliştirilen ağlar, tek bir omurga altında NSFNet olarak adlandırıldı ve ulusal boyutu aşarak dünyaya yayıldı. İnternet'in doğuşu da bu tarihe denk gelir.

Bir sokak üzerinde yeralan evlerin adresleri gibi, İnternet'e bağlı olan her makinanın da bir adresi vardır. Bu adres sayesinde bir bilgisayardan diğerine ulaşmak mümkün olur. İnternet adresi 4 bayttan (32 bit) ibaret olup yazılırken her bayt arasına bir nokta konulur. En çok kullanılan bu gösterim şekline örnek olarak, ODTÜ'deki bir bilgisayarın adresi olan `144.122.199.20`'yi verebiliriz.

Bununla beraber, 4 baytlık numaraların kolayca hatırlanmasının mümkün olmadığından, İnternet üzerindeki makinalara alfanümerik adlar da verilebilir. Yukarıda örneğini verdiğim adresin ismi, *knidos.cc.metu.edu.tr*'dır. ilk noktaya kadar olan kelime, makina adıdır(yukarıdaki örnekte knidos), bundan sonraki noktayla ayrılmış bölümler özelden genele doğru makinanın ait olduğu kurum, kurumun tipi ve ülke gibi bilgiler içerir. Bu adlandırma yöntemiyle ilgili ayrıntılı bilgiyi Alan İsimlendirme Sisteminin (Domain Name System - DNS) anlatıldığı bölümde bulabilirsiniz.

8.2 İnternet Adresleri

İnternet üzerinde 3 sınıf adres vardır. Avrupa'da RIPE (Réseaux IP Européens) tarafından dağıtılan bu adresler daha sonra o yerin ağ yöneticisi tarafından uygun şekilde bölünebilir. Bu bölümlendirmeye "subnetting" işlemi adı verilir. Bu sayede ağlar gruplanarak herbirisinin yönetimi bağımsız hale getirilmiş, aynı zamanda da kısıtlı olan IP adresleri daha verimli bir şekilde kullanılmış olur.

Üç çeşit İnternet adresi şunlardır :

- **A sınıfı İnternet adresi:** Adresin ilk baytı 1 ile 126 arasında bir sayıdır. Bu adrese verilen yetkiyle toplam 2ˆ24 makina adreslenebilir. Dünya üzerinde 126 tane A sınıfı adres vardır.
- **B sınıfı İnternet adresi:** Adresin ilk baytı 128 ile 192 arasında bir sayıdan oluşur. Bu adresin subnetlere bölünmesiyle 65534 farklı makina adreslenebilir.
- **C sınıfı İnternet adresi:** Adresin ilk baytı 192 ile 223 arasındadır. C sınıfı bir adresi blokuyla bağlı 254 bilgisayar adreslenebilir.

A ve *B* sınıfı adreslerin hepsi dağıtılmış ve şu anda intenet *C* sınıfı adreslerde de sıkıntı çekilmektedir. Adres yetersizliğine çözüm getirmek amacıyla *IPv6* ya da *IPNG* adlarıyla daha uzun İnternet adresleri kullanan protokoller geliştirilme ve test aşamasındadır.

Bir İnternet adresi iki kısımdan meydana gelir: *ağ adresi* ve *düğüm adresi*. Her makina, bir ağ üzerinde bulunur ve bu adres "ağ adresi" olarak adlandırılır. Üç sınıf (A, B ve C) İnternet adresinin ağ ve düğüm adresleri farklı farklıdır. A sınıfı İnternet adreslerinde ağ adresini 1 bayt tayin eder. Örnek olarak hayali yazılan 74.198.59.33 makinasını tanımlayan ağ adresi 74'tür. Ağ adresi uzun halde yazıldığı zaman kalan 3 baytın yerine 0 konur. Bu durumda yukarıda adı geçen makinaya ait ağ adresi 74.0.0.0 olacaktır.

Ağ adresinden geriye kalan düğüm adresi, bir makinanın İnternet sınıfına göre 1,2 veya 3 bayttan ibaret olabilir. Örnek olarak,

195.12.288.3 makinasının (C sınıfı) ağ adresi 195.12.288.0, düğüm adresi 3'tür.

130.11.195.62 makinasının (B sınıfı) ağ adresi 130.11.0.0, düğüm adresi 195.62'dir.

Düğüm adresleri yerine 4 baytı da kullanmak da olası. Yukarıdaki iki örnek için düğüm adreslerini sırasıyla 195.12.288.3 ve 130.11.195.62 olarak kabul edebiliriz.

Yukarıda sözedilen *subnetting* (alt ağlara ayırma), çok sıkça kullanılan bir yöntem olup getirdiği birtakım kolaylıklar vardır. Büyük bir iletişim ağı alt ağlara ayrılırsa, kontrol edilmesi daha kolaylaşır.

Bir kuruluş, kendine ait olan B sınıfı adresi subnetlere bölmek isteyebilir. Gerekli düzenlemeleri yaparak bir B sınıfı adresi (örneğin) 255 adet alt adrese ayırabilir. ODTÜ'nün NIC'den aldığı 144.122.0.0 ağı, 254 parçaya bölünmüş, bu parçalardan 144.122.71.0 alt ağı Bilgisayar Mühendisliği bölümüne, 144.122.34.0 alt ağı Kimya Mühendisliği bölümüne verilmiştir. Bundan sonra her bölüm kendi ağı ve üzerindeki makinalardan sorumlu olur.

8.3 Protokoller

Internet protokollerinin yaygınlaşmasındaki en önemli etkenlerden biri herhangi bir ağ donanımına ya da firmaya özel olmamalarıdır. Bilgisayarınız, işletim sistemiz ya da ağ bağlanma yönteminiz ne olursa olsun intenete bağlamanız mümkündür. İnternet protokolleri RFC adı verilen dökümanlarda açıklanır, bu dökümanlar herkese açıktır ve ücretsizdir. RFC'lere *NIC* `<ftp://nic.ddn.mil/rfc/>` ya da *ODTÜ'deki yansısından* `<ftp://ftp.metu.edu.tr/pub/documents/rfc/>` erişebilirsiniz.

Internet protokollerinin temeli *Intenet Protocol (IP)*'dir. İnternet üzerinde yönlendirme(routing) gibi temel ağ işlemlerinin gerçekleştirildiği protokol katmanıdır. IP paketlerinin her biri kendi başlarına aradaki ağ cihazları tarafından yönlendirileren paket içinde belirtilen adrese ulaştırılır. Bu sırada fiziksel ağ farklılıklarından kaynaklanan paket parçalanmaları(fragmentation) ve bunların yeniden birleştirilmeleri aradaki ağ cihazlarının aşırı yüklenmelerini önlemek gibi görevler de IP katmanı tarafından gerçekleştirilir.IP bağlantı temelli(connection oriented) bir ağ protokolü değildir. Bunun yanı sıra IP paketlerin içeriklerinin doğruluğunu da garanti etmez. IP katmanı sadece başlık kısmında oluşan hataları bulur ve düzeltir. Internet üzerinde yönlendirme, yukarida sözü edilen adreslerden yararlanılarak yapılır.

Kullanıcı uygulamalarının IP katmanına doğrudan ulaşımları yoktur. IP ve uygulama programları arasındaki bağlantıyı sağlayan iki protokol vardır: *Transmission Control Protocol (TCP)* ve *User Datagram Protocol (UDP)*. *TCP*, İnternet protokolleri arasındaki en önemli protokollerden biridir (İnternet protokol ailesi bu nedenle TCP/IP diye adlandırılır.). TCP, IP katmanının sağlamadığı bağlantı temelli, güvenilir servisi sağlar. TCP kullanarak ağ üzerinden veri aktaran programlar, bir dosyadan okuyormuş ya da yazıyormuş gibi güvenle ağ bağlantısını kullanabilirler. Arada oluşan hatalar TCP tarafından onarılır. IP protokol katmanına uygulma programları doğrudan erişemediklerinden, hata kontrolü ve bağlantı gerektirmeyen ya da bu işlemleri kendileri gerçekleştirmek isteyen uygulamalar *UDP* kullanarak ağ üzerinden iletişim sağlarlar.

Yukarıdaki protokollerin yanı sıra İnternet standardı olmuş birçok uygulama protokolü de vardır. Bunlar arasında, *TELNET*, *FTP*, *SMTP* ve *HTTP* gösterilebilir.

İnternet üzerinde paketler son makinaya ulaştıktan sonra, ilgili uygulama programına ulaşabilmesi için *port* adı verilen sanal numaralar kullnılır. Servis veren uygulamalar, önceden belirlenmiş standart port numraları kullanırlar. Örnek olarak SMTP 25 numaralı TCP portunu, talk ise 518 numaralı UDP portunu kullanır. Sisteminizin kullandığı port numaraları ve bunların isimlerini `/etc/services` dosyasından görebilirsiniz.

8.4 İnternet'e Bağlanma

Linux'u İnternet'e bağlamak ilk aşamada biraz zor görünse de biraz teori, biraz pratik yardımıyla mantık kolayca anlaşılabiliyor. Ağlar, bilgisayar sistemlerinde hata olma olasılığı en yüksek, en çok sorunun çıktığı alandır. Bu bölüm, özelde bir ethernet kartı ile bağlantı yapıldığı varsayılarak anlatılmıştır. Ancak anlatılan birçok kavram herhangi bir yapıdaki İnternet bağlantısı için geçerli olacaktır.

Öncelikle çekirdeği -daha önce yapmamışsanız- ağ bağlantısına destek verecek şekilde tekrar derleyin. Bunun için gerekli birkaç soruya "y" cevabı vermek yeterli olacaktır :

```
CONFIG NET ? [Y/n] y
...
TCP/IP networking (CONFIG_INET) [Y/n/?] y
...
Network device support (CONFIG_NETDEVICES) [Y/n/?]
...
Ethernet (10 or 100Mbit) (CONFIG_NET_ETHERNET) [N/y/?] y
...
Other ISA cards (CONFIG_NET_ISA) [N/y/?] y
...
NE2000/NE1000 support (CONFIG_NE2000) [N/y/m/?] y
```

Yukarıdaki örnekte ethernet kartınızın ne2000/ne1000 ya da uyumlu bir kart olduğu var sayılıyor. Kartınızın marka/modeline uygun seçeneneğe 'y' cevabı vermelisiniz. Ethernet kartlarının dışında bağlanma yönteminize bağlı olarak çekirdeği *PPP*, *SLIP*, *Token Ring* gibi bir destekle derlemeniz gerekebilir. Bunları *modül* olarak derlemeniz de mümkün. Çekirdek derlemek konusunda daha ayrıntılı bilgiyi bu kitabın ilgili bölümünde bulabilirsiniz.

Tabi ki çekirdeği derlemek yetmiyor, derleme aşamasından sonra ağ desteğine sahip olan çekirdek ile makinanın açılması gerekir.

Bağlantıyı gerçekleştirmek için ağ sorumlusundan bazı önemli bilgileri almalısınız. Makinanızın *IP adresi* bunların başında gelir. Diğer gerekli bilgiler de *subnet mask* ve *broadcast adresi*dir.

DNS adres çözümleyici olarak çalışan DNS makinası ve makinanızı İnternet'e bağlayacak olan yönlendiricinin IP adreslerini de biliyor olmalısınız.

Bu aşamadan sonra konfigürasyon işlemlerine geçilebilir.

8.4.1 Ağ Konfigürasyonu

Bu bölümde anlatılan `rc.*` konfigürasyon dosyaları *Slackware* dağıtımına özeldir. Diğer dağıtımlarda da açılışta çalışan benzeri *rc* dosyaları değiştirilerek, ya da sistemle birlikte gelen sistem yönetim araçları kullanılarak istediğiniz konfigürasyonu yapabilirsiniz.

Linux, açılırken bazı dosyalara bakarak düzgün çalışması için gerekli konfigürasyonu yapar ve bazı yazılımları çalıştırır. Bu dosyalar `/etc/rc.d` dizini altında yeralır. İnternet bağlantısı için sadece iki dosyaya gerek duyulur. Slackware dağıtımı içinde bunlar, konfigürasyonu yapan `/etc/rc.d/rc.inet1` ve temel ağ programlarını çalıştıran `/etc/rc.d/rc.inet2` dosyalarıdır.

Bu dosyadaki değerleri gerektiği şekilde düzenleyerek İnternet'e bağlanacağız. Öncelikle rc.inet1 dosyasının içeriğine göz atalım:

```
HOSTNAME=‘cat /etc/HOSTNAME‘

/sbin/ifconfig lo 127.0.0.1
/sbin/route add -net 127.0.0.0

IPADDR="144.122.71.55"
NETMASK="255.255.255.0"
NETWORK="144.122.71.0"
BROADCAST="144.122.71.255"
GATEWAY="144.122.71.1"

/sbin/ifconfig eth0 $IPADDR broadcast $BROADCAST netmask $NETMASK
/sbin/route add -net $NETWORK netmask $NETMASK
/sbin/route add default gw $GATEWAY metric 1
```

ifconfig komutu, ağ bağlantısı yapan kartın belirli parametreler ile konfigürasyonunun sağlar. Kart kimliğini bilgisayara tanıtmak için bu komut kullanılır.

İlk satırdaki ifconfig programı, makinanın kendisiyle haberleşmesini sağlamak amacıyla kendi ağ adresini (loopback device) 127.0.0.1 olarak tanıtmıştır. İnternet üzerindeki her makinanın loopback adresi 127.0.0.1 'dir. 127 ile başlayan hiç bir adres, İnternet üzerinde gerçek bir adres olarak kullanılamaz.

Her paket, gönderileceği adrese gitmeden önce bir tabloya bakar. Bu tablo, paketin muhtemel rotasını içerir. route komutu yardımıyla bu tablo oluşturulur veya kontrol edilir. Genellikle bir kart için bir `ifconfig` kullanılırken, paketin yollanacağı birden fazla rotanın olması durumunda `rc.inet1` dosyasında birden fazla route komut girdisi yapılacaktır.

Kullanılan her ağ kartının bir veya birden fazla internet adresi olabilir. Linux'a takılan birinci kart `eth0` ismini alırken ikinci karta eth1, üçüncüye eth2 ... denir. Loopback cihazının ismi `lo` dur. ifconfig komutunu parametre vermeden kullanırsanız, Linux'a tanıtılan çevresel birimlerin listesi ekrana gelir.

```
$ ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Bcast:127.255.255.255  Mask:255.0.0.0
          UP BROADCAST LOOPBACK RUNNING  MTU:3584  Metric:1
          RX packets:493 errors:0 dropped:0 overruns:0
          TX packets:493 errors:0 dropped:0 overruns:0

eth0      Link encap:10Mbps Ethernet  HWaddr 48:54:33:00:E8:3A
```

```
inet addr:144.122.71.55  Bcast:144.122.71.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:171956 errors:0 dropped:0 overruns:0
TX packets:113212 errors:3 dropped:0 overruns:0
Interrupt:3 Base address:0x300
```

cclub.metu.edu.tr makinasının ifconfig bilgisi yukarıdaki şekildedir. Bu makina üzerinde tek ethernet kartı olup, ifconfig bilgisinde ikinci sırada yeralıyor. Kartla ilgili detaylı bilgi (her kart özgü donanım numarası, alınan ve verilen paket sayısı gibi) ifconfig çıktısında var.

`route` komutunun genel kullanımı şu şekildedir:

```
route [add | del] [-net | -host] varis_adresi [gw yonlendirici] secenekler...
```

host: paketin gideceği yer başka bir makina
net: paketin gideceği yer başka bir ağ
varış_adresi: paketin gideceği yer
yönlendirici: bu paketin gidebilmesi için geçmesi gereken yönlendirici
add: tabloya ekle
del: tablodan sil
Tablonun genel durumunu kontrol etmek için `route` komutuna parametre vermeden komut satırından yazın:

```
cclub:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
localnet        *               255.255.255.0   U     0      0       67 eth0
loopback        *               255.0.0.0       U     0      0        3 lo
default         router71.ceng.m 0.0.0.0         UG    1      0      404 eth0
```

Örnek tabloda üç ayrı yönlendirme girişi vardır. Birincisi, makinanın kendisi ile haberleşmesini sağlayacak olan loopback. İkinci satırda, yerel ağa gidecek olan tüm paketlerin yönlendiriciye verilmeden ağa yönlendirilmesi söylenmiş. Son sırada ise (default) paket ilk iki satırda yeralan varış adresinden farklı bir adrese gidecekse, önce yönlendiriciden geçip sonra dış dünyaya açılacağından yönlendiricinin IP adresi yazılmıştır.

`ifconfig`, biraz karmaşık bir komuttur. Sistem yöneticisinden öğrenmeniz gereken subnet maskesi, broadcast adresi ve IP adresini kullanıp kartların konfigüre eder.

```
ifconfig eth0 144.122.71.55 netmask 255.255.255.0 broadcast 144.122.71.255
```

Yukarıda `ifconfig` yardımıyla eth0 kartına hem bir IP adresi verilmiş, hem de bilgisayarın bulunduğu ağın özellikleri tanıtılmıştır.

Bu andan sonra istendiği kadar `route` komutu ile makinadan çıkacak paketler için yeni yollar tanımlanabilir.

```
route add -net ...
route add -host ...
```

Dosyanın içeriğini sistem görevlisinin verdiği bilgiler doğrultusunda değiştirdikten sonra komut satırından çalıştırmanız, İnternet'e bağlantı için yeterli olacaktır.

```
# /etc/rc.d/rc.inet1
```

Bağlantıyı kontrol etmek İnternet adresini bilip çalıştığından emin olduğunuz bir makinaya telnet ile bağlantı yapmaya çalışın :

```
$ telnet orca.cc.metu.edu.tr
Connected to orca.cc.metu.edu.tr
Escape character is ^]
```

Ethernet kartının bağlantısını yazılım yoluyla kesmenin yolu `ifconfig` 'in down seçeneğidir. Örneğin tek ethernet kartı üzerinden İnternet bağlantınız var ve bunu hemen kesmek için aşağıdaki satırı yazın.

```
# ifconfig eth0 down
```

ethernet bağlantısını yeniden açmak için

```
# ifconfig eth0 up
```

komutunu girmelisiniz. Bu işlem sonrasında yönlendirme bilgilerini yeniden girmeniz gerekebilir.

`ping` komutu, bağlantıları ve aynı zamanda bağlantı hızını ölçmek için kullanılan bir komuttur. Ping, ICMP protokolü yardımıyla hedef makinaya belirli uzunlukta paketler gönderir. Karşıdaki makina bunları aldığı anda bekletmeden geri yollar. Arada geçen zaman farklı, iki makina arasındaki bağlantı hızı hakkında fikir verir.

```
cclub:~# ping knidos.cc.metu.edu.tr
PING knidos.cc.metu.edu.tr (144.122.199.20): 56 data bytes
64 bytes from 144.122.199.20: icmp_seq=0 ttl=253 time=34.0 ms
64 bytes from 144.122.199.20: icmp_seq=1 ttl=253 time=487.7 ms
64 bytes from 144.122.199.20: icmp_seq=2 ttl=253 time=3.6 ms

--- knidos.cc.metu.edu.tr ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3.6/175.1/487.7 ms
```

`netstat` komutu, makinanın ağ kartı üzerinden yapılan bağlantıların dökümünü çıkarır.

```
cclub:~# netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 cclub.metu.edu.tr:1947  ftp02.quicktime.appl:20 ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1943  despair.thebandit.c:ftp ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1424  despair.thebandit.co:20 ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1932  ftp02.quicktime.app:ftp ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1407  165.113.58.253:20       ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1392  165.113.58.253:ftp      ESTABLISHED
tcp        0      0 cclub.metu.edu.tr:1377  ppp23.cc.metu.ed:telnet ESTABLISHED
tcp        0      0 cclub.metu.edu.t:telnet pclabI_23.feas.met:7075 ESTABLISHED
tcp        1      0 cclub.metu.edu.t:telnet ppp23.cc.metu.edu.:1054 TIME_WAIT
tcp        0      0 cclub.metu.edu.t:telnet labmonitor.feas.m:10345 ESTABLISHED
tcp        0   9495 cclub.metu.edu.t:telnet a5-03-asy27.bil-ro:1026 ESTABLISHED
tcp        0      0 cclub.metu.edu.t:telnet pclabI_14.feas.me:15694 ESTABLISHED
tcp        0      0 cclub.metu.edu.t:telnet stu21.math.metu.e:16158 ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  1      [ ]         STREAM     CONNECTED     232
unix  1      [ ]         STREAM     CONNECTED     299
unix  1      [ ]         STREAM     CONNECTED     327
unix  1      [ ]         STREAM                   591
```

İkinci önemli dosya olan `/etc/rc.d/rc.inet2`, belli başlı tüm İnternet sunucularını çalıştırır. Aşağıda bu dosyanın bir kısmı görülüyor.

```
#!/bin/sh

NET="/usr/sbin"
IN_SERV="lpd"
LPSPOOL="/var/spool/lpd"

echo -n "Starting daemons:"

# inetd calistiriliyor.
if [ -f ${NET}/inetd ]; then
  echo -n " inetd"
  ${NET}/inetd
else
  echo "no INETD found.  INET cancelled!"
  exit 1
fi

# degisik internet servisleri sirasiyla calistiriliyor.
```

```
for server in ${IN_SERV} ; do
  if [ -f ${NET}/${server} ]; then
    echo -n " ${server}"
    ${NET}/${server}
  fi
done
```

Bu açılış dosyası, ağ üzerinden servs veren bir dizi sunucu programı çalıştırıyor ve hataları engellemek için programların yerinde olup olmadığına da bakıyor.

8.4.2 Kontrol Dosyaları

Konfigürasyonu tamamlamak için `/etc` dizini altında yeralan bazı dosyaların elden geçirilmesi gerekir. Bu dosyalar sistemin daha güvenli ve verimli çalışmasına yardım eder. Bu dosyaların belli başlı olanları ve ne işe yaradıklarını görelim.

`/etc/hosts` dosyasında makinaların IP numaraları ve İnternet adresleri yazılır. Buraya sürekli kullanılan belli başlı önemli makinaların girişi yapılabilir. Her makinanın uzun isimleri yerine kısa bir kelime yazılarak sözkonusu makinaya ulaşılabilir. Örnek bir `/etc/hosts` dosyası :

```
127.0.0.1              localhost
144.122.199.20     knidos.cc.metu.edu.tr knidos
```

İlk satırda loopback IP numarası ve buna karşılık gelen makina ismi vardır. Bu satırı silmemelisiniz. İkinci satırda ise knidos.cc.metu.edu.tr makinasının IP numarasını 144.122.199.20 olarak belirtilmiş ve bu bilgisayara "knidos" ismi ile de ulaşılabilmesi sağlanmıştır.

`/etc/hosts.allow` ve `/etc/hosts.deny` konfigürasyon dosyaları, başka bir makinadan Linux'unuza yapılan bağlantıları denetler. `/etc/hosts.deny` içine, sınırlandırılma getirmek istediğiniz İnternet servislerini ve hangi makinalar için sınırlandırma istediğinizi yazabilirsiniz.

```
$ cat /etc/hosts.deny
#

in.fingerd: ALL except localhost, localhost : \
        echo "request from %d@%h" >> /tmp/req;
in.telnetd : .net.tr
```

Yukarıdaki `hosts.deny` dosyasında dışarıdan gelebilecek finger istekleri filtreleniyor ve bu istekler hakkında detaylı bilgi (hangi kullanıcı, hangi makinadan) `/tmp/req` dosyasına ekleniyor.

Sonraki satır `in.telnetd` yazılımını çalıştıran isteklerin *.net.tr adreslerinden gelmesi halinde bunların gözardı edileceğini belirtir.

`hosts.allow` dosyasında, genellikle `hosts.deny` dosyasına gruplanan makina isimlerinin sınırlandırma istenmeyen bir veya birkaç tanesi yazılır. Eğer bir makina ismi ve karşılık gelen İnternet adresi her iki dosyada da varsa, sözkonusu servise izin verilir.

Daha fazla bilgiyi `hosts_access(5)` man sayfalarından bulabilirsiniz.

`/etc/resolv.conf` dosyasında alan adı ve alan adı sunucusunun IP numarası yazılır.

```
$ cat /etc/resolv.conf
domain linux.org.tr
nameserver 144.122.199.20
```

Yukarıdaki satırlardan üzerinde bulunduğum makinanın alan adının (domain name) linux.org.tr, alan adı sunucusunun ise 144.122.199.20 IP numaralı knidos makinası olduğu anlaşılıyor.

8.5 Yönlendirme

Daha önce de belirtildiği gibi birden fazla ağa fiziksel bağlantısı bulunan ve ağlar arası bilgi paketlerinin geçişine olanak tanıyan araçlara yönlendirici (router) adı verilir.

İki ayrı ağda yeralan makinaların haberleşmesine imkan tanıyan yönlendiriciler, kendilerine gönderilen paketleri bir ağdan diğerine iletirler.

Herhangi bir ağ içerisinde yer alan makinalar aynı IP ağ içerisinde bulunan makinalarla (ağ konfigürasyonları doğru olduğu sürece) haberleşebilirler. Başka adreslere ulaşabilmek için bulunulan ağ dışındaki ağlarla bağlantısı olan ve o ağ üzerinde bulunan bir makina ile haberleşmeleri gereklidir. Bu makina bilgisayar tanımlarında *gateway* (yönlendirici, arabirim, arayol) olarak verilir.

Minimum yönlendirme, bir ağa bağlı olmayan makinalarda, `ifconfig` yardımıyla yapılan yönlendirmedir. En çok kullanılan statik yönlendirme çeşidinde, az sayıda arayol ile dışarıya bağlı olan bir makinada, route komutuyla yapılır. Aynı adrese ulaşan birden fazla arayol olması durumunda ise, kendi yönlendirme tablosunu ağın durumuna göre değiştirebilen dinamik yönlendirmeye başvurulabilir.

Linux'un yönlendirici olarak kullanılmasına bir örnek verilebilir. Elimizde var olduğunu bildiğimiz iki ağ olsun. Bu ağlardan birincisini, 144.122.71.0 ağı, diğeri de 144.122.1.0 ağı olarak kabul edelim. 144.122.71.0 ağının İnternet'e bağlanabilmesi için 144.122.1.0 ağına ulaşması gereklidir. Yönlendirici olarak kullanılacak makinayı 71.0 ağında kurup gerekli yönlendirme talimatları verildiği anda bu ağa bağlı olan diğer makinalar da İnternet'e açılabilecektir.

Örnek olarak 144.122.71.0 ağı üzerindeki bir makina için ifconfig satırı

```
ifconfig eth0 144.122.71.1 broadcast 144.122.71.255 netmask 255.255.255.0
```

olarak görünüyor. Bu makinanın yönlendirici görevi yapabilmesi için `route` komutu kullanılır. Bu sayede bu bilgisayar, diğerlerine ulaşacaktır.

```
route add -net 144.122.71.0 netmask 255.255.255.0 dev eth0
```

Bu, 144.122.71.0 ağında olan makinaların eth0 kartı üzerinden erişilebileceğini gösterir. Diğer adreslere erişebilmek için ise bir çıkışa (gateway) ihtiyaç duyulmaktadır.

```
route add default gw 144.122.71.1 dev eth0
```

Bu satır sayesinde eldeki tablo yardımıyla çözümlenemeyen adreslere gitmesi gereken tüm paketler 144.122.71.1 üzerinden gönderilmektedir. Bu durumda bu makinanın paketleri nereye göndereceğini bildiği varsayılmaktadır.

Bu makina içinde iki adet ethernet bağlantısı bulunmaktadır. Bunlardan biri yönlendiriciyi 144.122.71.0 ağına bağlarken diğeri 144.122.1.0 ağına bağlamaktadır. İlk olarak bu iki kart için gereken tanımlamalar /etc/rc.d/rc.inet1 içerisinde aşağıdakine benzer şekilde tanımlanmıştır.

```
ifconfig eth0 144.122.71.1 broadcast 144.122.71.255 netmask 255.255.255.0
ifconfig eth1 144.122.1.200 broadcast 144.122.71.255 netmask 255.255.255.0
```

Birinci ethernet kartı için eth0, ikinci ethernet kartı için de eth1 kısaltmaları kullanılabilir. Doğal olarak her iki ağ için bir 'route' satırı gerekmekte :

```
route add -net 144.122.71.0 netmask 255.255.255.0 dev eth0
route add -net 144.122.1.0 netmask 255.255.255.0 dev eth1
```

Bu sayede diğer makinalardan farklı olarak yönlendirici hem 144.122.1.0 ağı, hem de 144.122.71.0 ağı üzerindeki adreslere direk ulaşabilmektedir. Diğer adreslere ulaşabilmek için benzer bir yöntem ile 144.122.1.0 ağı üzerindeki arayola (gateway) yönlendirme yapılır.

```
route add default gw 144.122.1.200 dev eth1
```

Bu çeşit yönlendirmeye **statik yönlendirme** adı verilir. Burada tüm bağlantılar bir tablo halinde verilmiştir. Birbiri arkasına ağ eklendiği zaman statik yönlendirme basitlik avantajını kaybetmeye başlar. Bu amaçla yönlendiricilerin kendi aralarında bildikleri yolları (yönlendirme bilgilerini) birbirlerine aktardıkları dinamik yönlendirme kullanılır.

Ağa bağlanmak amacıyla birden fazla kart kullanıyorsanız, *arp* komutunu incelemenizi tavsiye ederim. Bu komut pekçok opsiyon alır ve sistem yöneticisinin Linux üzerindeki ağı kontrolünü kolaylaştırır.

```
# arp -a
           IP address      HW type                  HW address
           191.72.1.3      10Mbps Ethernet          00:00:C0:5A:42:C1
           191.72.1.2      10Mbps Ethernet          00:00:C0:90:B3:42
           191.72.2.4      10Mbps Ethernet          00:00:C0:04:69:AA
```

8.5.1 Birden Fazla Ethernet Kartının Tanıtılması

Yukarıda örneğini verdiğimiz Linux yönlendirici üzerinde iki ethernet kartı bulunuyor. Bu sayede Linux makinası iki ethernet ağı arasında bir yönlendirici olarak kullanılabilmektedir. Bir Linux'a birden fazla ethernet kartı takmak mümkündür. Bu kartları takarken aşağıdaki maddelere dikkat etmek gerekir:

1. Çekirdek içerisinde her iki kart için destek bulunmalıdır.
2. Ethernet kartları sistemdeki diğer IO ve IRQ ayarlarını çakışmayacak şekilde ve farklı değerlerde olmalıdır.

Birden fazla ethernet kartı (ya da gibi herhangi iki veya daha fazla PPP, SLIP gibi ağ bağlantısı) olan makina bir yönlendirici olarak kullanılacaksa, çekirdek *IP FORWARDING* desteğiyle derlenmelidir. Bunun yanı sıra Linux'unuzun birinci görevi yönlendiricilik olacaksa *optimize as router* sorusuna da '*y*' yanıtı vermeniz yararlı olacaktır.

```
IP: forwarding/gatewaying (CONFIG_IP_FORWARD) [N/y/?] y
...
IP: optimize as router not host (CONFIG_IP_ROUTER) [N/y/?] y
```

Linux açılırken çekirdek tanıdığı donanımları teker teker arayarak bulmaya çalışır (probing). Birden fazla ethernet kartı kullanımına sık raslanmadığından Linux ilk ethernet kartını bulduktan sonra ikinci ve daha fazla ethernet kartı aramadığından iki kartınız bile olsa Linux sadece birini görecektir. Çekirdeğe ethernet kartı ile ilgili parametreler vererek birden fazla ethernet kartının açılışta bulunması sağlanabilir. LILO kullanıldığı zaman `/etc/lilo.conf` içinde

```
append="ether=0,0,eth1"
```

Yukarıdaki paramatre ile çekirdek ikinci bir ethernet kartını (`eth1`) da arayacaktır. Ayni şekilde arka arkaya `ether=....` parametreleri vererek varsa üçüncü, dördüncü ve daha fazla ethernet kartının aratılması da mümkündür. Ethernet kartlarınızın adreslerini biliyorsanız, kartların yanlış sırayla bulunması tehlikesini de önlemek için,

```
append="ether=5,0x340,eth0 ether=15,0x300,eth1"
```

gibi bir satır eklenmesi yeterli olacaktır. Burada eth0 isimli bağlantı IRQ 5 ve IOBASE 0x340 ve diğer ethernet bağlantısı eth1 IRQ 15 ve IOBASE 0x300 olacak şekilde ayarlanır, bu durumda çekirdek bir daha kartların hangi adres ve IRQ'da olduğuna dair tahmin yürütmez, girilen değerlerde bulunan kartları bulmaya çalışır.

Sistem açıldıktan sonra gerçekten istenen kartların istenen adreste çalıştığına emin olmak için `/var/adm/messages` dosyasına bakabilirsiniz, ya da `dmesg` komutuyla çekirdek mesajlarını görebilirsiniz.

8.6 NFS

NFS (Network File System), bir ağ üzerinden disklerin paylaşımını sağlamak üzere geliştirilmiştir. NFS sayesinde bir makinada yeralan belirli bir disk bölümü, başka makinalar tarafından okunabilir veya yazılabilir. Bu işleme özellikle büyük organizasyonlarda, disk alanından tasarruf etmek için başvurulur. Her makinada aynı dosyalar, çalıştırılabilir programlar olması yerine bunlar tek sunucuda toplanır, diğer bilgisayarlar bu alanı ortaklaşa paylaşırlar. Paylaşılan dizin, sanki yerel makinanın bir diziniymiş gibi davranır. Kendi disk alanını paylaştıran makinaya NFS sunucusu, bu diske erişim yapan makinalara da NFS istemcisi adı verilir.

NFS kullanımının pek çok avantajı sayılabilir. Bunlar kısaca:

- Yerel bir işletmede tüm kullanıcılar tek bilgisayar üzerindeki aynı disk alanını paylaşabilirler. Diğer bilgisayarlar açıldıktan sonra bu diskleri kendi dosya sisteminlerine bindirebilirler. NFS'in NIS ile birlikte kullanımı sayesinde bir kullanıcı herhangi bir makinaya girip diğer kullanıcılarla birlikte aynı dosya yapısı ile çalışabilir.
- Paylaşım sayesinde büyük disk alanı isteyen programlar tek bir sunucuda toplanabilir ve bu sayede önemli miktarda diskten tasarruf sağlanabilir.
- Bir makina üzerinde yapılan değişiklik, her makinada yapılmış gibi olur. Böylece aynı yazılımı örneğin 20 bilgisayara yüklemek zorunda kalmazsınız.

8.6.1 Ağ Üzerinde Paylaşım

NFS'in çalışması, yerel bir diskin mount edilmesi kadar kolaydır. Örnek olarak NFS sunucusu olan aspendos makinasında yeralan `/home` dizinini, yerel makinadaki `/users` dizinine yerleştirmek için :

```
# mount -t nfs aspendos:/home /users
```

kullanılır. mount, karşı makinanın erişim izinleri doğru olduğunu teyit ettiği anda bu işlemi gerçekleştirir. Sunucu ile istemci arasındaki mesaj alış-verişini `mountd` programı üstlenir.

NFS kullanabilmek için çekirdekte NFS desteği olması şarttır. Çekirdeğin hangi dosya sistemlerini desteklediğini `/proc` dosya sistemi altındaki filesystems dosyasını okuyarak anlayabilmek mümkündür. Eğer aşağıdaki satırlarda nfs kelimesi geçmiyorsa çekirdeğe NFS desteği verilerek tekrar derlenmesi gerekir.

```
$ cat /proc/filesystems
        minix
        ext2
        msdos
nodev   proc
nodev   nfs
```

NFS için mount programının genel kullanımı şu şekildedir :

```
mount -t nfs <sunucu-adresi:paylasma-dizini> <yerel-dizin> -o <secenekler>
```

Sistem açılırken `/etc/fstab` dosyasında yeralan bilgilerle uzak makinanın diski otomatik olarak paylaştırılabilir. Mount programı bazı seçenekleri destekler. Bu seçenekler `-o` yardımıyla komut satırında verilebildiği gibi /etc/fstab içinde de belirtilebilir. Her iki durumda da birden fazla seçenek kullanılırsa bunlar birbirlerinden virgülle ayrılırlar.

Açılış sırasında *news* makinasındaki /usr/spool/news dosyasını yerel makinadaki /usr/spool/news dizini altına yerleştiren /etc/fstab satırı

```
news:/usr/spool/news  /usr/spool/news   nfs   timeo=20
```

şeklinde yazılır. NFS ile kullanılan timeo seçeneğinin karşısında yeralan sayı (saniyenin onda biri olarak), bağlantının kurulabilmesi için geçecek maksimum zamanı gösterir. Bu zamanın aşılması halinde sunucuyla bağlantı kesilir.

8.6.2 NFS Kurulumu

Yerel makinanın sunucu olabilmesi için nfsd ve mountd programlarının çalıştırılması gerekir. Bu yazılımlar açılış esnasında aşağıda görülen /etc/rc.d/rc.inet2 dosyasından başlatılırlar. telnetd, fingerd ve benzeri süreçlerin aksine bu iki program inetd tarafından yönetilmezler. Dikkat etmeniz gereken bir durum da nfsd ve mountd programlarının rpc.portmap programından sonra çalıştırılması gerektiğidir :

```
if [ -x /usr/sbin/rpc.mountd ]; then
# (rpc.mountd programi var ve calistirilabilir mi ?)
        /usr/sbin/rpc.mountd; echo -n " mountd"
fi
if [ -x /usr/sbin/rpc.nfsd ]; then
# (rpc.nfsd programi var ve calistirilabilir mi ?)
        /usr/sbin/rpc.nfsd; echo -n " nfsd"
fi
```

Sunucunun hangi dizinlerini, hangi istemci(ler) tarafından paylaşılabileceği /etc/exports dosyasında belirtilir. Örnek bir exports dosyasının görünümü aşağıdaki gibidir :

```
/home           davul.cclub.metu.edu.tr, ordek.cclub.metu.edu.tr
/usr/ftp        (ro)
/cdrom          *.edu.tr
```

Her satır, yerel makinadaki dizin ismi ile başlar, bu dizini mount etmeye izin verilen bilgisayar isimleri ile devam eder. Makina isimlerinde ? veya * joker karakterleri kullanılabilir. Dosyanın üçüncü satırı, sonu .edu.tr ile biten tüm makinaların /cdrom dizinine ulaşabileceğini belirtir.

Aynı dizini paylaşacak birden fazla makina adı, aralarında virül konarak yazılabilir. Parantez içinde yeralan ro kelimesi, dizinlerin salt okunur mount edilmesi gerektiğini belirtir.

Daha geniş bilgi için mount programının man dosyasına göz atmanızı tavsiye ederim.

8.7 Uzaktan Bağlantı Kurulması

İnternet üzerinde kesinlikle yanlız olmadığınıza göre, kimi durumlarda kendi Linux'unuz dışındaki sistemlere girmek isteyeceksiniz. Bunun için uyarlanmış çeşitli komutlar mevcuttur. Bu komutlar sadece bağlanmak için değil, iki makina arasında birtakım dosya transferi gerçekleştirmek için de kullanılabilir.

Linux (ve UNIX) için ağ uygulamalarında kullanılmak üzere geliştirilen temel olarak iki sistem vardır. Bunlardan **UUCP** (Unix to Unix CoPy), genel olarak iki makina arasında bir kere bağlantı kurup veriyi iletir ve bağlantıyı kapatır. Eski bir protokol olan UUCP, genellikle modemler arası veri iletişiminde işe yarar.

TCP/IP ise daha önce de belirtildiği gibi yerel ağ veya İnternet üzerinde rahatlıkla kullanılabilen bir protokoldür. TCP / IP 'nin uzaktan bağlantı komutları "r" harfi ile başlar. Örneğin `rlogin` (remote login) bir sistemden diğerine telnet bağlantısı yaparken `rcp` (remote copy) ise bir makinadan diğerine dosya kopyalar. Bu ve buna benzer uzaktan bağlantı komutları iyi bilmeniz size çalışmalarınız sırasında zaman kazandıracaktır.

"`r`" komutlarını kullanabilmeniz için kesinlikle TCP/IP protokolünü tanıyan bir ağ (PPP, Ethernet, SLIP gibi) üzerinde olmalısınız.

rlogin : Geçerli bir hesabınızın olduğu makinaya girmenizi sağlar. telnet komutuyla benzerlik gösterir. Ayrıldıkları nokta, karşıdaki sisteme girerken gerekli konfigürasyonlar yapılırsa şifre sormamasıdır. Bu kısma ileride değineceğim.

Aşağıdaki komut ile ordek isimli makinaya giriş yapılıyor.

```
$ rlogin ordek
```

Eğer yukarıdaki gibi yazarsanız, ordek makinasına ait IP numarası veya bu adresin tam isminin, örneğin `ordek.cclub.metu.edu.tr/etc/hosts` dosyasında yeraldığına emin olmanız gerekir. Karşı makinada bağlanılacak kullanıcı ismi de komut satırından belirlenebilir.

```
cclub:~$ rlogin ordek -l httpd
Password:
Last login: Fri Feb 21 22:02:46 from cclub.metu.edu.tr
Linux 2.0.27.
ordek:~$
```

Yukarıda yaptığımız, cclub makinasından ordek isimli makinadaki httpd kullanıcısına bağlanmak oldu. Üzerinde Linux çalışan ordek makinası doğal olarak bana bu kullanıcıya ait şifreyi sordu. ordek makinasında yeralan bir dosya üzerinde değişiklik yaparak istersem şifre sorulmadan anında girilmesini sağlayabilirim. Bunu `.rhosts` (remote hosts) dosyası sağlar.

`.rhosts` dosyasında makinaya bağlanabilecek olan kullanıcılar ve bunların hangi makinalardan bağlanacaklarını gösteren bir liste vardır. Biraz karışık mı oldu ? Şu örneği inceleyelim :

```
ordek:~$ uname -a
```

```
Linux ordek 2.0.27 #1 Sun Jan 19 13:06:08 EET 1997 i486
ordek:~$ whoami
httpd
ordek:~$ cat ~/.rhosts
cclub.metu.edu.tr gorkem
```

Yukarıda ördek makinası üzerindeki httpd kullanıcısının çalışma dizininde bulunan `.rhosts` dosyasında iki sütun yeralıyor. İlki, güvenilen makinanın ismi (cclub.metu.edu.tr), ikincisi de bu makina üzerindeki güvenilen kullanıcı (gorkem). `.rhosts` dosyası şifrenizi başkalarına vermeden istediğiniz makinadaki bir kullanıcının hesabınızı kullanmasını sağlar. Kısaca bir satırda herhangi bir kullanıcının makina ismi ve hesap ismi bu dosyada varsa, sisteme giriş izni verilir.

Bu aşamadan sonra cclub.metu.edu.tr adresindeki gorkem kullanıcısı şifre girmeden ordek.cclub.metu.edu.tr adresindeki httpd kullanıcısı olarak çalışmak isterse yapması gereken :

```
cclub:~$ whoami
gorkem
cclub:~$ rlogin ordek.cclub.metu.edu.tr -l httpd
Last login: Sun Feb 23 11:19:04 from cclub.metu.edu.tr
Linux 2.0.27.
ordek:~$
```

Artık ordek makinası üzerindeyiz. Şifre girmeden bir makinadan diğerine atlamış olduk. `.rhosts` dosyasında eğer makina ismi yerine "+" işareti kullanılırsa her makinanın veya kullanıcı ismi yerine "+" işareti kullanılırsa her kullanıcının giriş izni olduğu anlaşılır. Makina isimleri yerine IP adreslerini de kullanabilirsiniz. Birkaç örnek :

```
144.122.71.55  +
# (144.122.71.55 makinasi uzerindeki tum kullanicilara giris izni ver)
+ murat
# Butun makinalar uzerindeki murat kullanicilarina giris izni ver.
# (Cok buyuk bir guvenlik acigi)
```

rsh : Bazı durumlarda çalıştığınız makinada dışındaki bir sistemde bir veya birkaç kısa komut çalıştırmak isteyebilirsiniz. Örneğin karşı sistemde kimler, hangi işleri yapıyor veya çalışma dizininde yeralan dosyaları görmek gibi. Bütün bunlar telnet veya rlogin programına gerek duymadan `rsh` (remote shell) ile yapılabilir. Tek şart, üzerinde komut işletilecek makinadaki `.rhosts` dosyasında gerekli makina ve kullanıcı ismi girişi olmasıdır.

Diyelim ki **davul.cclub.metu.edu.tr** makinası üzerindeki `/home/gorkem` dizininde yeralan tüm dizin ve dosyaları görmek için :

```
$ rsh davul.cclub.metu.edu.tr ls -al /home/gorkem
```

komutunu girmelisiniz. `rsh` komutundan sonra uzaktaki sistemin ismi, ardından da çalıştırılacak olan komut gelmelidir. Diğer bir örnek

```
$ rsh davul.cclub.metu.edu.tr cat README | lpr
```

karşı sisteme giriş yapmadan README dosyasını yazıcıdan çıkmasını sağlayacaktır.

8.8 Linux'ta Örnek Ağ Uygulamaları

Aşağıda, Linux ile gerçekleştirilebilecek birkaç örnek ağ uygulamaları yeralmaktadır. Her konu ile ilgili ayrıntılı bilgileri *HOWTO* dosyalarından öğrenebilirsiniz.

8.8.1 IP Accounting

IP accounting özelliği, çekirdeğin ağ üzerindeki trafiğin analizini yapmasını sağlar. Toplanan veriler, makinanın açıldığı andan itibaren kart üzerinden geçen paket ve toplam bayt sayısıdır. Bir konfigürasyon dosyası yardımıyla bu bilgiler belirli sınıflara ayrılabilir. Çekirdeğin derlenmesi aşamasında IP accounting seçeneği kullanılmalıdır.

```
* Networking options
...
IP: accounting (CONFIG_IP_ACCT) [N/y/?] y
```

Çekirdeğin derlenmesinden ve makinanın yeni çekirdekle açılmasından sonra, ipfwadm komutu ile IP accounting işlemlerine başlanabilir. ipfwadm paketi standart Slackware dağıtımı ile beraber gelmektedir.

8.8.2 IP Aliasing

Tek ağ arabirimi (örneğin ethernet veya seri bağlantı) kartı için birden fazla IP numarası gereken durumlarda IP Aliasing kullanılır. Genellikle İnternet Servis Sağlayıcıları, müşterilerinin ftp ve www sunucularını tek makina üzerinde toplamak için bu yola başvururlar. Bu sayede bir makinaya atanmış birden fazla www adresi ve herbirinin ayrı html dosyaları farklı makinalardaymış gibi kullanılabilir.

Bu özellikten yararlanabilmek için çekirdekte IP Aliasing desteği vermelisiniz.

```
*
* Networking options
...
Network aliasing (CONFIG_NET_ALIAS) [N/y/?] y
...
IP: aliasing support (CONFIG_IP_ALIAS) [N/y/m/?] y
```

Her sanal ağ adresi için bir *arabirim* (interface) tanımlamalısınız. Bu sayede aynı bilgisayarda 255 adede kadar sanal makinaya izin verilebilir. Her sanal arabirim, bağlı bulunduğu gerçek arabirim üzerinden işgörür.

Örnek olarak, elimizde kullanabileceğimiz ikinci adres (örnek olarak 100.100.100.101) ve tek ethernet kartı bulunsun. İnternet üzerindeki diğer makinaların, bu iki adrese ulaşabilmeleri için şu satırlar yeterli olacaktır :

```
# ifconfig eth0:0 100.100.100.101 netmask 255.255.255.0 up
# route add -net 100.100.100.0 netmask 255.255.255.0 eth0:0
```

Sanal bir arabirimi kaldırmak için ise, ismin sonuna '-' karakteri getirmek yeterli olur.

```
# ifconfig eth0:0- 0
```

8.8.3 IP Masquerading

IP Masquerading, bir ağın arkasındaki birden fazla makinanın bir Linux üzerinden dışarıya bağlanmasıdır. Tüm makinalar dışarıdan aynı IP adresi gibi görünürler.

Örnek olarak İnternet'e bağlandınız. İnternet Servis Sağlayıcınız size sadece bir IP numarası verdi ve odanızda İnternet'e bağlanmayı bekleyen birden fazla bilgisayar var. Servis Sağlayıcıya her makina için para ödeyip bir hesap ismi almaktansa odadaki tüm makinalar bir Linux ve telefon bağlantısı yardımıyla İnternet'e erişebilirler.

IP Masquerading kullanabilmek için çekirdekte bu desteğin verilmesi gerekiyor. Bunun için IP Forwarding, IP Firewalling ve IP Masquerading seçeneklerini açık tutmalısınız.

IP Masquerading için gerekli yazılımlar Slackware dağıtımı ile birlikte gelmektedir.

8.8.4 SAMBA (NetBEUI ve NetBios desteği)

SMB, Windows 3.11, NT ve 95 tarafından kullanılan ve disk ve yazıcı paylaşımına olanak veren bir protokoldür. Linux, Samba desteği ile bu işletim sistemleri veya kendi üzerinde bulunan disk ve yazıcıların tüm makinalar tarafından paylaşılmasını sağlar.

Samba ile 4 işlem gerçekleştirilebilir : Windows makinalarının Linux diskini paylaşmaları, Linux makinalarının Windows sistemi diskini paylaşmaları, Windows makinalarının Linux yazıcısını paylaşmaları ve Linux makinalarının Windows yazıcısını paylaşmaları.

Yine bu özelliği kullanabilmek için çekirdekte Samba desteği vermeniz gerekecektir.

```
SMB filesystem support (to mount WfW shares etc..) (CONFIG_SMB_FS) [Y/m/n/?]
```

smbfs, SMB protokolünü tanıyan dosya sistemine verilen isimdir. smbfs servisi verebilmek için smbfs.tgz paketini indirip derlemelisiniz.

smbfs.tgz paketini *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/linux/suniste/systems/filesystems/smbfs/>` adresinden bulabilirsiniz.

8.8.5 Novell Desteği

NCP (NetWare Core Protocol), IPX üzerinde yeralan ve Novell NetWare istemcilerinin sunucularla konuşmasını sağlayan protokole verilen isimdir. Bu seçenek ile, NetWare dosya sunucularına diğer Unix makinalardaki gibi bağlanmak olasıdır.

NetWare dosya sistemlerine erişebilmek için özel bir mount programı (`ncpmount/ncpumount`) kullanmanız gerekecektir. Bu ve diğer programlar, *ncpfs* `<ftp://ftp.metu.edu.tr/pub/linux/sunsite/system/filesystems/ncpfs>` paketinin içinde vardır. Bunun için çekirdekte

```
The IPX protocol (CONFIG_IPX) [Y/m/n/?]
```

ve

```
NCP filesystem support (to mount NetWare volumes) (CONFIG_NCP_FS) [Y/m/n/?]
```

desteğini vermelisiniz.

lwared ve *mars_nwe* paketleri, Linux İşletim Sistemi'ne Novell sunucu olma özelliği verir.

8.8.6 Apple desteği

Aynı ağ üzerinde yeralan Apple Macintosh makinaları ile aynı disk ve yazıcıların paylaşımını sağlayan pakete *netatalk* ismi verilir. Bu desteği kullanabilmek için çekirdeğin derlenmesi sırasında

```
IP: multicasting (CONFIG_IP_MULTICAST) [Y/n/?]
...
The IPX protocol (CONFIG_IPX) [Y/m/n/?]
...
Appletalk DDP (CONFIG_ATALK) [Y/m/n/?]
```

parametrelerine evet cevabı vermelisiniz. Appletalk için daha fazla bilgiye ve disk ile yazıcı paylaşımı yapan pakete *http://www.umich.edu/ rsug/netatalk* `<http://www.umich.edu/ rsug/netatalk>` adresinden erişebilirsiniz.

8.9 NIS

NIS Network Information Service kelimelerinin baş harflerinden oluşmaktadır. Amacı bir ağ üzerindeki bütün makinalara, bütün ağ tarafından bilinmek kaydıyla bilgi akışı sağlamaktır. NIS tarafından dağıtılacak bilgiler, hesap isimleri, şifreler, kullanıcı dizinleri ve grup bilgileri olabilir.

NIS yardımıyla, örneğin size ait şifre NIS veri tabanında kayıtlı ise, ağ üzerinde NIS istemci programlarını çalıştıran bütün makinalara girebilirsiniz.

NIS'in biraz daha gelişmiş bir sürümü olan NIS+ , verileri korumak amacıyla şifreleme yöntemi kullanır. Bu avantajına rağmen NIS+ 'ın kurulumu ve kullanımı daha zordur.

8.9.1 NIS'in Çalışma Prensipleri

Bir ağ üzerinde en az bir makina NIS sunucusu olmak zorundadır. İster herbiri farklı NIS "domain" lerine hizmet eden çoklu NIS sunucularına sahip olabilirsiniz, ya da beraber çalışan NIS sunucularınız olur. Bu durumda sunuculardan biri ana NIS sunucusu, diğerleri ise ikincil (slave) NIS sunucusu olabilirler.

İkincil sunucular sadece NIS veritabanının kopyalarına sahiptirler ve bu kopyaları ana NIS sunucusunda bulunan veri tabanlarında değişiklik yapıldığı zaman ana sunucudan alırlar. Ağ üzerinde bulunan makina sayısına ve ağ güvenliğine bağlı olarak bir veya daha fazla ikincil sunucu kurulabilir. Bir NIS sunucusu kapandığında veya çok yavaş cevap verdiğinde, diğer NIS istemcileri çalışan veya daha hızlı bir sunucu bulmaya çalışacaktır.

NIS veritabanları DBM adı verilen yapıdadır ve bu yapı ASCII veritabanlarından çıkarılır. Ana NIS sunucusu hem ASCII veritabanlarını hemde DBM veritabanlarını içermelidir. NIS sunucusu, ASCII yapıdaki `/etc/passwd` gibi veri tabanlarını DBM veri tabanına çevirerek `/var/yp` dizini altında tutar.

İkincil sunucular NIS haritalarında oluşacak değişikliklerden `yppush` programı aracılığıyla haberdar edilecek ve veritabanlarını senkronize etmek için otomatik olarak gereken değişiklikleri alacaklardır. NIS istemcilerinin bunu yapmasına gerek yoktur çünkü sunucudaki DBM veritabanlarında bulunan bilgileri okumak için devamlı olarak NIS sunucusu ile bağlantı kurmaktadır. İkincil sunuculara gayet büyük ağlar üzerinde ihtiyaç duyulur.

8.9.2 NIS İstemcisi

NIS'i kurmak için herşeyden önce `/usr/sbin/rpc.portmap` programını çalıştırmanız gerekir. Bazı Linux dağıtımları (örneğin Slackware) bu daemon'u çalıştırmak için gereken kodu `/etc/rc.d/rc.inet2` dosyasında barındırırlar.

RPC portmapper (`man 8 portmap`) programı bir sunucu olup RPC program numaralarını TCP/IP (veya UDP/IP) protokol port numaralarına çevirir. RPC portmapper o makina üzerinde RPC çağrılarını (NIS istemci yazılımının yaptığı iş) RPC sunucularına (NIS sunucuları gibi) yapılabilmesi için çalışıyor olmalıdır. Bir RPC sunucusu başladığı zaman, portmap'a hangi port numarasını dinlediğini ve hangi RPC program numaralarına hizmet edeceğini bildirir. Bir istemci verilen bir program numarasına RPC çağrısı yapmak istediği zaman, ilk önce sunucu makina üzerindeki portmap ile bağlantı kurarak RPC paketlerinin nereye gönderileceğini belirler.

Normal olarak, bazı RPC sunucuları *inetd(8)* tarafından başlatılır, bu yüzden portmap, inetd'den önce çalıştırılmalıdır.

NIS kurmadan önce, yukarıda da belirtildiği gibi iki durumu değerlendirmek zorundasınız. Ya sizin makinanız halihazırdaki NIS sunucularının bulunduğu bir ağın parçası olacaktır, ya da henüz ağınızda herhangi bir NIS sunucusu yoktur.

Birinci durumda, sadece istemci programlara (`ypbind, ypwhich, ypcat, yppoll, ypmatch`) ihtiyacınız vardır. En önemli program `ypbind`'dır. Bu programı her zaman çalıştırıyor olmalısınız. `ypbind` çalışır çalışmaz, makinanız bir NIS istemcisi durumuna gelir.

İkinci durumda, eğer NIS sunucularınız yoksa, bir de NIS sunucu programına (`ypserv`) ihtiyacınız vardır.

NIS İstemcisi Kurulması

Slackware dağıtımıyla gelen NIS yazılımını daha önce kurmamışsanız, sisteminize yükleyin. NIS istemcisi için `ypbind` dışındaki çalıştırılabilir dosyalar (`ypwhich, yppoll, ypmatch`) bütün kullanıcılar tarafından erişilebilir bir dizinde bulunmalıdır. Öncelikle yapılması gereken `ypbind`'ı test etmektir. Bunun için,

- Domain adınızı belirlediğinizden emin olun. Bunun için,

  ```
  /bin/domainname-yp nis.domain
  /bin/domainname-yp > /etc/defaultdomain
  ```

  komutunu girin. İkinci satırda, sistemde daha sonra kullanılmak üzere `/etc/defaultdomain` dosyasına domain adı yazılır. Yukarıdaki komutta *nis.domain* herhangi bir kelime olabilir.

- Eğer çalışmıyorsa `/usr/sbin/rpc.portmap` programını çalıştırın.

- `/usr/sbin/ypbind` programını çalıştırın.

- `rpcinfo -p localhost` komutunu kullanarak `ypbind` programının `portmapper`'da servislerini kayıt ettirdiğini kontrol edin. `rpcinfo` şöyle bir çıktı üretmelidir:

  ```
  program vers    proto   port
  100000  2       tcp     111     portmapper
  100000  2       udp     111     portmapper
  100007  2       udp     637     ypbind
  100007  2       tcp     639     ypbind
  300019  1       udp     660
  ```

- Ayrıca `rpcinfo -u localhost ypbind` komutunu da çalıştırabilirsiniz. Bu komut ise şöyle bir çıktı verir:

  ```
  program 100007 version 2 ready and waiting
  ```

- En son olarak, makina isim taramaları (host lookup) için `/etc/host.conf` dosyasında lookup sırasına nis bilgisini ekleyin. Bu dosyada yeralan *order* satırını istediğiniz gibi düzenleyebilirsiniz. Daha fazla bilgi için *resolv(8)* man sayfasına bakın.

  ```
  $ cat /etc/host.conf
  order nis, bind, hosts
  multi on
  ```

Bu noktadan sonra `ypcat`, vb gibi NIS istemci programları kullanılabilir. Örneğin, `ypcat passwd` komutu bütün NIS şifre veritabanını gösterir.

Şimdi sistemde bulunan `/etc/rc.d/rc.inet2` dosyasını ypbind programının sistem açılırken çalışması ve sisteminizin bir NIS istemcisi olarak görev yapması için değiştirmek gerekir. Ayrıca, `/etc/rc.d/rc.inet2` dosyasını açın ve `ypbind`'ı çalıştıran ve domain adını belirleyen satırların başındaki # işaretini kaldırın.

```
# domainname

if [ -r /etc/defaultdomain ] ; then
domainname-yp 'cat /etc/defaultdomain'
fi

# Start the ypbind daemon

if [ -f ${NET}/ypbind -a -d /var/yp ]; then
  echo -n " ypbind"
  ${NET}/ypbind
fi
```

NIS'i çalıştırmak için `/etc/passwd` ve `/etc/group` dosyalarının sonuna "+:*:0:0:::" satırını ekleyin. `ypbind`'ı öldürüp yeniden çalıştırın.

NIS Sunucusu Kurulması

NIS aracılığı ile kullanıma açık olacak dosyalardan ihtiyacınız olanların belirlenmesi gerekir. `/var/yp/Makefile/` dosyasında gerekli ekleme/çıkartmaları yapın.

NIS veri tabanını aşağıdaki satırı yazarak oluşturun:

```
$ cd /var/yp; make
```

Bu programın çalıştırılmasından sonra `/var/yp` dizini altında NIS domain adıyla bir dizin yaratılır ve tüm veri tabanları burada tutulur. `portmapper` programının (`rpc.portmap`) çalıştığından emin olduktan sonra `ypserv` sunucusunu çalıştırın. `rpcinfo` komutu ile `ypserv`'in çalışmasını kontrol edin.

```
$ rpcinfo -u localhost ypserv
program 100004 version 2 ready and waiting
```

Artık sunucu çalışmaktadır.

8.9.3 yppasswdd Programı

NIS şifre veritabanının güvenilir olması için "yppasswd daemon" gereklidir. Kullanıcılar `yppasswd` komutu yardımıyla sunucu üzerindeki şifrelerini değiştirebilirler. `yppasswd daemon` (kısaca `yppasswdd`) bu değişikliklerin doğru bir şekilde yapılmasını ve bu şifre değiştirme sürecini kontrol eden sunucudur.

8.9.4 NIS'in Kontrol Edilmesi

NIS, birkaç basit komut ile kontrol edilebilir.

```
$ ypwhich
```

`ypwhich` NIS sunucusunun hangi makina olduğunu söyleyecektir. Bu komut ise

```
$ ypmatch userid passwd
```

(*userid* herhangi bir kullanıcının login adı olmak kaydıyla) size bu kullanıcının NIS şifre dosyasındaki bilgilerini verir. `ypwhich` ve `ypmatch` programları NIS dağıtımı içinde gelirler.

Bir NIS sunucusu kapandığı ve tekrar başladığı zaman `ypbind` aşağıdaki gibi bir mesaj ile başlayabilir:

```
yp_match: clnt_call:RPC: Unable to receive; errno = Connection refused
```

ve NIS veritabanında kayıtlı olanlar sisteme giremezler. root olarak sisteme girmeyi deneyin ve başarılı olursanız `ypbind` işlemini öldürüp yeniden başlatın.

Chapter 9

DNS (Alan İsimlendirme Sistemi)

9.1 Giriş

Bilgisayar ağlarının günümüzde oldugu kadar yaygın olmadığı dönemlerde makina, sunucu, yazıcı vb. ağ üzerindeki cihazlara ilgili nümerik adresleri ile erişmek kullanıcılar için fazla zor değildi. İnternet'in beklenenden daha hızlı büyümesiyle birlikte artan kullanıcı ve ağ üzerindeki cihazlarla birlikte nümerik isimler yerine alfanümerik hatırlanması kolay bir yapı geliştirildi. Bu yapıya Alan İsimlendirme Sistemi (Domain Name System) adı verilmektedir. Genellikle kısaltılmış şekliyle DNS olarak da kullanılmaktadır.

Alan isimleri verilirken alt isimler (subdomain) genelden özele doğru sıra izler. Örneğin, Türkiye Linux Kullanıcıları Grubu olarak ODTÜ'de yer alan kullanıcıların kullanabileceği alan adı,

`metu.linux.org.tr`

şeklindedir. Burda alt isim (.) ile birbirinden ayrılmıştır. En alt seviye olan 'metu' ODTÜ'de yer alan Linux kullanıcılarını, 'linux' Linux grubunu, 'org' Linux grubunun ticari kaygılar gütmeyen bir organizasyon olduğunu, 'tr' ise ISO tarafından belirlenen Türkiye'nin ülke kodudur.

En üst düzeydeki bazı alan isimleri asağıda listelenmiştir.

1. .com ticari kuruluşlar (commercial)
2. .edu üniversite düzeyindeki eğitim kurumları (educational)
3. .k12 lise düzeyindeki eğitim kurumları
4. .org ticari olmayan organizasyonlar
5. .mil askeri kuruluşlar (military)
6. .net ağ organizayonları (network)
7. .ülke kodu ISO tarafından belirlenen ülke kodu

9.2 TCP/IP uygulamaları ve DNS

Makinalara ait isimlerin nümerik adresleri yerine alfanümerik isimlerin kullanılmasıyla birlikte DNS standart bir hale gelmiş olup İnternet üzerindeki çoğu uygulamalarda doğrudan alan isimleri kullanılmaktadır. TCP/IP uygulamalarında ise alan isimleri günlük kullanımda IP (32-bitlik) adreslerinin yerini almış durumdadır. Örneğin, `metu.linux.org.tr`'a telnet bağlantısı yapabilmek için hem,

```
$ telnet metu.linux.org.tr
```

komutu

```
$ telnet 144.122.199.199
```

komutu yardımıyla ulaşabilirsiniz. Fakat herhangi bir şekilde ODTÜ'de yeralan `metu.linux.org.tr` sunucusunun IP adresinde değişiklik olursa yukarıdaki IP adresini kullanarak erişemezsiniz. Oysa IP adresi yerine `metu.linux.org.tr` sunucusuna bağlanırken alan adını kullanırsanız bu ve benzeri değişiklikler sizi etkilemez.

Şimdi kabaca herhangi bir kullanıcının `metu.linux.org.tr` adresine bağlanmak istediğinde gerçeklesen DNS sorgusunun nasıl işlediğini ve Türkiye'deki yapıyı da anlatacak şekilde örnekleyelim.

Örneğin, Amerika'da herhangi bir İnternet Servis Sağlayıcısı'ndan (İSS) bağlantı yapan bir kullanıcı `http://www.linux.org.tr` adresine bağlanmak istedi. Kullanıcının bu adresi tarayıcısına girmesinden itibaren öncelikli olarak kendi alan adı sunucusuna (name server) bu adrese karşılık gelen IP adresini bulmak için gerekli sorguyu yollayacaktır. Kendi alan adı sunucusu en üst düzeyde yer alan `.tr`'ı gördüğü andan itibaren bu sorguya kendinin yanıt veremeyeceğini anlayıp bir üst alan adı sunucusuna ilgili sorguyu yollacaktır. Türkiye'nin alan adı sunucusunun `knidos.cc.metu.edu.tr` olduğu bilgisini aldığı andan itibaren sorguyu bu kez `knidos.cc.metu.edu.tr` sunucusuna gönderecek ve buradan da `linux.org.tr` alanına ait bilgilerin `metu.linux.org.tr` sunucusunda tutulduğunu öğrenecek. Nihayet son sorgu `metu.linux.org.tr` sunucusuna yollanacak ve buradan da `www.linux.org.tr` adresine karşılık gelen IP adresini öğrenecek.

9.3 BIND Konfigürasyonu

BIND, Linux dağıtımlarının hepsi ile gelen ve alan adı sorgularına yanıt veren yazılım paketidir. Açılımı Berkley Internet Domain Name şeklindedir.

Doğrudan Linux kurulu olan bir sunucu üzerindeki BIND konfigürasyonuna geçmeden DNS ile ilgili temel birkaç konudan bahsetmekte fayda var. DNS ile ilgili yazılımları *istemci* ve *sunucu* olarak ikiye ayırmak daha doğru olur:

Sunucu, istemciden gelen sorgulara yanıt veren ve sürekli çalışan bir işlem olup BIND paketinden çıkan `named` yazılımı bu işi görür. İstemcilerse Linux dağıtımlarıyla gelen `host`, `nslookup` vb.

programlar olup sorguyu yapan yazılımlardır. Bununla birlikte BIND konfigürasyonunda sıklıkla karşımıza çıkacak olan birkaç kavram aşağıda açıklanmaktadır. Yeri geldikçe diğer kavramlara da değinilecektir.

primary name server

O alana ait bütün bilgilerin bulunacağı sunucu olup bu amaçla konfigüre edilmiş sunucudur. "primary name server" bir alan için ancak bir tane olabilir. "primary name server" tanımlarken, `named.hosts`, `named.boot`, `named.rev` ve `named.cache` gibi dosyalara ihtiyaç vardır.

secondary name server

O alana ait veritabanındaki bilgileri "primary name server" dan belirli aralıklarla alan sunucudur. Belirli aralıklarla transfer edilen bu dosyalara zone-file adı verilir. "primary name server" dan alınabilecek yanıtların aynısı "secondary name server" dan da doğru şekilde alınabilir.

caching only name server

Bu şekilde konfigüre edilen DNS sunucuları yapılan sorguları "cache" de tutup bir sonraki sorgu için bir önceki aynı sorguya alınan yanıt kullanılır. Bir çok DNS sunucusu bu şeklide konfigüre ediliyor olmasına rağmen "caching only" sunucular her zaman sorgulara doğru yanıt vermeyebilir.

BIND paketi bütün Linux dağıtımları ile gelir. BIND'ın yeni versiyonuna ihtiyaç duyduğunuzda *ftp.metu.edu.tr* `<ftp.metu.edu.tr/sunsite/system/daemons/dns/>` adresinden paketi transfer edebilirsiniz. Gerekli kurulumları yaptığınızda `/usr/sbin` dizini altına `named` binary dosyasını koyacaktır.

`named` konfigürasyonu ile ilgili dosyalar ve işlevleri aşağıda yeralıyor.

named.boot

Genel `named` parametrelerini ve o alana ait veritabanı bilgilerinin nerede olacağı belirtilir.

named.cache

İnternet üzerindeki root domain sunucularının listesini içerir.

named.local

loopback adresine ait tanımları içerir.

named.hosts

O domain ait makina isimlerine karşılık gelen IP numaralarının bulunduğu dosya.

named.reverse

IP adresine karşılık gelen isimlerin listesinin tutulduğu dosya.

Yukarıda belirtilen dosya isimlerinden named.boot dışındaki dosya isimleri genelde karşılaşılan isimler olduğu için kullanılmıştır. Bu dosya isimlerini istediğiniz gibi değiştirebilirsiniz.

“named” konfigürasyonuyla ilgili kullanılan örnek dosyalar `linux.org.tr` alanının alan adı sunuculuğunu yapan `metu.linux.org.tr` makinasından alınmıştır (`linux.org.tr` ve `metu.linux.org.tr` aynı sunuculardır).

9.3.1 named.boot Dosyası

`named.boot` dosyası `named` programı çalıştığı zaman baktığı ilk konfigürasyon dosyasıdır ve `/etc` dizini altındadır. `named`, bu konfigürasyon dosyası ile diğer dosyalara veya DNS tanımlarının tutulduğu diğer dosyalara nasıl ulaşacağını öğrenir. Aşağıda `linux.org.tr` sunucusuna ait `named.boot` dosyasını göreceksiniz.

```
;
;    boot file for name server
;
directory /etc/namedb

;type           domain                source host/file          backup file

cache           .                                               root.cache
primary         linux.org.tr                                    named.hosts
primary         0.0.127.IN-ADDR.ARPA                            named.local
forwarders      144.122.199.20
```

`named.boot` dosyasında yer alan satırlara kısaca göz atacak olursak,

directory

`named.boot` dosyasında belirtilen diğer dosyaları hangi dizin altında `named` programının arayacağını belirtir.

cache

cache dosyasını belirtir.

primary

primary’den hemen sonra gelen alanın primary name server olduğunu belirtir ve ilgili bilgileri de alan adından hemen sonra verilen dosya içerisinde tutar. Bu satır birden fazla olabilir, bu durumda ise konfigüre edilen sunucu birden fazla alan için primary name server olarak görev yapar.

forwarders

Sorgularına yanıt veremediği istekleri yönlendireceği sunucuların listelendiği satır.

slave

Yukarıdaki konfigürasyon dosyasında olmayan fakat başka bir yerde karşınıza çıkabilecek olan slave satırı eklendiği zaman name server sorgulara yanıt vermek için sadece ve sadece forwarders satırında belirtilen sunucuları kullanabilir.

9.3.2 Caching Only Alan Adı Sunucu Konfigürasyonu

`linux.org.tr` sunucusu üzerindeki alan adı sunucu sadece *caching only* olarak çalışacak olsaydı, yukarıda belirtilen konfigürasyon satırı aşağıdaki hale gelirdi.

```
named.boot
-----------
;
;    boot file for name server
;
;type           domain                source host/file          backup file

cache           .                                               root.cache
primary         0.0.127.IN-ADDR.ARPA                            named.local
```

Bu konfigürasyonla çalışan alan adı sunucusu, `named.cache` dosyasında yer alan root sunucuları kullanarak gelen sorgulara yanıt verip, sorguların yanıtlarını cache'te tutmaktadır.

9.3.3 Primary ve Secondary Alan Adı Sunucu Konfigürasyonu

Yine `linux.org.tr` `named.boot` dosyasına bakacak olursak,

```
 primary         linux.org.tr            named.hosts
```

satırıyla `linux.org.tr` makinası `linux.org.tr` alanı için primary sunucu ve bu domain ile ilgili bilgilerin tutulduğu dosya ise `named.hosts` dosyasıdır.

Yukarıdaki satırı named.boot dosyasına yazmamızla aslında işimiz bitmiyor.

`linux.org.tr` sunucusu `linux.org.tr` alanı için primary olduğunu, Türkiye için root sunucu olan `knidos.cc.metu.edu.tr` sunucusunun da bilmesi gerekiyor. Bunun için de `http://www.metu.edu.tr/~dnsadmin` adresinden gerekli form doldurup tanımların ODTÜ'deki yöneticiler tarafından yapılması gerekiyor. Bu form doğrultusunda da `knidos.cc.metu.edu.tr` sunucusunda aşağıdaki satırların girilmesi şarttır.

```
linux.org.tr.           IN      NS      metu.linux.org.tr.
metu.linux.org.tr.      IN      A       144.122.199.199
```

Şu an bu işlemler zaten yapılmış olup sadece bilginiz olması için bu satırlar yazılmıştır.

```
primary          0.0.127.IN-ADDR.ARPA    named.local
```

satırıyla `linux.org.tr` makinasında 127.0.0.1 IP adresine karşılık gelen *loopback* isminin tanımlarının yer alacağı dosya adı verilmiştir.

Eğer Türkiye Linux Kullanıcıları Grubunun kendine ait bir IP adres bloğu, örneğin şu an ODTÜ'de kullanılan 144.122.199.0 bloğu verilmiş ve bu blok kendi alt gruplarına dağıtılmış olsaydı bu adreslere ait reverse tanımlar yine `linux.org.tr` makinasında tutulacaktı. Böyle bir durumda `named.boot` dosyasına aşağıdaki satır benzeri bir satır daha eklemek gerekecekti.

```
primary          199.122.144.IN-ADDR.ARPA    named.144.122.199
```

Türkiye Linux Kullanıcıları Grubu için kullanılan IP adresine ait gerekli reverse tanımı ilgili üniversitenin alan adı sunucusunda girilmiş durumdadır. Örneğin, `metu.linux.org.tr` için `knidos.cc.metu.edu.tr` sunucusunda aşağıdaki satır tanımlanmıştır.

```
199.199.122.144         IN       PTR      metu.linux.org.tr.
```

Bazı programların reverse tanımlarını kullanıyor olması nedeniyle (örneğin tcpd) bu tür tanımların girilmesi önemlidir. Bir veya daha fazla C sınıfı şeklinde IP adres blokları aldığınız zaman IP adresi aldığınız kurumdan, bu adres aralığının reverse tanımlarını sizin alan adı sunucunuza yönlendirilmesini isteyebilirsiniz. Bir C sınıfından az sayıdaki IP adresleri içinse yine IP adresini aldığınız kurum gerekli reverse tanımları sizin için yapmak zorundadır.

Şu an `linux.org.tr` makinası herhangi bir başka primary sunucunun secondary sunucusu olmadığı için `named.boot` dosyasında secondary geçen bir satıra rastlamadık. Örneğin `linux.org.tr` sunucusu `linux.org` sunucusunun secondary alan adı sunucusu olduğunu varsayalım, bu durumda aşağıdaki satırı `named.boot` dosyasına eklemememiz gerekecek.

```
secondary          linux.org      198.182.196.49     SEC.linux.org
```

Bu satırı eklediğimiz andan itibaren *named* belirli aralıklarla gidip linux.org sunucusundan DNS dosyalarının bir kopyasını linux.org.tr sunucusuna aktaracak. linux.org alanının secondary sunucusu oldugumuza dair bilginin de ayrıca linux.org alanının primary tanımlarını tutan sunucuda belirtilmesi gerekir.

9.3.4 named Konfigürasyon Dosyası

named konfigürasyon dosyalarında (`named.hosts, named.local, named.cache, named.rev`) karşılaşılan konfigürasyon parametreleri ve ne anlama geldikleri aşağıda yeralmaktadır.

SOA (Start of authority)

İlgili dosyanın ne kadar zaman aralıklarında yenilenebileceği gibi bilgileri içerir.

NS (Name Server)

Alan adı sunucusu olduğunu belirtir.

A(Adress)

Makina ismini IP adresine çevirir.

PTR (Pointer)

IP adresini makina ismine çevirir.

MX (Mail Exchange)

O alana ait mesajları hangi makinanın dağıtacağını belirtir.

CNAME(Canonical Name)

Bir makinaya verilen isimler birden fazla ise bu isimleri belirtirken kullanılır.

HINFO (Host Information)

Belirtilen sunucunun işletim sistemi veya donanımı konusunda bilgi verir.

`named.hosts, named.local` vb. dosyalara girilen standart bir DNS kaydı aşağıdaki yapıda karşımıza çıkar.

```
[name] [ttl] IN type data
```

[ ] içerisinde belirtilen değerler isteğe bağlı olup gerekmediği durumlarda yazılmayabilir.

name

Bu kısma tam domain ismi, makina ismi yazılabilir.

ttl (time-to-live)

Burada belirtilen süre kadar (bu süre saniye cinsindendir) cache tutulabilecegini belirtir. Eğer herhangi bir süre verilmemiş ise SOA kısmında verilen süre kullanılır.

IN

Girilen kaydın DNS kaydı olduğunu belirtir.

type

Yukarıda bahsettiğimiz kısaltmalardan ilgili olanı yazılır.

data

Bu kısma IP adresi yazılır.

Şimdi de bu kısaltmaların bulunduğu `named.cache, named.hosts, named.local` gibi dosyalardan birer kesit alıp içeriklerine göz atalım.

```
metu:/etc/namedb# more named.cache
;       last update:    Nov 8, 1995
;       related version of root zone:   1995110800
;
;
; formerly NS.INTERNIC.NET
;
.                       3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
;
; formerly NS1.ISI.EDU
;
.                       3600000      NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.     3600000      A     128.9.0.107
;
;
```

Yukarıdaki dosya, root name server'ların listesinin yer aldığı dosyadır. Süregelen değişiklikler doğrultusunda bu dosyada yer alan kayıtlar da değişebilirler. Sürekli güncel tutmak amacıyla `ftp://nic.ddn.mil/netinfo/root-servers.txt` adresinden dosyanın yeni halini belirli aralıklarla almakta fayda vardır.

ttl süresi ise olabildiğince büyük verilerek root sunucuların herhangi bir şekilde cache'ten silinmesi önlenmiş olur.

```
metu:/etc/namedb# more named.local
;
;       @(#)named.local 1.1     (Berkeley)      86/01/21
;
$ORIGIN 0.0.127.in-addr.arpa.
@       IN      SOA     metu.linux.org.tr. root.metu.linux.org.tr.  (
                                1996012201 ; Serial
                                3600    ; Refresh
                                300     ; Retry
                                3600000 ; Expire
                                14400 )  ; Minimum
        IN      NS      metu.linux.org.tr.
1       IN      PTR     localhost.
```

`IN SOA` ile başlayan satırda `metu.linux.org.tr` adresi ilgili domain konusunda yetkili sunucuyu, `root.metu.linux.org.tr` ise herhangi bir şekilde sorun olduğunda bağlantı kurulacak e-posta adresini (*root@metu.linux.org.tr*) belirtir.

Diğer bilgilerse sırayla `metu.linux.org.tr`'ın "primary name server" olduğunu ve 127.0.0.1 IP adresine localhost isminin karşılık geldiğini belirtir.

```
metu:/etc/namedb# more named.hosts
; Authoritative data for Berkeley.EDU (ORIGIN assumed Berkeley.EDU)
;
$ORIGIN linux.org.tr.
@               IN      SOA     metu.linux.org.tr. root.metu.linux.org.tr. (
                                1997072902      ; Serial
                                10800   ; Refresh 3 hours
                                3600    ; Retry   1 hour
                                3600000 ; Expire  1000 hours
                                86400 ) ; Minimum 24 hours
                IN      MX          10 metu
                IN      NS          metu
                IN      HINFO       "i486" "Linux 2.0.27"
localhost.      IN      A           127.1
metu            IN      A           144.122.199.199
                IN      A           144.122.199.199
www             IN      CNAME       metu
ftp             IN      CNAME       metu
gopher          IN      CNAME       metu
listproc        IN      CNAME       metu
ege             IN      A           155.223.97.31
www.ege         IN      CNAME       ege
```

burada diğer `named.local` dosyasından farklı olarak makina ismi - IP adresi eşlemesi yapılmış olup `MX` ve `CNAME` kavramları kullanılmıştır.

MX tanımın bulunduğu satırda `linux.org.tr` domaini için e-postaları alabilecek sunucu `metu.linux.org.tr` olarak tanımlanmış. 10 rakamı ise `metu.linux.org.tr`'nin `linux.org.tr` olarak gelmesi gereken mesajlardaki önceliğini belirtir. Sayı ne kadar küçük verilirse önceliği o kadar fazladır.

HINFO tanımının bulunduğu satır `metu.linux.org.tr`'ın donanım, işletim sistemi ve sürüm bilgilerini tutuyor.

CNAME geçen satırlar ise `metu.linux.org.tr` makinasının olması düşünülen diğer isimlerini tanımlamak için kullanıldı. Örneğin, `www.linux.org.tr` ile `metu.linux.org.tr` aynı sunucular olup `www.linux.org.tr`, `metu.linux.org.tr`'ın bir diğer adı (alias'ı) olarak gözükmektedir.

Bütün bu konfigürasyon işlemleri bittikten sonra,

```
metu:/etc/namedb# named
```

komutunu çalıştırıp `nslookup` veya `host` gibi programlar yardımıyla name server'ımızın çalışıp çalışmadığını denemek kalıyor. Çalıştırdığımızda verilen hata mesajları `/usr/adm/mesages` dosyasına yazılır.

9.4 host ve nslookup Programlarının Kullanımı

`host` veya `nslookup` programları yardımıyla *bind* konfigürasyonumuzun ilgili sunucuda doğru çalışıp çalışmadığını test edelim.

Örnek olarak, `host` komutunu kullanarak `linux.org.tr` için `knidos.cc.metu.edu.tr` 'da girilmesi gereken tanımların doğruluğunu kontrol edebiliriz.

```
$ nslookup
Default Server:  knidos.cc.metu.edu.tr
Address:  144.122.199.20

> linux.org.tr
Server:  knidos.cc.metu.edu.tr
Address:  144.122.199.20

Name:    linux.org.tr
Address:  144.122.199.199

> exit
```

Yukarıdaki satırlarda `nslookup` yardımıyla öntanımlı DNS sunucusu olan `knidos` makinasına linux.org.tr makinasının IP adresini sorduk. `nslookup` komutunun -t seçeneği yardımıyla bir makina hakkında daha başka veriler almak da mümkündür.

```
> server metu.linux.org.tr
Default Server:  metu.linux.org.tr
Address:  144.122.199.199

> ls -t HINFO linux.org.tr
[metu.linux.org.tr]
 linux.org.tr.                  Linux       2.0.27
>
```

`nslookup` üzerine daha fazla bilgi için komuta ait man sayfasına gözatın.

`host` komutu da `nslookup` komutunun bir tamamlayıcısı gibidir. Aşağıda birkaç örnekle `host` komutu anlatılıyor.

```
$ host linux.org.tr
linux.org.tr has address 144.122.199.199
linux.org.tr mail is handled (pri=10) by metu.linux.org.tr
$ host www.ege.linux.org.tr
www.ege.linux.org.tr is a nickname for ege.linux.org.tr
ege.linux.org.tr has address 155.223.97.31
```

Yukarıda yeralan ilk komut yardımıyla `linux.org.tr` makinasının isim-IP dönüşümü yapılmış ve makinaya karşılık gelen adresin 144.122.199.199 olduğu anlaşılmıştır.

İkinci `host` komutuyla da benzer şekilde `www.ege.linux.org.tr` adresinin IP numarasını öğrendik. Ayrıca aynı IP numarasına karşılık gelen ve linux.org.tr domain'ine ait olan iki isim bulunduğu da yukarıdan kolayca anlaşılıyor.

Chapter 10

Diğer Kullanıcılarla İletişim

İnternet, başdöndürücü bir hızla gelişmekte. Dünyada İnternet'e her gün onbinlerce kullanıcı eklenirken İnternet'e bağlı Türk insanının sayısı (şimdilik) yüzbinlerle ifade ediliyor. Bunda özellikle medyanın, İnternet'in tanıtımına yaptığı etki çok büyük. Konuyla ilgili hemen her yazıda, bu dev ağın diğer ucundaki kullanıcılardan bahsediliyor. İletişim teknolojisiyle birlikte karşılıklı konuşmayı iyice azaltan kesimlerin İnternet'e olan merakı da buradan geliyor.

Gerçekten de İnternet'e bağlandığınızda hiç de yalnız olmadığınızı farkediyorsunuz. Biraz merakla, biraz uğraşla dünyanın öteki ucundan bağlanan insancıklarla bağlantı kurmak gayet kolay. İnternet'le bilgisini, kültürünü artıran, yalnızlıktan kurtulan, hoşça vakit geçiren, hatta evlenenlere bile rastlanıyor. Bazen de bunun tam tersine, hayal kırıklığına uğrayanlar da var. Ama çoğunluk bu sonsuz derinliğe bir girdiği zaman sihrinden kurtulamıyor.

Diğer kullanıcılarla iletişim kurabilmek için sayısız yol yöntem var. Bunlardan en çok bilineni *e-posta*. Mektupla varışı günlerce sürebilecek bir mesaj, e-posta ile gönderdiğinizde bakmışsınız ki birkaç dakika sonra karşıda.

Bunun yanında haber gruplarına bağlanmak ve onbinlerce *tartışma grubundan* hangisini okuyacağını şaşırmak da var.

irc (internet relay chat) ile binlerce odalı bir eve girip her ülkeden, her zevkte süresiz arkadaşlıklar kurmak mümkün.

Bunların hepsi Linux ile mümkün. Yazılımları kurduğunuz anda yüzlerce ülkeden milyonlarca kullanıcı sizi ekran karşısında bekliyor. Aşağıda iletişim kurabilmek için kullanan yazılımlardan birkaçı anlatılıyor.

10.1 Pine

Pine, e-posta atma ve okuma için en çok kullanılan yardımcı programlardan biridir. Kurulumu ve kullanılması kolay olan pine, Linux altında sürekli güncellenen bir yazılımdır.

Daha önce pine kurmadıysanız size en yakın sunsite arşivinden indirip kurmak mümkündür. Pine, açıldığı zaman aşağıdaki gibi bir görüntüyle sizi karşılar :

```
PINE 3.95   MAIN MENU                                Folder: INBOX  4 Messages

       ?     HELP                 -  Get help using Pine
       C     COMPOSE MESSAGE      -  Compose and send a message
       I     FOLDER INDEX         -  View messages in current folder
       L     FOLDER LIST          -  Select a folder to view
       A     ADDRESS BOOK         -  Update address book
       S     SETUP                -  Configure or update Pine
       Q     QUIT                 -  Exit the Pine program

Copyright 1989-1994.  PINE is a trademark of the University of Washington.
                    [Folder "INBOX" opened with 4 messages]
? Help                     P PrevCmd                 R RelNotes
O OTHER CMDS L [ListFldrs] N NextCmd                 K KBLock
```

Pine programı, kursör tuşları ile kolayca yönetilebilir. En üst satırda pine hakkında detaylı bilgi alabileceğiniz bölüm vardır. Diğer satırlar sırayla :

- **COMPOSE MESSAGE :** E-posta hazırlayıp göndermek için
- **FOLDER INDEX :** Gelen ve giden e-postaları gruplanabilir. Bu seçenek seçilen grup içindeki e-postaları ekrana getirir.
- **FOLDER LIST :** Posta grubu seçmek için
- **ADDRESS BOOK :** Adres rehberi
- **SETUP :** Pine'ın konfigürasyonu bu seçenekle yapılır.
- **QUIT :** Programdan çıkar.

Pine'ın kullanıcı dostu özelliklerinden birisi, her komutu tek tuş ile belirtebilmesi ve bu tuşların ne işe yaradıklarının ekranın en alt kısmında sürekli görülebilmesidir. Bir komutu hatırlayamazsanız ekranın en alt kısmına bakın.

10.1.1 E-posta Gönderme

Bir e-posta gönderebilmek için karşıdaki kullanıcının e-posta adresini bilmeniz gerekecektir. Ana menüde COMPOSE MESSAGE satırının karşısına gelin ve return tuşuna basın. Bu andan itibaren pine'ın editörüne gireceksiniz.

```
To      :
Cc      :
Attchmnt:
```

```
Subject :
----- Message Text -----
```

To: kısmına mesaj göndermek istediğiniz kişinin e-posta adresini yazın. *Cc: (Carbon Copy)* bölümünde ise, gönderilen e-postanın ikinci kişi tarafından alınması da istiyorsanız o kişinin de e-posta adresini belirtin.

Pine ile sadece mesajları değil, herhangi bir dosyayı da göndermek mümkündür. *Attchmnt* kısmına, dosyanın adını yazın. Pine, dosyayı ev dizininde arayacak ve atılacak e-postanın içine ekleyecektir.

Subject (başlık) kısmında, mesajın içeriğini belli eden birkaç kelime yazın ve return tuşuna basın. Artık istediğinizi yazabilirsiniz.

```
To      : sonmez@cclub.metu.edu.tr
Cc      :
Attchmnt: 1. /home/murat/adresler (2.8 KB) ""
Subject :
----- Message Text -----

Selam,
Istediklerini ekte yolluyorum. Umarim isine yarar.
Gorusuruz.
```

Mesajı yollamak için `Control-X`, vazgeçmek için `Control-C` tuşuna basın. Yukarıda da belirtildiği gibi, `pine`'ın tüm komutları ekranın alt kısmına yazılmıştır.

10.1.2 E-posta Okuma

Pine, açıldığı anda ana menüye gelir. Herhangi bir alt menüden ana menüye geçmek için m tuşuna basın. Buradayken gelen e-postaları okuyabilmek için *FOLDER INDEX*'e geçin.

```
+   1   Sep 15 To: root@linux.lin  (4,394) Register with the Linux counter proje
+   2   Sep 15 Patrick J. Volkerd  (2,671) Welcome to Linux!
```

Burada gelen e-postaların ne zaman, kim tarafından yollandığı, uzunluğu ve başlığı yeralır. Okumak için ok tuşlarıyla bir tanesinin üzerine gelip return tuşuna basın.

10.1.3 E-posta Silme

Gelen e-postaların diskte fazla yer tutmasını önlemek için bazen kontrol etmelisiniz. Özellikle e-posta aracılığıyla fazla miktarda program alış-verişi yapıyorsanız kısa sürede yer kalmadığını görmeniz işten bile değil. Ayrıca her eklenen e-posta, pine'ı açılış sırasında yavaşlatacaktır.

E-postaları silebilmek için, ana menüden FOLDER INDEX'e geçin. Burada silmek istenilen e-postanın üzerine yön tuşları ile gelip D tuşuna basın.

```
+ D 1   Sep 15 To: root@linux.lin  (4,394) Register with the Linux counter project
+   2   Sep 15 Patrick J. Volkerd  (2,671) Welcome to Linux!
```

Satırın başında D karakteri belirecektir. Silmek istediğiniz dosya diskten hemen silinmez. Sistemden çıkarken Pine, bunları silmek isteyip istemediğinizi sorar, sonra siler.

10.2 Etkileşimli İletişim Yöntemleri

Pine, kullanım kolaylığı ile kullanıcıya cazip gelirken, atılan mesajların karşıya anınında gitmemesi, hatta yolda kaybolması olasılığı bile vardır. Buna karşın bir kere atıldığı zaman e-postayı alan kişi bunu istediği zaman okur, yani e-posta atılırken karşıdaki kullanıcının sistemde o an bulunması şart değildir.

`write, talk` ve `irc`, daha pratik ve anında iletişime izin veren programlardır. `write` ile aynı makina üzerindeki kullanıcıyla anında iletişim kurmak mümkün olur. Komut satırında `write` ve konuşmak istediğiniz kullanıcının ismini yazmak yeterlidir.

```
linux:~$ write gorkem
cok aciktim. yemege gidelim
```

Mesaj gönderme bittiği zaman Control-C yardımıyla çıkılabilir. Aynı şekilde mesajı alan kullanıcı da `write` ile cevap verebilir.

`write`, kısa süren mesajlaşmalarda etkili olurken uzun sürecek konuşmalarda genellikle `talk` kullanılır. `talk` yardımıyla mesajlaşan kullanıcıların herbirinin ekranları enine ikiye bölünür. Ekranın bir yarısında gelen mesajlar, öteki yarısında giden mesajlar görünür.

`talk` ve `write` genellikle iki kişi arasında iletişim kurmak için kullanılır. Bunların yanında `irc`, aynı anda binlerce kullanıcının bağlandığı bir dünyadır. Uygun bir yazılımla dünyanın dört bir yanında bulunan sunuculara bağlanmak suretiyle `irc`'ye girilebilir. Çok basit birkaç komutla yüzlerce kanala girip her kanaldaki onlarca kişiyle yazışmak olasıdır. İster özel mesaj gönderirsiniz, ister belirli bir konuda yapılan konferansa katılırsınız.

Linux'unuzda `irc` yazılımı yoksa sunsite arşivinden alıp kolayca kurabilirsiniz.

10.3 Haber Grupları

Haber gruplarını, milyonlarca kişinin e-posta alıp gönderdiği bir yer olarak düşünebiliriz. Dünya üzerinde haber gruplarına ait e-postaları dağıtma işlemini üstlenmiş pek çok omurga vardır. Bunlar, diğer alt haber grubu sunucularına ücreti karşılığında e-postaları gönderirler. Normal bir kullanıcı da bu sunuculara bağlanıp haberleri okur.

İnternet üzerinde bu haber sunucularına bağlanmak için *NNTP* protokolü kullanılır. Alternatif olarak kendi Linux makinanız üzerinde siz de küçük bir haber sunucu yaratabilirsiniz.

Haber gruplarına bağlandığınızda karşınıza bir menü çıkar. Bu menüde konularına göre sınıflandırılmış gruplar vardır. Kullanıcı bu gruplardan birini seçerek daha önce yollanmış e-postaları okuyabilir ve isterse kendisi de bir e-posta gönderebilir. Her grubun bir ismi vardır ve bu isimlerde alt grupları belirtmek amacıyla noktalar kullanılır. Örneğin Linux grubuna ait onlarca haber grubundan *comp.os.linux, comp.os.linux.x* ve *comp.os.linux.setup* örnek verilebilir.

Haber gruplarını okumak için en çok kullanılan iki program `tin` ve `trn`'dir. tin'e girince ekranda üye olunmuş haber gruplarının bir listesi çıkar. Bunlardan istediğiniz bir tanesine return tuşunu kullanarak girebilirsiniz. Ardından ekranda bu gruba gönderilen e-postalar yeralacaktır. Burada yapabilecekleriniz arasında bir mesajı okumak veya cevap yazmak vardır. q tuşuyla `tin`'den çıkabilirsiniz.

Aşağıda tin komutu çalıştırıldığı anda ekrana gelen haber grupları yeralıyor. İstediğiniz haber grubuna s (subscribe) tuşuyla üye olabilir, istemediklerinizi u (unsubscribe) ile silebilirsiniz. Linux'unuzun bağlandığı sunucunun kapasitesine göre teorik olarak onbinlerce haber grubunu okumak mümkündür.

```
           Group Selection (news.metu.edu.tr  18)              h=help

    1        alt.os.linux              Use comp.os.linux.* instead.
    2        comp.os.linux
    3        comp.os.linux.admin
    4        comp.os.linux.advocacy    Benefits of Linux compared to other o
    5        comp.unix.shell           Using and programming the Unix shell.
    6        gnu.bash.bug              Bourne Again SHell bug reports and su
    7        comp.lang.perl
    8        comp.graphics.algorithms  Algorithms used in producing computer
    9        metu.cc.hot-line
   10        alt.2600                  The magazine or the game system.  You
   11        comp.sys.cbm        The commodore computers
   12        alt.2600.codez            No description.
   13        alt.3d                    Use alt.3d.misc instead.
   14        alt.c64                   Use comp.sys.cbm instead.
   15        alt.cd-rom                Discussions of optical storage media.
   16        alt.cookies.yum.yum.yum   Use rec.food.baking instead.
   17        alt.music.ozzy            Blizzard of Osbourne.

     <n>=set current to n, TAB=next unread, /=search pattern, c)atchup,
  g)oto, j=line down, k=line up, h)elp, m)ove, q)uit, r=toggle all/unread,
    s)ubscribe, S)ub pattern, u)nsubscribe, U)nsub pattern, y)ank in/out
```

Ekranın en altında tin komutlarının bir listesi vardır. Bir e-postaya cevap yazmak için o mesajı okurken f tuşuna basın.

Diyelim ki farklı bir haber sunucusuna bağlanmak istiyorsunuz. tin komutu, `NNTPSERVER` kabuk değişkenini okuyarak bu değişkende bulunan sunucuya bağlanır. Buradaki makinanın ismini değiştirmek suretiyle dilediğiniz bir sunucuya bağlanmanız olasıdır.

```
$ export NNTPSERVER=news.metu.edu.tr
$ tin -r
```

Chapter 11

İnternet Araçları

11.1 FTP

FTP (file transfer protocol), bir makinadan diğerine yapılan dosya transferinde en çok kullanılan yollardan biridir. İnternet desteği olan işletim sistemlerinin hepsinde (MS-DOS, Windows, OS/2 ...) ftp, standart program olarak kullanıcıların hizmetine sunulmuş durumdadır. Linux altında da ftp desteği vardır. Ekte İnternet üzerinde yeralan Linux dökümanlarının listesine baktığınız zaman çoğunun ftp adreslerinden temin edilebildiğini göreceksiniz.

FTP, dosyaları alabilmenizle birlikte istediğiniz adrese yollamanıza da izin verir. Eğer İnternet'e bağlıysanız ve Linux kullanıyorsanız `ftp` vazgeçemeyeceğiniz bir program olacakır.

11.1.1 Temel FTP Komutları

FTP, İnternet'e bağlı olan makinalarda anlam kazanır ve gücünü hissettirir. Saniyeler içinde onlarca dosya, yüzlerce kilobayt bilgi transfer edilebilir. FTP'ye başlamak için kullanabileceğiniz iki komut var. Birisi, standart ftp paketinden çıkarn 'ftp' komutu, diğeri de 'ncftp' komutu. İkisinin de avantajları olmasına rağmen 'ncfp' nin kullanıcıya yönelik kullanım kolaylığı vardır. Siz de 'ncftp' kullanın.

Bir FTP adresine bağlanmak için o adresin IP numarasını veya ismini bilmeniz yeterli. Çok geniş bir Linux arşivi olan linux.org.tr adresine bağlanmak için:

```
# ftp linux.org.tr
```

yazın. Ekranda aşağıdaki gibi bir görüntü gelecektir. *Ncftp*, ftp yapmak için gerekli olan kullanıcı kodunu ve şifreyi kendisi gönderir ve otomatik olarak karşıdaki adrese bağlanır. Eğer ftp kullanıyor olsaydınız, kullanıcı kodu için *anonymous* veya *ftp*, şifre için de e-posta adresini girmeniz gerekecekti.

```
linux:~$ ftp linux.org.tr
```

```
Connected to linux.org.tr.
220 metu FTP server (Version wu-2.4(1) Tue Aug 8 15:50:43 CDT 1996) ready.
Name (linux.org.tr:root): ftp
331 Guest login ok, send your complete e-mail address as password.
Password:
230-Welcome, archive user!  This is an experimental FTP server.  If have any
230-unusual problems, please report them via e-mail to root@metu
230-If you do have problems, please try using a dash (-) as the first character
230-of your password -- this will turn off the continuation messages that may
230-be confusing your ftp client.
230-
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Karşıdaki makinaya bağlandık ve önümüze bir komut satırı çıktı. Bu komut satırında kullanılabilecek ftp komutlarının listesini görmek için

```
ftp> help
```

yazın. Bir komut hakkında daha ayrıntılı bilgi elde etmek için ise 'help' komutunun hemen ardından ftp komutunu getirebilirsiniz. Şimdi patika ismini bildiğimiz bir dosyayı kendi hesabımıza çekelim. Öncelikle nerede olduğumuzu görelim. Bunun için Linux komutunun benzeri olan `ls` veya `dir` komutunu kullanın.

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 9
drwxrwxr-x   8 root     wheel        1024 Jan  7  1980 .
drwxrwxr-x   8 root     wheel        1024 Jan  7  1980 ..
drwxrwxr-x   2 root     wheel        1024 Jan  7  1980 bin
drwxrwxr-x   2 root     wheel        1024 Jan  7  1980 etc
drwxrwxr-x   2 root     wheel        1024 Dec  3  1993 incoming
drwxrwxr-x   2 root     wheel        1024 Nov 17  1993 lib
drwxrwxr-x  12 root     wheel        1024 Feb 14 08:08 pub
drwxrwxr-x   3 root     wheel        1024 Jan  7  1980 usr
-rw-r--r--   1 root     root          312 Aug  1  1994 welcome.msg
226 Transfer complete.
```

Bir ftp adresine bağlandıktan sonra genelde her adreste yeralan *pub* , *incoming*, *bin* gibi dizinler göreceksiniz. Bunlardan şu an için işimize yarayanı pub dizini olacaktır. Bir dizine girmek için yine bir UNIX komutu olan 'cd' ve ardından 'pub' yazın.

```
ftp> cd pub
```

Şimdi 'pub' dizini altındayız. Genellikle her türlü program, uygulama yazılımı, ağ uygulamaları ve çeşitli dosyalar bu dizin altında bulunur. Burada hangi dosya ve dizinler olduğunu görelim.

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 12
drwxrwxr-x  12 root     wheel        1024 Feb 14 08:08 .
drwxrwxr-x   8 root     wheel        1024 Jan  7  1980 ..
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 X
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 applications
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 development
drwxr-xr-x   2 root     root         1024 Feb 14 08:08 distributions
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 docs
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 games
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 graphics
drwxr-xr-x   2 root     root         1024 Feb 14 08:09 kernel
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 network
drwxr-xr-x   2 root     root         1024 Feb 14 08:07 sunsite
226 Transfer complete.
ftp>
```

Örneğin 'kernel' dizini altında yeralan bir çekirdek dosyasını kendi makinamıza alalım.

```
ftp> cd kernel
250 CWD command successful.
```

Ftp protokolünde, dosyaların bir makinadan diğerine taşınması esnasında dosyanın çalıştırılabilir (arşivler, sıkıştırılmış dosyalar ve diğerleri) veya metin dosyası olması hayati önem taşır. İnternet üzerinde, zamandan tasarruf amacıyla tüm dosyalar ASCII formatta taşınırlar. Çalıştırılabilir dosyalar 8 bitlik karakterlerden oluştuğu için bunları indirebilmek için taşıma işleminin 8 bit halinde yapılması gerekir. 'ftp' komutunu kullanıyorsanız çalıştırılabilir dosyaları çekmeden önce

```
ftp> binary
```

yazın. Dosyanın türünü bilmediğiniz durumlarda yine 'binary' komutunu kullanın. Siz en iyisi, ftp yaparken karşıdaki makinaya girdikten hemen sonra 'binary' yazın. `ncftp` kullanırken program bunu sizin için otomatik olarak yapacaktır.

Dosyayı indirmeden önce, kontrol edilmesi gereken bir şey de dosyanın indirileceği yerel dizinini belirtmektir. Burada `linux-1.3.56.tar.gz` programını 'kernel' dizininin altına çekelim. Bunun için `lcd` komutunu kullanalım.

```
ftp> lcd /root/kernel
Local directory now /root/kernel
```

Artık istediğimiz dosyayı indirebiliriz. Dosyaları almak için `get` komutu kullanılır.

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 5838
drwxr-xr-x   2 root     root         1024 Feb 14 08:09 .
drwxrwxr-x  12 root     wheel        1024 Feb 14 08:08 ..
-rw-r--r--   1 root     root      5951488 Feb 14 08:10 linux-1.3.56.tar.gz
226 Transfer complete.
ftp> get linux-1.3.56.tar.gz
200 PORT command successful.
150 Opening BINARY mode data connection for linux-1.3.56.tar.gz (5951488 bytes).
226 Transfer complete.
5951488 bytes received in 26 secs (2.2e+02 Kbytes/sec)
```

26 saniye içinde yaklaşık 6 MB büyüklüğündeki dosyayı çektik. Sizin İnternet'e bağlantı hızınıza bağlı olarak dosyayı yerel diske yazma süresinde değişiklikler olabilir. Yukarıdaki işlemleri istediğiniz kadar tekrar edip gereken dosyaları çektikten sonra ftp'den çıkmak için

```
ftp> quit
221 Goodbye.
```

komutunu kullanın. Sunucu makinayla bağlantınız kesilecek ve kabuk komut satırına geri döneceksiniz.

11.2 WWW (World Wide Web)

İnternet'in kimliğini bulmasını sağlayan etmenlerden başlıcası WWW'dir. WWW sayesinde çeşitli yapıda dosyalar, grafik arabirimi kullanılsın veya kullanılmasın kolaylıkla iletilebilir. Linux'u bir web sunucu haline getirmek için tek yapılması gereken, bu dosyaları belirli bir dizin altına yerleştirip WWW sunucusunu başlatmak.

İnternet'te yeralan bilginin hesabı yoktur. Bu bilgi deposunu verimli şekilde kolayca kullanabilir, hatta günlük yaşamınızla birleştirebilirsiniz. Öğrenciler, ev ödevlerini ansiklopediler karıştırmak yerine İnternet'ten bulup çıkarabilirler. Öğretmenler, sınav sorularını İnternet'ten bulabilirler. Üniversitelerdeki bazı fakültele hocaları ödevleri İnternet'te sunuyor ve cevapları da e-posta olarak istiyorlar.

Web üzerinde tarama yapmak amacıyla kurulan ve her hafta milyonlarca ziyaretçisi bulunan bazı adresler vardır. Bu sayfalara bağlanmak suretiyle bir anahtar kelime yardımıyla bu anahtar kelimenin geçtiği WWW adresleri taranabilir. Aşağıda sıkça kullanılan arama-tarama adresleri yeralıyor.

- *altavista* `<http://www.altavista.digital.com>`
- *yahoo* `<http://www.yahoo.com>`
- *lycos* `<http://www.lycos.com>`
- *webcrawler* `<http://www.webcrawler.com>`
- *infoseek* `<http://www.infoseek.com>`

Özellikle birinci sıradaki altavista WWW sunucusu, 40 milyonun üzerinde adres girdisiyle inanılmaz genişlikte bir bilgi bankası olarak göze çarpıyor.

WWW sayfalarına erişebilmek için kullanılan programlar arasında metin temelli olan ve ekranda resim göstermeye imkan vermeyen *lynx* ve X Window altında çalışan ve onlarca çeşit döküman yapısını tanıyabilen *Netscape* sayılabilir.

Web sayfalarına bağlanırken kullanılan protokole HTTP (HyperText Transfer Protocol) adı verilir.

11.2.1 WWW Sunucusunun Konfigürasyonu

Linux'un web sunucusu olarak görev yapabilmesi için httpd daemon'un çalışması gereklidir. Bu yazılım genellikle sistemin açılışı sırasında çalıştırılarak arkaplana atılır.

Diğer kullanıcıların sorumlu olduğunuz Linux üzerindeki dosyalara WWW üzerinden erişmelerini sağlamak için sadece birkaç dakikaya ihtiyaç var. Ftp adreslerinde, özellikle Sunsite arşivinden birkaç değişik sunucu programı alınabilir. Halen Linux için benim bildiğim ve sıkça kullanılan APACHE, NCSA veya CERN http (web) sunuculardan birini çekerek kurmanız gerekecektir. Bununla birlikte, olumsuz yanıt vermemişseniz httpd yazılımı Linux'un kurulumu aşamasında sisteme yerleştirilmiştir.

httpd paketi kurulduktan sonra birkaç alt dizin oluşur. Bunlardan önemlileri,

- **conf**, http konfigürasyon dosyaları
- **cgi-bin**, örnek CGI (Common Gateway Interface) programları
- **cgi-src**, CGI kaynak programları
- **logs**, sunucunun sistem kayıtları

Sunucunun çalıştırılabilmesi için conf dizini altında yeralan 4 dosyanın içinde bazı değişiklikler yapmak gerekiyor. Bunlardan başlıcaları, `httpd.conf` dosyasındaki *ServerRoot* (httpd yazılımının açıldığı yer) ve *ServerName* (makina ismi), `srm.conf` dosyasındaki *DocumentRoot* (ana sayfanın dosyalarının tutulduğu dizin), *ScriptAlias* (CGI programlarının bulunduğu dizin) ve *Directory* (CGI programların bulunduğu dizine ait parametreler) değişkenleridir. Dosyada hiç değişiklik yapmadan da sunucunun istediği dizinler üzerinde gerekli sembolik bağlantıları, istediğiniz dizini işaret edecek şekilde yerleştirmek mümkündür.

httpd programının başlatılması gayet kolaydır. `/etc/rc.d/rc.inet2` programının içinde httpd ile ilgili satırın başında varsa '#' karakterlerini kaldırın. Böyle bir satır yoksa aşağıdaki komutu ekleyebilirsiniz.

```
httpd -f <konfigurasyon-dosyasi>
```

konfigürasyon dosyası yerine `httpd.conf` dosyasının bulunduğu yer girilir. (örneğin `/home/httpd/conf/httpd.conf`)

Web dosyalarını hazırlamak, gerekli düzenlemeleri yapabilmek için ayrı bir kullanıcı tanımlanabilir.

11.3 HTML ve CGI

Bir http protokolünün sunduğu dosya tipi genellikle HTML dosyasıdır. HTML dökümanları tamamı ASCII karakterlerden ve herhangi bir editörde yazılabilen metinlerden oluşur. UNIX üzerinde pico veya vi kullanarak HTML dökümanlarınızı hazırlayabilirsiniz. Bunun dışında, kullanıcının bu dili bilmediği farzedilerek Windows ve DOS altında çalışan çeşitli programlar hazırlanmıştır (FrontPage, HotDog gibi). Ayrıca, HTML kodunun doğruluğunu kontrol eden yardımcı programlara da Internet üzerinden erişilebilir.

HTML, Hypertext tabanlı bir dildir ve program içinde çoklu ortam (multimedia) programlarının konulmasına izin verir. Ses, hareketli veya hareketsiz görüntü yanısıra Java adı verilen ve kullanıcı ile etkileşime girebilen programlar da HTML dosyaları içinde yeralabilir. Grafik destekli Web programlarının tablo desteği ile çok çeşitli istatistiki bilgiler, programlar, her türlü listeler ekranda derli toplu gösterilebilir. Linux için Java desteğini *ftp.blackdown.org* `<ftp://ftp.blackdown.org>` adresindeki *pub/Java/linux* `<ftp://ftp.blackdown.org/pub/Java/linux>` dizini altında bulabilirsiniz.

11.3.1 Temel HTML Belirteçleri

Daha önce bir programlama diliyle uğraşmamış olsanız bile HTML'in kolaylığı ve çok kısa zamanda bu dile hakimiyet sağlayabilmek sizi hayrete düşürecektir. Çeşitli çeviriciler yardımıyla hemen her tür dosyanın bu dile çevrilebilmesi, HTML dilini bilmeyenlerin bile dikkati bu yöne çekmeye yeter. Burada HTML komutlarına kısa bir giriş yapılacaktır, buradaki bilgilerle kendi web sayfanızı hazırlamanız mümkün olacak. Daha geniş bilgi için *www.w3.org* `<http://www.w3.org>` veya *www.yahoo.com* `<http://www.yahoo.com>` adreslerine göz atabilirsiniz. Bu adreslerin yansıları *Linux Kullanıcıları Grubu* `<http://www.linux.org.tr>` ana web sayfasından da erişilebilir.

Basit bir HTML dosyası aşağıdaki gibi yazılabilir:

```
<title> Burasi konunun yazilacagi yer </title>
<h1> Bu, 1 numarali baslik </h1>
```

```
HTML dunyasina hosgeldiniz. <br>
Birinci paragrafimiz. <p>
Bu da ikinci.. <p>
```

HTML yazarken, metnin Web programının anlayacağı şekilde gösterilebilmesi için belirteçler kullanır. Yukarıdaki örnekte:

<`title`> ve <`/title`> belirteçleri, dökümanın konusunu gösterir.

<`h1`> ve <`/h1`> belirteçleri, dökümanın başlığını tanımlar. <p> belirteci, paragraf tanımlar.

Tüm HTML belirteçleri, küçüktür işareti (<), belirteç ismi ve büyüktür işaretinden (>) oluşur. Genellikle her belirtecin <h1> ve </h1> örneklerindeki gibi bir çifti olur ve sondaki belirtecin ismi önüne ayraç gelir. Yukarıdaki örnekte <h1> , Web programına 1 numaralı başlık formatında yazmasını, </h1> ise bu formatı kapatmasını söyler. <p> belirteci istisna bir durumdur ve </p> olarak kapatılmaz.

Her HTML sayfasının ile ayrılan bir konusu olmak zorundadır. Konular sayfadan bağımsız olarak ekranın en üstüne basılır.

```
<title> Sayfamin konusu </title>
```

HTML, 1'den 6'ya kadar numaralanmış 6 çeşit başlık destekler. Başlıklar normal karakterlerden daha büyük ve kalın yazılırlar. Temel olarak,

```
<Hy> Sayfamin basligi </Hy>
```

olarak tanımlanmış bir belirteçte y, 1 ile 6 arası bir değer alabilir. Y sayısı arttıkça yazıtipinin büyüklüğü azalır. Pek çok uygulamada, sayfanın konusu ile başlığını aynı tutabilirsiniz.

Diğer kelime işlemcilerdekinin aksine, HTML dökümanlarında programcı özel bir belirteç kullanmadıkça bir cümle herhangi bir yerinden ayrılıp kalan kısmı altta görünebilir, birden fazla boş satır tek satır olarak algılanır. Aşağıdaki örnekte, kaynak kodunda iki satır olmasına karşın, WWW tarayıcı bunu algılamaz ve ekranda tek satır olarak görüntüler. Bir html tarayıcı satır sonlarını ve boş satırları gözardı edeceğinden, paragrafları mutlaka <p> belirteci ile ayırmalısınız.

```
HTML'ye hosgeldiniz <br>
Bu ilk paragraf <p>
```

Satır sonu belirteci : Paragraf, iki satır arasında bir satır boşluk bırakırken, satır sonu belirteci kullanıldığı kursör alta geçer ve takip eden tüm metin,boşluk bırakmadan bir alttan yazılır.

```
Bu ilk satir. <br>
Ikinci satir daha uzun. <br>
Ama bu bir paragraf sonu.. <p>
```

Paragraflar ortalanırken <center> ve </center> belirteçlerinden yararlanılır. Ortalanması istenen tüm metin, bu iki belirtecin arasına yazılır.

```
<center>
Linux Sayfasi
</center>
```

HTML'nin en büyük özelliklerinden birisi, tek sayfa ile sınırlı kalmamasıdır. Böylece bir sayfadan diğerine bağlantı yapılabilir. HTML'in bu görevini yerine getirmesini sağlayan belirteç <a>'dır. Dökümanınızdan başka dökümana bağlantı yapabilmek için:

- Belirteci girin. (Kısaca ilk satıra <a yazın)
- Hangi dökümana geçiş yapmak istiyorsanız, ismini yazın. (HREF="dosyaismi")
- Bu dökümanı ekranda hangi isimle göstermek istediğinizi belirtin.
- Belirteci kapatın.

Kısa bir örnekle açıklayalım:

```
<a href="internet.html"> Internet nedir? </a>
```

Ekranda "İnternet Nedir?" yazısı belirecek ve kullanıcıdan burayı seçmeyi bekleyecektir. Kullanıcı fare ile bu yazı üzerine tıkladığında ise program kontrolü yine bir HTML dosya olan internet.html dosyasına bırakacaktır. Bu durumda bulunduğunuz dizinden farklı bir dizindeki dosyaya bağlantı yapmak isterseniz, o dosyanın ait olduğu dizini yazmak zorundasınız. Buna göre `linux/shells/bash.html` dosyasını kullanabilmek için

```
<a href="linux/shells/bash.html"> Bash </a>
```

şeklinde bir belirteç yazmak gerekir.

HTML'in bir diğer özelliği ise, tek bir makinaya bağımlı kalmadan diğer İnternet servisleriyle bağlantı kurabilmesidir. Bunun için *URL* (Uniform Resource Locator) kullanılır. URL, http,gopher, news ve telnet gibi alt hizmet birimleri olabilir. Kullanımı şu şekildedir :

```
URL-ismi://makina-ismi[:port]/dizinler/dosya-ismi
```

Burada URL,

- **file**, Kendi sisteminizde bir dosyaya,
- **http**, Bir WWW sunucusu üzerindeki dosyaya,
- **gopher**, Gopher sunucusu üzerindeki dosyaya,
- **news**, Bir UseNet haber grubundaki dosyaya,
- **WAIS**, WAIS sunucusu üzerindeki dosyaya

erişmek için kullanılır. Örneğin sayfanızdan Linux Kullanıcıları Grubu'nun ana web sayfasına ulaşmak için :

```
<a href=http://www.linux.org.tr/> Linux Kullanıcıları Grubu </a>
```

satırını web dosyasında girmek gerekir. Port numarası, genellikle yazılmaz. Size aksi durum belirtilmedikçe, kullanmanıza gerek yoktur.

Dökümanların göze hoş görünmelerini sağlamak amacıyla listeler yaygın olarak kullanılır. HTML, pek çok liste çeşidi destekler. Bunlar, düz listeler, numaralı listeler, tanımlı listeler ve içiçe listelerdir.

Düz liste yaratmak için,

- Listeye başlamak için belirteç açılır. ( )
- Liste elemanlarını teker teker girerken başına ( ) belirteci girilir. Kapatmak için belirtecine gerek yoktur.
- Listeyi bitirmek için belirteç kapatılır. ( )

Örnek olarak,

```
<ul>
<li> Elma
<li> Armut
</ul>
```

Örnek, ekranda şu şekilde görülür :

- Elma
- Armut

 belirteçleri içinde paragraflar, diğer dökümanlara bağlantılar, ve diğer belirteçler kullanabilirsiniz.

Numaralı listeler, düz listelerden farklı olarak, <ul> belirteci yerine <ol> kullanırlar. Ekranda liste elemanlarının başında 1'er artan numaralar görünür. Aşağıdaki HTML kodu,

```
<ol>
<li> Linux
<li> Digerleri :-)
</ol>
```

ekrana şunları yazar:

1. Linux İşletim Sistemi

2. Diğerleri :-)

HTML'de, programda yazıldığı gibi ekrana çıktı vermeyi sağlayan komutlar <pre> ve </pre> belirteç çiftleridir. Bunlar kullanıldığı zaman tüm metin, yazıldığı gibi ekranda görünür. Aşağıdaki satırlar,

```
<pre>
  PATH=.:~/bin/:$PATH
  export PATH
  # Set up the terminal:
         stty erase "^?" kill "^U" intr "^C" eof "^D"
         stty hupcl ixon ixoff
  date '+Tarih :%D'
  TERM=vt100
</pre>
```

ekranda şu şekilde görünür :

```
    PATH=.:~/bin/:$PATH
    export PATH
    # Set up the terminal:
           stty erase "^?" kill "^U" intr "^C" eof "^D"
           stty hupcl ixon ixoff
    date '+Tarih :%D'
    TERM=vt100
```

Eğer Web sayfalarını gezerken grafik destekleyen bir program (Mosaic, Netscape) kullanıyorsanız, ekranda resimlerin, arkaplanların ve hatta animasyonların olduğunu farketmişsinizdir. Bu resimler genellikle X Bitmap (XBM) , GIF, veya JPG formatlı olurlar ve dosyaya görsel bir çekicilik katarlar. Buna rağmen aynı ekranda çok miktarda resim kullanmaktan sakınmalıdır, çünkü bu durumda resimler kullanıcıya daha geç bir sürede ulaşır.

Ekranda resim görüntülemek için,

```
<img src="resmin bulundugu dizin">
```

demeniz yeterlidir. Burada, nasıl HTML dökümanların hepsi `.html` ile bitiyorsa, tüm resim dosyalarının sonu da `.xbm` , `.gif` veya `.jpg` ile bitmelidir.

Şimdi küçük bir örnek hazırlayalım. ~ dizininde iken `public_html` isimli bir dizin açın ve buraya girin. İstediğiniz örnek bir HTML dosyası yaratıp `index.html` isminde kaydedin.

```
$ cd ~
/home/gorkem
$ mkdir public_html
```

```
$ cd public_html
$ ls index.html
index.html
```

Ardından aşağıdaki satırları yazın. Burada web için kullanılan tarayıcı lynx'tir. Eğer X ortamında çalışıyorsanız Netscape de kullanabilirsiniz.

```
$ lynx http://cclub.metu.edu.tr/~ilker
```

Hazırladığınız web sayfası ekrana gelecektir.

11.3.2 Form Hazırlama (CGI)

CGI web üzerinden kullanıcı arayüzleri hazırlamak için kullanılan en yaygın yöntemdir. En yaygın kullanım alanları arasında veri tabanı sorgu sistemleri sayılabilir. Aşağıdaki dökümanda genel olarak CGI tanıtımı yaparken özellikle C ile CGI programları hazırlayacaklara yardımcı olabilir.

Bu dökümanın bazı kısımları okuyucuların az da olsa C ve programlama bilgisinin olduğu varsayılarak hazırlanmıştır.

İnternet üzerindeki hemen hemen bütün kullanıcı arayüzleri, Common Gateway Interface (CGI), kullanılarak hazırlanmıştır. CGI WWW kullanıcılarının WWW sunucusunun çalıştığı makina üzerinde belirlenen programları çalıştırmasını sağlayan bir sistemdir. CGI ile hazırlanan kullanıcı arayüzlerinin başlıca avantajlarını aşağıdaki gibi sıralayabiliriz:

- Kolay ve hızlı hazırlanabilmesi. CGI arayüzlerinde, diğer kullanıcı yüzlerindeki birçok çabaya gerek yoktur. Kullanıcıyla ilişkiyi WWW tarayıcısı (web browser) yaptığından, siz zaten web tarayıcısı için çoktan hazırlanmış karmaşık kullanıcı arayüzü işlemleriyle uğraşmak zorunda kalmazsınız.
- İstediğiniz herhangi bir programlama dili kullanabilirsiniz. Buradaki uygulamamızda C kullandığımız için aşağıda CGI programlarını özel olarak C ile nasıl yazacağınız anlatılmıştır, ama deneyimli programcı herhangi bir programlama dili için gerekli ipuçlarını aşağıdaki açıklamadan çıkarabilir.
- Değişik ortamlarda çalışabilir. Programınızın asıl çalıştığı sistem web sunucusunun çalıştığı sistem olmasına karşın, programınıza bilgisayar ağına bağlı herhangi bir bilgisayardan ulaşılabilir. Kullanabileceğiniz hemen hemen bütün işletim sistemleri ve ortamlar için web tarayıcısı bulabileceğiniz için, yazdığınız tek program birçok farklı ortamdan kullanılabilir.
- Dağıtık ortamlarda kullanılabilme olasılığı. Programınızın kullanıcı arayüzü (web tarayıcısı) başka bir ortamda, web html formlarınız başka bir sistemde, CGI programlarınız başka bir sistemde olabilir.

Bu avantajların yanında sayılabilecek birkaç dezavantaj,

- Dikkatsiz yazılmış CGI programları güvenlik açıklarına neden olabilirler.
- Kullanıcı arayüzleri çok kolay ve hızlı oluşturulmalarına karşın, web tarayıcısının yetenekleriyle sınırlıdır.

CGI Yapısı

En basit haliyle bir CGI programı, standart giriş dosyasına çıktılarını veren herhangi bir program olabilir. Tek farklılık, herhangi bir çıktıdan önce yazmanız gereken iki satır. Bu satırlardan ilki,

```
Content-type: text/html
```

olmalı. İkincisi ise boş bir satır. Aşağıdaki C örneği bu konuda daha açıklayıcı olacaktır. Bundan sonra ekrana yazacağınız herşey web tarayıcısı tarafından HTML ile yazılmış bir döküman olarak değerlendirilecektir.

CGI programları girdilerini HTML formlarından alırlar. HTML formları ayrı bir HTML dosyasında olabilecekleri gibi, bir CGI programının çıktısı olabilirler, hatta aynı program hem formu oluşturup, hem de çıktıyı değerlendirerek gerekli işlemi yapabilir.

Aşağıda örnekleriyle formları nasıl oluşturulabileceğini, ve CGI programlarının nasıl yazılacağı konusunda ayrıntılı bilgi bulacaksınız.

Form *HTML2.0* standardıyla gelen bir kullanıcı arayüzü. Formlarda bir dizi değişik yöntemle kullanıcıdan girdi alınabilir. Bunları aşağıda maddeler halinde göreceğiz.

Bir html formu *<form>* ile başlayıp, *</form>* ile biter. <form> birkaç parametre alır. HTML'de parametreler <form p1=değer1 p2=değer2 p3> şeklinde verilir. Bu örnekte formun p1 ve p2 iki değer alan parametre, p3 adlı değer almayan bir parametresi var. forma verilebilecek ve verilmesi gereken parametreler,

METHOD : Bu parametre, formun ne şekilde hangi html metodu kullanılacağını belirler. POST ya da GET değerini alabilir. POST daha çok karşı tarafta herhangi bir etkisi olan programlar için, GET ise sadece sonuç veren programlar için kullanılır. Ama yaygın işlem her ikisi için de POST kullanmaktır. Biz uygulamamızda her iki durum için de POST kullandık.

ACTION : Bu parametreyle, formun çalıştıracağı CGI programı belirtilir. Verilen CGI programı aynı makinada olabileceği gibi Internet üzerindeki herhangi bir makinada da olabilir.

ENCTYPE : Form'un sonuçlarının CGI programına ne şekilde gönderileceği bu parametreyle belirtilir. Belirtilmemesi durumunda standart 'application/x-www-form-urlencoded' kullanılır. Bizim önerimiz, bu seçeneğin belirtilmemesi ve standart kullanıma gidilmesi. Bu şekilde programınız hem heryerde çalışır, hem de bu kodlama biçimi için hazır kütüphanelerden yararlanabilirsiniz

Formun içinde bir dizi değişik girdi tipi tanımlayabilirsiniz. Bunlar <INPUT> ile verilir. INPUT için bir dizi input tipi kullanılabilir. Bu tipleri aşağıdaki gibi sıralayabiliriz:

TEXT : Bu tip tek satırlık metin girdiler için kullanılır, birden fazla satır için <TEXTAREA> kullanmalısınız.

Yukarıda belirtilen her bir INPUT tipi için verebileceğiniz ya da vermeniz gereken bir ya da birkaç parametre vardır. Bu parametrelerin ayrıntılarına burda yer vermiyoruz. Bu ayrıntıları *www.w3.org* `<http://www.w3.org>` 'dan ulaşabileceğiniz birçok dökümandan ya da bu dökümanın kaynak kodu dahil olmak üzere İnternet üzerinde bulacağınız birçok formun kaynak kodundan yararlanarak öğrenebilirsiniz.

Örnek CGI Programı

Sıradan bir CGI programı bir html formundan girdilerini alıp, çıktıları HTML formatında ekrana yazar. CGI programının çıktıları web tarayıcınız tarafından yorumlanır.

Bir GCI programı, herhangi bir programalama dilinde yazılabilir. Çok kullanılan diller arasında C, Perl ve daha çok basit işler için de olsa shell gösterilebilir. Bu uygulamada C kullandık. Aşağıda en basit haliyle CGI için bir `helloworld.c` göruyorsunuz.

```
/* helloworldcgi.c -- ilk CGI programi. Bu program formdan gelen girdiyi
 * dikkate almayip, sadece ekrana(html tarayicisina) 'hello world' yazisini
 * basar
 */
#include <stdio.h>

main (){
/*      DIKKAT!!! asagidaki satir olmazsa programiniz calismaz.
*/
        printf("Content-type: text/html\n\n");
        printf("hello world!");
}
```

Aynı örneği bir bash programı olarak yazacak olsaydık:

```
#!/bin/bash
# CGI icin ornek kabuk programi

echo 'Content-type: text/html'
echo
echo
echo 'hello world!'
```

Bu dökümanın bundan sonraki kısmındaki örnekler C için hazırlanmıştır. CGI programı yazmak için gereken daha fazla bilgi sadece, C (ya da ilgili dil) ve HTML bilgisine ek olarak sadece programınızın girdiyi nasıl alacağı. C için en sevdiğiniz C kitabını, HTML içinse İnternet üzerinde bulacağınız dökümanları öneririz.

HTML Form'unun çıktısı 'ad=değer' şeklinde '&' ile ayrılmış tek satırlık bir karakter dizisinden oluşur. Bu dizideki boşluklar '+' ile ASCII setindeki alfa-nümerik karakterler dışındaki bütün karakterler (%, +, ç, Ö gibi) üç karakterlik '%XX' şeklinde bir ifadeyle gösterilir. Buradaki XX ifade edilen karakterin onaltılık sayı sistemindeki kodudur. Formun çıktısını, gets(), scanf() gibi herhangi bir C I/O fonksiyonuyla okuduktan sonra, eldeki karakter dizisinden istediğiniz bilgileri almalısınız. Bu diziyi isterseniz kendi yazdığınız programla isterseniz İnternet üzerinde heryerde bulabileceğiniz hazır kütüphanelerle işleyebilirsiniz.

CGI programının çıktısı istediğiniz gibi bir HTML dökümanı olabilir. Örnek bir HTML ve FORM dosyası aşağıdaki gibidir :

```
<hr>
<p>
<form method="POST" action="http://penguen.cclub.metu.edu.tr/cgi-bin/deneme"<P>
E-mail adresiniz :  <INPUT NAME="e-mail" SIZE=38></P><P>
Isminiz : <INPUT NAME="isim" SIZE=42><P>
Buraya birseyler yazabilirsiniz: <P>
<TEXTAREA name="body" rows=10 cols=50> birseyler </TEXTAREA>
<P>
<input type="submit" value="Gonder">
<input type="reset" value="Temizle"></P>
</FORM>
<p>
<hr>
```

Form, hayali penguen.cclub.metu.edu.tr adresi üzerindeki "deneme" isimli programı çalıştıracak şekilde ayarlanmıştır.

Chapter 12

X Window Grafik Arabirimi

Açık sistemlerin kullanıcıya sunduğu en büyük özelliklerinden biri olan X Window, Linux'un doğduğu andan itibaren destek görmeye başladı. İnternet üzerinde bedava dağıtılmasıyla Linux dağıtımı altında bir standart olarak kendine yer edindi. Daha önce bir UNIX sistem üzerinde X Window kullandıysanız bu bölümü kavramak gayet kolay olacaktır.

X Window'un Linux altındaki son sürümü Nisan 1994'te çıkan *X11R6*'dır. Aradan uzunca bir süre geçmesine rağmen bu sürümde kayda değer bir değişiklik olmadı. Bunun yanında X Window'da kullanılan pencere yöneticisi (window manager) çeşidi neredeyse 20'ye yaklaştı. Pencere denetleyicilerin herbirinin farklı zevkten kullanıcılar için farklı özellikleri vardır. Bunlar, X Window'a açılan ve uygulama programları ile kullanıcı arasındaki bilgi alış-verişini sağlayan arabirimdir.

X Window (veya kısaca X) istemci-sunucu modeline göre çalışır. Ana makina üzerinde çalışan X sunucusu, grafik donanımı üzerindeki tüm giriş-çıkış yetkilere sahiptir. Bir X istemcisi, sunucuya bağlanarak istediği işlemleri sunucuya yaptırır. İstemcinin görevi emir vermek, sunucunun ise verilen emri görünür hale getirmektir.

Aslında bilinmesi gereken en önemli kavram, X'in ağ tabanlı bir yapıya sahip olmasıdır. Bir istemci ve sunucunun olduğu her yerde ağdan da söz edebiliriz. X, ister konsol üzerinde (yerel makinada) isterse ağ üzerinde (uzak makinada) çalışabilir. Başka bir makina üzerindeymiş gibi o makinanın X Window uygulama ve kaynaklarından yararlanabilirsiniz.

Yukarıda kısaca bahsedilen pencere yöneticisi, ekrandaki pencerelerin denetiminde söz sahibi olan tek programdır. Her pencerenin nasıl görüneceğini, ne kadar büyük olduğunu bilir; kullanıcı tarafından verilen küçültme, pencere kapatma gibi komutları işler.

12.1 X Window Kurulumu ve Konfigürasyonu

X Window' un 386 ve yukarı tabanlı makinalar için geliştirdiği sisteme *Xfree86* adı verilir. Linux da Xfree86'yı Slackware dağıtımına ekleyerek kullanıcıların hizmetine sunmuştur. Xfree86 sürüm 3.1'in desteklediği kartlara ait çipsetler kitabın başında verilmiştir. Bu çipsetler, PCI ve VLB dahil, her tür veri yoluna sahip anakartlar üzerinde sorunsuz çalışır.

Linux ile uyumlu çalışabilen yeralan çipsetlerin hepsi 256 renk (8 bit) desteklerler. Eğer kart üzerinde yeteri kadar RAM var ise (en az 2MB), ekranınızı 16 bit (65536 renk) açma imkanınız olur. Özellikle S3 kartları Linux Xfree86 altında sorun çıkartmadan 65536 rengi kolayca gösterebilirler.

X Window üzerinde rahatça çalışabilmek için en az 486 tabanlı bir bilgisayara ihtiyacınız var. Bana sorarsanız rahat bir çalışma için 16 MB RAM ve Pentium 100 bilgisayar işinizi görecektir. Renk sorunu yaşamamak için ise 2MB görüntü hafızası olan bir grafik kartı kullanın. X Window, sistemi fazlaca yüklediğinden takas alanı için fazla söz söylemeye gerek yok sanırım. Unutmayın ki, X'e yatırım yapmak istiyorsanız paranızı hafıza için ayırın.

Xfree86, Slackware tarafından dağıtılan paketlerin X bölümünde yeralıyor. Herhangi bir nedenden dolayı sistemde X yoksa 'pkgtool' yardımıyla sisteminize kurun. Gerekli kütüphaneleri eklemeyi unutmayın.

12.2 X Konfigürasyonu

X Window kurulumu basit olmasına rağmen konfigürasyonu biraz başınızı ağrıtabilir. Slackware 2.2 dağıtımı ile gelen konfigürasyon programları, kullanıcının hata yapmasını affetmiyordu. Yanlışlıkla basılan bir tuş yüzünden baştan başlamak zorunda kalınabiliyordu. Slackware 3.3 altında, grafik arabirimini kullanan `XF86Setup` programı ile bir fare yardımıyla rahatlıkla ve kısa sürede X konfigürasyonu tamamlanabiliyor. Ücretli sürümlerden bir tanesi elinizin altında ise işler daha da kolaylaşıyor, neredeyse donanım hakkında hiç bir şey bilmeden X kurulabiliyor.

Konfigürasyon işlemlerinden önce hazırlık aşamasında `XF86Config(1)` ve `Xfree86(1)` man sayfalarını okuyun. Bu dosyalarda X Window için oluşturulacak olan `/etc/XF86Config` dosyasının hakkında detaylı bilgi vardır. XF86Config dosyası X açılırken okunur ve fare, ekran çözünürlüğü, renk sayısı gibi özellikler belirlenir. Ekranınızın kalitesine göre açmak istediğiniz çözünürlük hakkında bir miktar fikir sahibi olmalısınız. Bazı eski ekranlar 1024x768 çözünürlüğü desteklemezler, bu yüzden 800x600 gibi düşük bir çözünürlük ile işe başlayın. Daha ayrıntılı bilgi için XFree86-HOWTO dökümanlarına göz atabilirsiniz.

Linux'un Slackware 3.3 sürümü, kullanıcıya X konfigürasyonu için iki program sunar. Birincisi, yukarıda da belirtilen ve zayıf bir arabirimi olan xf86config, diğeri de XF86Setup. İkincisini kullanmanız, her bakımdan avantaj olarak göze çarpıyor. Program, bazı ayarlamaları kullanıcıya bırakmadan (örneğin çipsetin seçimi) kendisi hallediyor. Burada yeralan soruları doğru bir şekilde cevapladıktan sonra X açmaya hazırsınız demektir.

Aşağıda XF86Config dosya formatı hakkında biraz bilgi verilmiştir. X Window'u kurduktan sonra kendi zevkiniz doğrultusunda bu dosya üzerinde değişiklik yapabilirsiniz.

Dosya, pekçok bölümden (section) oluşur ve her bölüm, *Section "bölüm-ismi"* ve *Endsection* kelimeleri arasında yeralır. Ekran yazıtiplerinin ve kullanılacak renklerin patikasını belirleyen bölüm dosya altında şu şekilde görünecektir :

```
Section "Files"
```

```
        RgbPath         "/usr/X11R6/lib/X11/rgb"
        FontPath        "/usr/X11R6/lib/X11/fonts/misc"
        FontPath        "/usr/X11R6/lib/X11/fonts/75dpi"
EndSection
```

İstediğimiz kadar yazıtipi girişi yapabiliriz. Üç tanım satırının arasına, örneğin X altında kullanacağımız Türkçe yazıtiplerini yerleştirelim. Bu yazıtiplerini daha önce kurmuş ve gerekli konfigürasyon işlemlerini yapmış olmanız gereklidir. X için Türkçe yazıtiplerini *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/metu/trfonts>` adresinde bulabilirsiniz. Adreste yazıtiplerinin nasıl kurulacağına ilişkin gerekli dökümanlar da bulunmaktadır.

```
Section "Files"
        RgbPath         "/usr/X11R6/lib/X11/rgb"
        FontPath        "/usr/X11R6/lib/X11/fonts/misc"
        FontPath        "/usr/X11R6/lib/X11/fonts/75dpi"
        FontPath        "/usr/X11R6/lib/X11/fonts/turkce/75dpi"
EndSection
```

Klavye ve farenin tanımlandığı bölümler, “Keyboard” ve “Pointer” adı altındadır. Açıklama satırlarının başında “#” karakteri yeralır. Örnek dosyanın yeraldığı makinada farenin kullandığı protokol, “MouseSystems” olup hemen her standart fare için bu protokol önerilir. Diğer protokoller arasında *Busmouse, Logitech, Microsoft, MMSeries, Mouseman, MouseSystems, PS*2, MMHitTab/ vardır. X açıldığı zaman herşey yolunda gitmesine rağmen farenin ikonu hareket etmiyor veya ekranda deli gibi oradan buraya koşuşturuyorsa farenin kullandığı protokolü değiştirin. Bazı farelerin altındaki küçük anahtarın bulunduğu konumu değiştirmeniz de işe yarayabilir.

```
Section "Keyboard"
        Protocol "Standard"
        Autorepeat 500 5
        ServerNumLock
EndSection

Section "Pointer"
        Protocol "MouseSystems"
        Device "/dev/mouse"
Emulate3buttons         # iki tuslu microsoft farede 3. tusun,
                        # iki tusla basilmasini saglamak icin...
ChordMiddle             # 3 tuslu Logitech fareler icin
EndSection
```

Farenin kullandığı seri port /dev/mouse ’dur. Bu dosya, genellikle uygun olan bir seri arabirime (örneğin /dev/ttyS0) bağlıdır.

`XF86Config` dosyasının en önemli bölümü monitörle ilgili bilgilerin yeraldığı kısımdır. Bu bölümde monitörün yatay (Horizontal Sync) ,düşey frekansları (VertRefresh) ve frekans aralıklarıyla sürücü saat frekansı (dot-clock frequency) belirtilir. Kullandığınız monitörle ilgi konfigürasyon sırasında girilen bilgiler bu bölümde yeralır.

```
Section "Monitor"
        Identifier "CTX 5468 NI"

HorizSync 30-38, 47-50
        VertRefresh 50-90

        # Modes : Name     Dotclock     horiz        vert

        Modeline "640x480"   25   640 664 760 800   480 491 ...
        ...

EndSection
```

“Identifier”, kullanılan monitörün tipini verir. X bunu kullanmaz, istediğiniz herhangi bir kelimeyi yazabilirsiniz. HorizSync, monitörün yatay frekansını kHz cinsinden belirtir. VertRefresh ise monitörün tazeleme frekansı olup Hz cinsindendir. Her iki değer, frekans aralığı veya tek sayı olarak verilebilir.

“Modeline” kelimesi ile başlayan satırda belirli bir çözünürlük için gerekli “Dotclock” (sürücü saat frekansı) ile yatay ve düşey tarama frekansları MHZ cinsinden yazılıdır. Bu tanımlar ile ilgili daha geniş bilgiyi /usr/X11/lib/X11/doc dizini içinde bulabilirsiniz. Yukarıda yeralan bilgileri kendiniz için kullanmamanız, monitörünüzün sağlığı açısından tavsiye edilir.

Monitör bilgilerinin yeraldığı bölümden sonra ekran ve alt bölümü olan görüntü bölümü gelir.

```
Section "Screen"
        ...
                Depth 8
                Modes "1024x768" "800x600"
                ViewPort 0 0
                Virtual 1024 768
EndSubsection
EndSection
```

Driver kısmında kullanılacak olan X sunucusunun tipi (SVGA, hızlandırılmış, vb.) girilir. S3, Weitek, ATI gibi hızlandırılmış bir görüntü kartı kullanılıyorsa “Accel”, normal (generic) bir kart ve kaliteli bir monitör varsa “SVGA” ve siyah beyaz VGA bir monitörle birlikte tek renk (monochrome) destekleyen kart kullanılıyorsa “VGA2” yazılıdır.

“Indentifier” ve “Monitor” bölümlerinde yazılanlar gözardı edilebilir.

“Display” alt bölümünde, görüntü özellikleri yeralır. “Depth” kelimesinin karşısında yeralan rakam, piksel başına kaç bitin, başka bir deyişle kaç rengin kullanılacağını belirtir. Monitörün belirli bir çözünürlükte kaç renk birden gösterebileceğini bulmak için bu sayıyı 2’nin kuvveti olarak yazın ve hesaplayın. Günümüz XFree86 sunucuları genellikle 256 (8 bit), 65536 (16 bit) ve 16.7 milyon (24 bit) renge kadar çıkabilirler.

“Modes” karşısında yeralan ekran çözünürlük listesi, Linux’un X altında çalışırken görüntüleyebileceği çözünürlükleri belirtir. X altında burada yazılı çözünürlükler arasında geçiş yapabilmek için Control-Alt-Nümerik + veya Control-Alt-Nümerik - tuşlarını kullanabilirsiniz. “Modes” ile başlayan satırdaki ilk çözünürlük, X Window açıldığı zaman kullanılacak olan çözünürlüktür (yukarıdaki örnekte 1024x768).

“Virtual” karşısında yeralan x ve y sayıları sırayla ekranın genişliğini ve yüksekliğini belirtir. X açıldıktan sonra ekran genişliği yetmediği durumlarda kullanılabilir. Bu iki sayıyı, görüntü kartı hafızası yetene kadar artırın. Örnek olarak 1024x768 ekran çözünürlüğünde 1152 ve 900 sayılarını yerleştirin. X Window açıldığı zaman fare ile ekranın sağ alt köşesine gidin. Ekranın kaydığını göreceksiniz. Gerçek ekran boyutunu “Virtual” parametreleri belirler.

“Virtual” girdilerinin ardından verilen “ViewPort” parametresi de ekranın sol üst köşesinin hangi koordinatlara tekabül edeceğini tanımlar. Yukarıdaki örnekte sözkonusu noktayı (0, 0) olarak belirttik.

12.3 X Window Komutları

Konfigürasyon işlemini bitirdiğimize göre sırada X Window’u denemek var. Aşağıdaki komutu yazmadan önce PATH değişkeninin içinde `/usr/X11/bin` olduğundan emin olun.

```
$ startx
```

Ekranda X sunucunun dosyadan okuduğu satırlar ile ilgili bilgileri sırayla göreceksiniz. Ardından ekran kararacak ve X açılacaktır. Hevesinizi aldıktan sonra Control-Alt-Backspace (Delete değil! :) tuşlarına basın ve X sunucusunu öldürerek başladığınız vt100 terminaline geri dönün.

Diyelim ki ekran karardı ama X açılmadı. Konfigürasyonda hatalar olabilir. Ekranda çıkan hata mesajlarını kaydedip incelemeniz gerekir. Çok yapılan konfigürasyon hatalarının başında çipsetin ve ekran kartı bilgilerinin yanlış girilmesi, “Virtual” parametrelerinin hatalı olması ve görüntü kartı hafızasının ekranı açmaya yetmemesi vardır. Bu sefer hata mesajlarını ayıklayıp bir dosyanın içine atarak sunucuyu tekrar çalıştırın. Sonra da dosyayı inceleyip sorunun nerede olduğunu bulabilirsiniz.

```
$ startx > /tmp/hatalar 2>&1
```

Ekrana çıkması gereken tüm satırlar `/tmp/hatalar` dosyasına yönlendirilecektir. Çipseti doğru olarak seçmek için `SuperProbe` programını kullanabilirsiniz. Bu komutu kullandıktan sonra ekrandaki çıktıdan aldığınız bilgileri konfigürasyon sırasında programa girin.

X sunucuyu başlatmaya yarayan `startx` komutu, ev dizini içindeki `.xinitrc` dosyasını arar. Bulamazsa sistemde `/usr/X11/lib/X11/xinit/` dizini içinde yeralan xinitrc dosyasını okur. Bu dosyanın içine X açıldığı zaman başlayacak olan programları koyabilir, ekranın renk düzenlemesini yapabilirsiniz. Örnek bir `.xinitrc` dosyası aşağıdaki gibidir :

```
#!/bin/bash
#
#start xterms
#
xterm -geometry 80x25+10+10 -bg black -fg white &
xterm -geometry 80x25 -bg darkgray -fg white &

xclock &
xload &
xbiff &
xsetroot -solid darkblue &

# startx wm
exec fvwm
#
```

`.xinitrc` dosyası gayet kolay ve anlaşılır bir dile sahiptir. Sadece hangi X programlarını çalıştırmak istediğinizi buraya yazın ve arkaplanda bırakın (bunu unutursanız X'ten çıkıncaya kadar sadece ilk program çalışır). İlk iki satırda xterm çalıştırılıyor. Hemen hemen tüm X programlarının ortak özelliği, kullanıcının bu programları ekranda hangi koordinatlara yerleştirileceklerini belirleyebilmesidir. İlk xterm programı 80x25 karakter boyutlarında olup sol üst köşeden itibaren 10 piksel sola, 10 piksel sağa yerleştirilecektir. Xterm'ün arkaplan ve yazıtipi rengi sırayla *-bg* ve *-fg* parametreleriyle belirtilir. Renkler ingilizce isimleriyle verilir. Bunlar hakkında daha fazla bilgi için `/usr/X11/lib/X11` dizini altındaki `rgb.txt` dosyasına göz gezdirin.

Ardından sırasıyla *xclock* (X altında çalışan saat programı), *xload* (sisteme ne kadar yük bindiğini gösterir) ve *xbiff* (e-posta gelince haber verir) çalıştırılıyor. Aslında boş vaktinizde bakmanızı bekleyen, X Window ile birlikte gelen onlarca uygulama programı vardır. Bunları da arkaplanda çalıştırın.

Sondan bir önceki komut olan `xsetroot`, bazı hoş özellikleri olan bir programdır. Bu satırdaki görevi, X Window arkaplan rengini koyu mavi yapmaktır. Unutmayın, `.xinitrc` dosyasının içine X açılırken başlatmak istediğiniz her programı koyabilirsiniz.

Dananın kuyruğunun koptuğu son satırda ise X sunucusu, pencere yöneticisini çalıştırıyor. Daha önce de belirtildiği gibi wm, kullanıcı ile X arasındaki idareyi sağlıyor. Eğer yeterli RAM'ınız varsa ve hızlı bir bilgisayara sahipseniz *bu sayfadaki* `<http://www.plig.org/xwinman>` AfterStep ve Enlightenment pencere denetleyicilerini deneyebilirsiniz.

12.4 Fvwm Pencere Yöneticisi

Fvwm, Linux için özel olarak geliştirilmiş bir wm'dir. Diğer wm'lere karşı belirli üstünlükleri olan fvwm'in rahatça konfigüre edilebilir bir rc dosyası, renkli ikon desteği, "pager" (birden fazla ekranı aynı anda kontrol etme yeteneği), farenin tuşlarına basıldığı zaman ekrana gelen "popup" menüleri ve Motif desteği vardır.

“Pager”, fvwm’i üstün kılan ve kullanıcının ”olsa da kullansak” beklentilerine tam anlamıyla cevap veren bir özelliktir. Ekranın bir köşesine yerleştirilen bir pencere ile farenin tuşlarına basarak istediğiniz başka bir ekrana geçmek mümkün olur.

Fvwm çalıştırıldığı zaman ev dizininde yeralan `.fvwmrc` dosyasını okumaya çalışır, bu dosyayı bulamazsa sistemdeki örnek fvwmrc dosyasını (`/usr/X11/lib/X11/fvwm/system.fvwmrc`) açılış konfigürasyon dosyası olarak kabul eder. Eğer Slackware 3.0’dan daha yeni bir dağıtım kullanıyorsanız, sistemde X açarken öntanımlı olarak fvwm95 kullanıyor olacaksınız. fvwm95 pencere yöneticisinin örnek konfigürasyon dosyalarını `/usr/X11/lib/X11/fvwm95-2` dizininde bulabilirsiniz. Beğeninize hitap eden pencere denetleyicisine uygun olarak sistemde yeralan örnek dosyayı kopyalayıp zevkinize göre istediğiniz değişiklikleri yapabilirsiniz. Burada fvwm’in birkaç özelliğine değinip örnek bir konfigürasyon dosyasındaki bazı satırları inceleyeceğiz.

```
#set up colors
StdForeColor            white
StdBackColor            midnightblue
HiForeColor             white
HiBackColor             red
Font                    -adobe-helvetica-meduum-r-normal-*-*-120-*
WindowFont              -adobe-helvetica-bold-r-normal-*-*-120-*
#
```

Farenin üzerinde kalan pencerelerin rengini (ki bu sırada pencereye klavyeden bilgi girişi yapılabilir) diğerlerinden ayırmak mümkündür. Böylece kullanıcı hangi pencerenin aktif olduğuna dair tereddüte düşmez. HiForeColor ve HiBackColor değerlerinin gösterdiği renk, aktif pencerenin rengidir. ForeColor, pencerede yeralan metnin rengi, BackColor ise pencere arkaplan rengini temsil eder.

Aktif olmayan pencereler için StdForeColor ve StdBackColor komutları geçerlidir. Font ve WindowFont parametreleri de sırayla pencerenin ismini gösteren yazıtipi ve pencere içinde kullanılan yazıtipini belirler.

Sıkça kullanılan bir başka parametre de “Style” dır. Style yardımıyla ekrandaki bir pencerenin diğer pencerelerle olan hiyerarşik düzenini ayarlayabilirsiniz.

```
Style "xclock" NoBorder
Style "xbiff" StaysOnTop
Style "xterm" Sticky
```

İlk sırada yeralan komut ile xclock saat programı, fvwm tarafından penceresiz olarak ekrana konuyor. İkinci komut ile xbiff uygulama programı, üzerine hangi pencere gelirse gelsin üstte kalıyor. Sticky ise Pager yardımıyla pencere değiştirdiğiniz zaman istediğiniz programın ekranda eski yerini almasını sağlıyor.

Farenin sağ tuşuna bastığınız zaman ekrana gelen menüye ”popup” ismi verilir. Bu menülere de .fvwmrc dosyası yardımıyla kolayca ekleme yapılabilir, üzerinde değişiklik uygulabilir. Menü, konfigürasyon dosyası içinde, ”Popup” ve ”EndPopup” satırları arasında tanımlanır ve bir isim verilir:

```
Popup "Menu ismi"
        Title "XClients"
        Exec "Mosaic"   exec Mosaic &
        Exec "xterm"    exec xterm &
        Exec "emacs"    exec emacs &
        Nop ""
Exec "rlogin"   exec xterm -e rlogin orca &
        Exec "telnet"   exec xterm -e telnet &
        Nop ""
Exec "xcalc"    exec xcalc &
        Exec "xman"     exec xman  &
EndPopup
```

Farenin tuşuna basılınca ekrana, *exec* kelimesinin sağında yeralan ve iki tırnak içine alınan menü başlıkları gelir. Her menü başlığının karşısında, o program seçildiği zaman çalışacak olan program yeralır. “Nop”, menü üzerinde yatay bir çizgi çeker.

X altında kullanılabilecek yazıtipi sayısı neredeyse sınırsız denilebilecek kadar çoktur. Bu yazıtiplerini kullanmak için iki yolunuz var. X Window açılırken çalıştırılan xterm'e standart parametre olarak girilir, veya doğrudan komut satırı üzerinde kullanmak istediğiniz yazıtipi seçebilirsiniz. Halihazırda kullanılabilecek olan yazıtiplerini görebilmek için şu komutu kullanın :

```
$ xlsfonts
```

Ekran büyük bir hızla kayacak ve xinitrc dosyasının FontPath parametresinin gösterdiği dizin altındaki dosyaları temsil eden yazıtipleri ekrana gelecektir. Her satır, bir yazıtipi ismine tekabül eder. Görüldüğü gibi X altında kullanılan yazıtipleri isimleri gayet uzun olup toplam 14 parametreden oluşur. Bu satırlardan uygun olan birini seçip (örneğin kalın yazıtipini)

```
$ xlsfonts | grep bold
```

veya belirli bir sırayı izlemek için

```
$ xlsfonts -fn -*-*-bold-*-*-*-*-*-*-70-iso8859-9
```

komut satırı üzerinden diğer xterm'ü açabilirsiniz.

```
$ xterm  -fn -misc-fixed-bold-r-normal--13-100-100-100-c-70-iso8859-9
```

iso8859-9, içinde Türkçe yazıyüzlerinin bulunduğu standartı gösterir.

12.4.1 X Altında Türkçe Yazıtipi ve Klavye Kullanımı

Kullandığınız her işletim sisteminde Türkçe karakterlerin gösterimi ve Türkçe klavye uyarlaması sıkıntısı çekiyor olmalısınız. Linux'ta Türkçe yazıtipleri `xterm` altında `-fn` parametresi ile rahatlıkla görülebilir, böylelikle Türkçe okuyabilirsiniz. *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/metu/trfonts>` adresinden temin edeceğiniz *Turkcebdf.tgz* paketinin içinden çıkan küçük program, Türkçe klavye dağılımını ayarlıyor. Bu program çalıştırıldıktan sonra Alt tuşu yardımıyla Türkçe karakterlerin basılmasını sağlanabilir. Örnek olarak *Alt-u* ile ü, *Alt-i* ile ı karakterleri görüntülenebilir.

Türkçe karakterlerinin görüntüleyebilmek için önceden Türkçe yazıtiplerini sisteminize kurmuş olmalısınız.

Bu çözümlerin çalışması için X konfigürasyonu sırasında XKB isteyip istemediğiniz sorusuna hayır, Alt keys -> Meta keys dönüşümü isteyip istemediğiniz sorusuna evet yanıtı vermelisiniz.

Ayrıca bazı yazılımlarda Türkçe harfleri kullanabilmeniz için iso8859-9 karakter setini seçmeniz gerekir. Örneğin `pine`'ın Setup->Config menüsünde ya da `.pinerc` dosyasında bulunan character-set değişkenine iso8859-9 değerini vermelisiniz. `elm` için de benzer şekilde `.elmrc` dosyasında ayarlama yapılmalıdır.

Chapter 13

Linux İşletim Sisteminde Güvenlik

Bilgisayar ağları ve özellikle İnternet, kullanıcılara büyük olanaklar verirken ciddi güvenlik sorunlarını da beraberinde getirdi. Ağa bağlı bilgisayarlarda çalışanların bu güvenlik sorunları konusunda bilgili olmaları zorunlu hale geldi. Ancak güvenlik ile kullanım kolaylığının çoğu zaman birbirleriyle çelişmesi ve kullanıcıların seçimlerini genellikle kullanım kolaylığından yana kullanmaları güvenlik önlemlerinin yerleşmesini engelledi. Bu bölüm, akademik ortamlarda yaygın olarak kullanılan Linux işletim sisteminde sistem sorumlularına güvenlik için yapabilecekleri konusunda yol göstermek ve kullanıcılara tehlikeleri tanıtmak amacıyla hazırlanmıştır.

13.1 Şifreler ve Şifre Seçimi

Çok kullanıcılı işletim sistemlerinde kullanıcının kimliğinin belirlenmesi büyük önem taşır. Hem sistemi kullanmaya yetkisi olmayan kişilerin sisteme girmelerinin engellenmesi, hem de sistemdeki kullanıcıların birbirlerinden ayırdedilebilmeleri için, her kullanıcıya bir şifre verilir ve sisteme giriş başta olmak üzere tüm kritik işlemlerde kullanıcıya şifresi sorulur. Şifreler, diğer kullanıcı bilgileriyle birlikte, `/etc/passwd` veya `/etc/shadow` dosyasında tutulur.

Bazı uygulamaların şifre dosyasının bazı alanlarına erişmeleri gerektiğinden şifre dosyası, sistemdeki bütün kullanıcılar tarafından okunabilecek bir dosya olmalıdır. Bu nedenle şifreler bu dosyaya açık halde değil, şifrelenerek yazılırlar.

Şifre yönteminin güvenilirliği, sistemdeki kullanıcıların şifre seçiminde gösterdikleri özene bağlıdır. Kolay akılda kalacak şifreler, çoğu zaman kolay tahmin edilebilir. Buna karşılık, zor tahmin edilebilen şifreler kullanıcıların akıllarında kalmayabileceklerinden bir yere yazmalarına ve böylece yeni tehlikeler oluşturmalarına zemin hazırlayabilir.

Şu tip şifreler kolay tahmin edilebilen şifreler sayılmaktadır:

1. Kullanıcı ile yakınlığı olan kişilerinkiler (kendisi, ailesi, arkadaşları, yakınları) başta olmak üzere bütün erkek ve kadın isimleri

2. Doğum tarihleri

3. Kullanıcı ile ilgili herhangi bir bilgi (kullanıcı adı, oda numarası, telefon numarası, arabasının plaka numarası, sosyal güvenlik numarası)
4. Klavyede belli bir düzene göre ardarda gelen harflerden oluşan şifreler (örneğin qwerty veya abcdefg gibi)
5. Anlamlı bir sözcük (bilgisayar terimleri, yer isimleri)
6. Yalnızca küçük (ya da yalnızca büyük) harflerden oluşan şifreler
7. Yukarıdakilerden birinin başına ya da sonuna bir rakam eklenerek oluşturulan şifreler
8. Yukarıdakilerin ters yazılışları

İyi bir şifre üretmek için önerilen iki yöntem vardır:

1. İki sözcüğün aralarına bir rakam ya da noktalama işareti konarak birleştirilmesi
2. Seçilen bir cümlenin sözcüklerinin baş harfleri

13.1.1 Tehlikeler

Şifre dosyasına erişimi olmayan bir saldırgan, hedef seçtiği bir kullanıcının şifresini deneyerek bulmak zorundadır. Olası bütün şifrelerin çokluğu gözönüne alınırsa bu tip bir saldırının etkisiz olacağı düşünülebilir. Ancak, çoğu sistemde, şifresi boş olan, kullanıcı adıyla aynı olan ya da sistem sorumlusunun geçici olarak verdiği basit şifreyi değiştir(e)meyen kullanıcıların sayısı azımsanamayacak düzeydedir. Bir tek kullanıcının bile şifresinin elde edilmesi sisteme giriş için yeterli olduğundan güvenlik açısından büyük sorunlar yaratabilir.

Saldırgan, şifre dosyasının bir kopyasını alabilirse işi oldukça kolaylaşır. Hem çok daha hızlı çalışabilir, hem de hedef sistemin sorumlusunun dikkatini çekmek tehlikesinden kurtulur. Bir saldırganın şifre dosyasını eline geçirmesi birkaç şekilde mümkün olabilir. Örneğin bir kullanıcının şifresini elde ederek sisteme girer ve dosyayı alır, bazı programlardaki hatalardan yararlanarak sisteme girmeden dosyayı elde edebilir, sistemdeki bir kullanıcı şifre dosyasını saldırgana gönderebilir veya saldırgan, sistemdeki kullanıcılardan biridir.

`/etc/passwd` dosyasındaki şifreleri kırmaya çalışan çeşitli yazılımlar bulunmaktadır. Bunların en etkililerinden ve en yaygın kullanılanlarından biri *Crack* isimli programdır. Alec E. Muffett tarafından geliştirilen Crack, sözlük saldırısı yöntemini kullanan bir şifre kırma programıdır. Giriş olarak verilen sözlüklerdeki sözcükleri şifre dosyasındaki 'salt' bilgileriyle şifreleyerek şifre olarak kullanılıp kullanılmadıklarına bakar. Ayrıca sistemin kullanıcı bilgilerinden de olası şifreler üretmeye çalışır. Bu paketi *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/sunsite/system/admin>` adresinden temin edebilirsiniz.

13.1.2 Önlemler

Şifre güvenliğinin en önemli şartı, başta sistem sorumluları olmak üzere, bütün kullanıcıların iyi şifre seçmenin önemini kavramalarıdır. Herkes, şifresinin yetkisiz birinin eline geçmesi durumunda kendisi ve diğer kullanıcılar açısından oluşan tehlikeleri anlamalıdır. Nelerin kötü şifre olduğu ve nasıl iyi şifre seçilebileceği konusunda kullanıcılar eğitilmelidir. Bu koşul gerçekleşmedikçe aşağıda sözü geçen önlemlerin çoğu da başarısız olacaktır.

1. **Şifre seçiminin kullanıcıya bırakılmaması** : Şifreler, sistem sorumlusu tarafından ya da rasgele şifre üreten bir program tarafından seçilerek verilir ve kullanıcılara şifrelerini değiştirme hakkı tanınmaz. Bu yöntem, iyi şifreler üretmekle birlikte, kullanıcıların akıllarında tutamayacakları şifreleri bir yere yazmalarına neden olur.
2. **Şifre seçiminin kısıtlanması** : Kullanıcının kötü olduğu bilinen bir şifre seçmesi engellenir.
3. **Şifre dosyasının sistem sorumlusu tarafından kırılması** : Sistem sorumlusu, zaman zaman, şifre dosyasını Crack tipi bir program ile tarayarak zayıf şifreleri arar. Bulduğu zayıf şifreli kullanıcıların hesaplarını kilitler.
4. **Şifrelerin geçerlilik sürelerinin kısıtlanması** : Yeterince sık değiştirilmeyen şifreler, kuvvetli de olsalar, saldırgana aramak için daha uzun zaman tanıdıklarından tehlike yaratırlar. Bu nedenle, şifrelerin belli bir süre sonunda geçerliliklerini yitirerek yeni bir şifre verilmesi zorunluluğu getirilebilir.
5. **Gölge şifreler (shadow passwords)** : Şifre dosyasının bazı alanlarının herkes tarafından okunabilir olması gerekse de, şifre alanının herkes tarafından okunabilir olması gerekmez. Bu nedenle, şifrelenmiş şifreler ancak sistem sorumlusunun okuyabileceği bir dosyaya alınabilir. Şifre dosyasındaki şifre alanına herhangi bir bilgi yazılırken (kullanıcı adı, * işareti gibi), gölge dosyasına (/etc/shadow) şifrelenmiş şifreler konur.

   John F. Hough II tarafından geliştirilen Shadow Password Suite, sistemdeki şifre güvenliğini artırmaya yönelik bir pakettir. Gölge şifrelerin yanısıra, daha uzun şifreleri (16 karakter) desteklemesi ve şifrelerin geçerlilik sürelerini kısıtlaması nedeniyle sistem sorumlularının çokça kullandığı araçlar arasında yer alır. *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/sunsite/system/admin>` adresinde bulunabilir. Slackware 3.2 sürümünden itibaren shadow desteği vardır.

13.2 Dosya Güvenliği

Her dosyanın bir sahibi, bir de grubu vardır. Dosya uzerinde kimin hangi işlemleri yapabileceğine dosyanın sahibi olan kullanıcı karar verir. Erişim hakları, dosyanın sahibi, grubu ve diğerleri için ayrı ayrı belirtilir.

```
-rwxr-x---   1 uyar     users           4030 Dec  4 15:30 dene
```

Dizinler için de aynı erişim hakları modeli geçerlidir. Bir dizin üzerindeki okuma izni, dizin altındaki programların listesinin alınıp alınamayacağını, yazma izni dizinde yeni bir dosya yaratılıp yaratılamayacağını, çalıştırma izni de o dizine geçilip geçilemeyeceğini belirler. Yetkili kullanıcının (root) bütün dosyalar ve dizinler üzerinde (birkaç sistem dosyası ve dizini haricinde) bütün işlemleri yapma yetkisi vardır.

13.2.1 Tehlikeler

İşletim sisteminde ya da uygulama programlarında bir hata olmadığı sürece erişim izni olmayanlar dosyayı zaten okuyamayacaklardır. Asıl tehlike, yetkili kullanıcının yetkisini kötüye kullanarak kullanıcıların kişisel dosyalarını ve mektuplarını okumasıdır. Her şeye yetkisi olan bir kullanıcı, sistemin kararlılığını korumak için gerekli olmakla birlikte, güvenliği ve özel bilgilerin gizliliğini bir kişinin ahlakına bırakması açısından Linux (ve Unix) işletim sisteminin güvenliğinin en zayıf noktalarından biri olarak değerlendirilmektedir.

Saldırgan, sisteme girince, hem sonraki girişlerini kolaylaştırmak, hem de daha rahat çalışabilmek için bazı sistem dosyalarını ya da programlarını değiştirebilir. Örneğin, şifre dosyasına bir kayıt ekleyerek kendine yetkili bir kullanıcı yaratabilir. Kullanıcıların şifrelerini öğrenmek için login, passwd gibi programları değiştirebilir.

13.2.2 Önlemler

Şifre güvenliği sağlandığı sürece dosya erişimlerinde fazla bir güvenlik sorunu olmayacaktır. Bu konuda sistem sorumlusuna düşen, kullanıcılarını erişim haklarını nasil düzenleyecekleri konusunda bilgilendirmektir.

Şifre dosyası gibi metin dosyalarında değişiklik olup olmadığı gözle inceleme yaparak ya da basit komut satırı programları kullanarak bulunabilir. Çalıştırılabilir dosyalar gözle kontrol edilemeyeceğinden en uygun yöntem, dosya imzaları oluşturarak sağlam olduğu bilinen imzalarla yeni hesaplanan imzaları karşılaştırmaktır. *Tripwire* paketi, dosyalarda yapılan değişiklikleri farketmekte sistem sorumlusuna ve kullanıcılara yardımcı olur. Önce sağlam olduğu bilinen dosyaların dosya imzaları oluşturularak bir yerde saklanır. Sonraki çalıştırmalarda imzalar yeniden hesaplanarak eskileriyle karşılaştırılır ve farklı olanlar varsa bildirilir. Düzgün çalışma için özgün imzaların iyi korunması, mümkünse, üstüne yazılamayan bir ortamda saklanması gerekir. *coast.cs.purdue.edu* `<ftp://coast.cs.purdue.edu/pub/COAST/Tripwire>` adresinden temin edilebilir.

13.3 TCP/IP Ortamında Güvenlik

IP paketleri ağa açık halde gönderildiklerinden iletişim fiziksel olarak tamamıyla güvensizdir. Ağdan paketleri alabilen herkes içeriklerini rahatlıkla okuyabilir. Özellikle, bir ortak yol (bus) ile gerçeklenen ağlarda çok belirgin olarak ortaya çıkan bu sorun, şifreler başta olmak üzere pek çok duyarlı bilginin kolayca ele geçirilmesine yol açar.

Bir istemci, hizmet almak için bir sunucuya bağlanırken gerçeğe aykırı bir sayısal ya da sembolik adres bildirebilir. Internet'teki temel güvenlik sorunlarından biri, sayısal - sembolik adres dönüşümleri için kullanılan sunucuların yerel sistem sorumlusunun denetimi dışında olmaları ve çoğu protokolün güvenlik koşullarının sağlanabilmesi için bunlara gereksinim duymalarıdır.

Bir saldırgan, sunucu bir makinayi çökerterek hizmetlerin verilemez hale gelmesine yol açabilir. Buna, TCP/IP protokol ailesinin tasarımından, sunucu süreçlerdeki hatalardan veya işletim sistemindeki hatalardan kaynaklanan açıklar neden olabilir.

`rsh`, `rcp` ve `rlogin` komutlarının kullanılmasında şifre denetimlerinin atlanabilmesi için ya tüm kullanıcılar için geçerli sistem dosyasında (`/etc/hosts.equiv`) ya da kullanıcıya özel erişim dosyasında (`.rhosts`) ayarlamalar yapılmalıdır. Sistem dosyasında yer alan makinalar güvenilen (trusted) makinalar kabul edilirler ve bu makinalardan bağlanan kullanıcıların isimleri yerel şifre dosyasında varsa şifre denetimi yapılmaz.

13.3.1 Önlemler

İnternet hizmetlerini gerçekleştiren programların çoğu halen üzerinde geliştirme ve hata ayıklama işlemleri süren programlardır. Sistem sorumlusu, kullandığı programların gelişimini izleyerek bilinen hatası bulunmayan son sürümlerini edinmeli ve çalıştırmalıdır. Ayrıca, Usenet'te güvenlik ile ilgili forumları izleyerek ve CERT (Computer Emergency Response Team - *www.cert.org* `<http://www.cert.org>`) önerilerine uyarak güncel olan güvenlik boşluklarını öğrenmeli ve önlemlerini almalıdır.

Yapılabilecek en temel işlerden biri, verilmesine gerek olmayan ya da taşıdığı riske değmeyeceği düşünülen hizmetlerin sunucu süreçlerini çalıştırmamaktır. Sözgelimi, yerel ağda hiçbir NIS istemci makinası bulunmadığı halde bir bilgisayarın NIS sunucu sürecini çalıştırıyor olması gereksiz bir risk içerir. Bazı sistem sorumluları, kullanıcı isimleri ve sistemde kimin çalıştığı gibi bilgileri dışarıya aktardığı için finger hizmetini tehlikeli bularak kapatırlar. Hizmetin engellenmesi için işletim sisteminin açılış dosyalarından (`/etc/rc.d/rc.*`) ilgili sunucuları çalıştıran satırlar (özellikle netstat ve systat) kaldırılmalıdır.

Bazı durumlarda, hizmeti bütünüyle kapatmaktansa, erişimi koşullara bağlamak daha anlamlı olabilir. Bu koşullar, hizmetin hangi makinalara, hangi kullanıcılara, hangi saatlerde verileceği ya da verilmeyeceği şeklinde belirtilebilir. Bu tip kısıtlamalar getiren programların en yaygın kullanılanları *tcpwrapper* ve *xinetd* paketleridir. Wietse Venema tarafından geliştirilen tcpwrapper, sunuculara erişimi bağlantı isteğinde bulunan makina ve kullanıcıya gore kısıtlayan ve bağlantı istekleriyle ilgili kayıtlar tutan bir pakettir.

Bu paket Linux'un Slackware dağıtımıyla birlikte gelmektedir.

Güvenliğin önem taşıdığı yerlerde giderek yaygınlaşan alev duvarı tekniği, yerel ağdaki bilgisayarların dışarıdaki ağa doğrudan erişimlerini kaldırmayı öngörür. Ağın yönlendiricisi, ağda güvenli çalıştığı bilinen yalnızca bir makinanın dışarıyla bağlantı kurmasına izin verir. Diğer bilgisayarlar dış ağa çıkmak için bu makinadaki süreçlere bağlanırlar.

13.4 Kayıt Tutulması

Güvenliğin en önemli parçalarından biri, sistemin sürekli izlenerek, güvenliğe aykırı durumlar oluşup oluşmadığının, oluştuysa bunların sorumlularının kimler olduğunun belirlenmesidir. Bunun için, güvenliği ilgilendirebilecek her türlü olayın kaydı tutulmalıdır. Şu tip bilgiler, güvenlik açısından değer taşırlar:

- Başarısız veya başarılı olmuş sisteme giriş denemeleri
- Nerelerden, hangi hizmetler için bağlantı istekleri geldiği
- Hizmetler sırasında gerçekleşen dosya aktarımları

Linux'ta kayıt tutulması işini syslogd süreci görür. Hangi tür mesajların hangi dosyaya yazılacağı konfigürasyon dosyasında (`/etc/syslog.conf`) belirtilir. Genellikle makina ilk açıldığı zaman `/var/adm/messages` dizini altındaki `messages`, `xferlog`, `syslog` gibi dosyalara yazılan bu bilgileri isteğinize göre daha sistematik bir yapıda saklamak da mümkündür. Ağ ile ilgili hizmetleri denetleyen süreçler (*tcpwrapper*, *xinetd* gibi paketler) kayıt dosyalarına girmesini istedikleri bilgileri syslogd sürecine bildirirler.

Chapter 14

EK

14.1 İnternet üzerinde yeralan Linux dökümanları

Linux Sıkça Sorulan Sorular (Linux ÇSS) : Bu işletim sistemi hakkında linux@bilkent.edu.tr listesinden seçilerek biraraya getirilen onlarca soru ve cevabı bu belgede bulabilirsiniz. Linux'u hiç bilmeyenler kadar az bilgisi olanlara da yol gösteren ve genel sorunlara çözüm getiren bu kaynak, ASCII ve HTML formatlarında şu adresten temin edilebilir :

- *ftp.bilkent.edu.tr* `<ftp://ftp.bilkent.edu.tr/pub/TURKCE>`
- *www.linux.org.tr* `<http://www.linux.org.tr>`
- *www.ankara.edu.tr* `<http://www.ankara.edu.tr/start>`

Linux-NASIL dökümanları : tr-ldp@metu.edu.tr listesinde bulunan gönüllülerin çıkardığı bu güzel çalışma, çeşitli Linux-HOWTO dökümanlarının çevirisinden oluşuyor. Her bir dosya, özel bir konuyu ele alıp yoğun olarak bu konunun işlenmesinden ibaret olup Linux'u öğrenmek isteyen, fakat ana dili İngilizce olmayanlar için büyük fırsat niteliğini taşıyor. Yaklaşık 20 kadar çeviriden oluşan Linux-NASIL belgelerinin bulunabileceği adresler :

- *cclub.metu.edu.tr* `<ftp://ftp.cclub.metu.edu.tr/pub/tr-ldp>`
- *ftp.linux.org.tr* `<ftp://ftp.linux.org.tr>`
- *www.linux.org.tr* `<http://www.linux.org.tr>`

Yukarıdakilerin dışında, *ftp.metu.edu.tr* `<ftp.metu.edu.tr/sunsite/docs/howto>` adresinde, Linux işletim sistemi hakkında yabancı kaynaklı (İngilizce, Fransızca ve Almanca) dökümanlar ve çeviriler vardır. Burada sözkonusu İnternet adresinde neler bulunduğunu belirtmek imkansız. Birkaç HOWTO örneği vermek gerekirse:

- **Linux Installation-HOWTO** : Genel olarak Linux yüklemeyi anlatan bir kaynak

- **Linux Meta FAQ** : Genel olarak Linux hakkında bilgi veren bir kaynak
- **Linux Bootdisk HOWTO** : Boot ve root disketlerini, bunların yaratılmasını inceleyen bir kaynak
- **Linux Distribution HOWTO** : Hangi dağıtımların bulunduğunu ve nasıl elde edilebileceklerini ayrıntıları ile anlatan bir kaynak.
- **Linux Hardware Compatibility HOWTO** : Linux tarafından desteklenen donanımları özetleyen bir kaynak.
- **Linux INFO-SHEET** : Linux hakkında genel bilgi.
- **Linux Kernel HOWTO** : Çekirdek hakkında bilgi veren, nasıl derlenebileceğini, nereden bulunabileceğini ve çekirdeğin işlevleri hakkında bilgi veren bir kaynak.
- **Linux NET-3 HOWTO** : Linux'ın Network desteği hakkında bilgi veren kaynak.
- **Linux DOS2Linux mini-HOWTO** : DOS kullanıcılarına Linux hakkında kısa bilgiler veren sevimli kısa bir yardım dosyası
- **Linux LILO mini-HOWTO** : Çeşitli değişik LILO örnekleri
- **Linux Linux+DOS+Win95+OS2 mini-HOWTO** : Adı üzerinde aynı bilgisayarda hepsi birden.
- **Linux Linux+OS2+DOS mini-HOWTO** : Adı üzerinde
- **Linux Linux+Win95 mini-HOWTO** : Adı üzerinde
- **Linux Multiple Disks Layout mini-HOWTO** : Değişik diskler üzerinde çeşitli disk bölümleri kullanırken dikkat ettiği noktaları açıklayan bir kullanıcının hazırladığı döküman
- **Linux Reading List** : Linux ile çalışırken yararlanılabilecek kaynakları sıralayan bir döküman.
- **Linux Upgrade mini-HOWTO** : Bilgisayarda yüklü mevcut Linux dağıtımının nasıl güncellenebileceğini anlatan bir kaynak.
- **Linux Win95+Win+Linux mini-HOWTO** : Yani belirtmeme gerek var mı ? :)

Siteye eriştiğiniz zaman farkedeceğiniz zenginlik sizi şaşırtacaktır. Bu adresin yansısını Türkiye'de ODTÜ ve BİLKENT Üniversitelerinde yeralan servisler tutmaktadır.

Yurtdışı HOWTO adresi

- *sunsite.unc.edu* `<ftp://sunsite.unc.edu/pub/linux>`

Diğer LDP ve HOWTO yansıları

- *www.metu.edu.tr* `<http://www.metu.edu.tr:8080/ldp>`

- *www.ehb.itu.edu.tr* `<http://www.ehb.itu.edu.tr/ldp/>`
- *www.bilkent.edu.tr* `<http://www.bilkent.edu.tr/pub/linux>`

Bu arada, tüm ftp adresleri için 'pub' dizininden sonra gelen neredeyse tek ortak dizin *linux*'dur. Düşük hız sizin için sorun taşımıyorsa buradaki ftp ve www adreslerine bağımlı kalmak zorunda değilsiniz.

Linux Documentation Project (Linux döküman projesi) adı altında, 1990 yılından beri Linux dökümanları hazırlanıyor. Bu dökümanlar hem İnternet üzerinde bedava bulunmakta, hem de yazıcı ve kağıt masrafı gibi ince işlerle uğraşmak istemeyenler için kitap olarak da bastırılmaktadır. Bir dizi gönüllünün oluşturduğu bu dökümanlar, hem işletim sisteminin hızla gelişmesi, hem de çalışma kapsamının gittikçe genişlemesi nedeniyle hergün güncellenerek geliştirilmektedir. Linux için kullanım klavuzları (ve bu klavuzları hazırlayan gruba) LDP Linux Documentation Project adı verilir. Bu dökümanlar:

- *Linux Installation and Getting Started, Matt Welsh* : Yeni başlayanlar için mükemmel denilebilecek bir kitap. Hemen her türlü bilgiye yer verilmiş. Tabii ki tek eksiği Türkçe olmaması.
- *Linux System Administrators' Guide, Lars Wirzenius* : Linux İşletim Sistemini kurduktan sonra çok kullanıcılı ortama geçmek isteyenlere yol gösteren bir kaynak. Unutmayın ki siz de bir gün sistem görevlisi olabilirsiniz.
- *Linux Network Administrator's Guide, Olaf Kirch* : Linux altında belli bir mesafe katettikten sonra İnternet ve ağ yapısına girmek için bulunmaz bir kaynak. Gayet güzel hazırlanan bu kitapta hemen her istediğiniz bilgiyi bulabiliyorsunuz.
- *The Linux Kernel Hackers' Guide, Michael Johnson* : Çekirdek kaynak kodu birlikte verilen tek işletim sistemi olan Linux'un en büyük avantajı, bu canavarı aslında istediğiniz gibi yönetmenize izin vermesi. Linux Kernel Hacker's Guide, sistem programcılığı hakkında biraz altyapısı olanlar için.
- *Linux Users' Guide* : Temel kullanıcıya Linux'un özelliklerini detaylarıyla verir ve bu işletim sistemi üzerinde açlıktan ölmemesi için yol gösterir.
- *The Linux Kernel, David A. Rusling* : Çekirdeğin çalışma prensipleri.
- *The Linux Programmer's Guide* : Programcılara yönelik.

Linux Gazette ve Linux Journal : Linux Journal, *SSC* firması tarafından hazırlanıp dünyanın dört bir yanındaki üyelerine dağıtılıyor. Aylık çıkan bu dergi, profesyonel bir yayımcı kitlesi tarafından hazırlanıyor ve Linux ile uğraşanlar tarafından çok tutulan bir kaynak özelliğini taşıyor. Bunun yanında Linux Gazette, içerik olarak Linux Journal'a yakın, ama sadece İnternet üzerinden bedava dağıtılıyor.

Linux Gazette ve Linux Journal ana sayfası

- *www.ssc.com* `<http://www.ssc.com>`

Türkiye'deki yansıları

- *www.metu.edu.tr* `<http://www.metu.edu.tr/ldp>`
- *linux.org.tr* `<http://www.linux.org.tr>`
- *www.ehb.itu.edu.tr* `<http://www.ehb.itu.edu.tr>`

Haber Grupları : Usenet olarak da bilinen haber grupları, onbinlerce konu altında tartışma gruplarını biraraya getirir. Bu konu yoğunluğu içinde Linux için yaklaşık 10-15 kadar tartışma grubu açıktır. İlk olarak alt.os.linux ismi kullanılarak yapılan tartışmalar, Linux'un büyük bir hızla yayılmasıyla alt konulara ayrılmıştır.

- *comp.os.linux:* İnternet'in en çok okunan 20 haber grubundan biri olan comp.os.linux'ta her türlü soru ve cevapları bulabilirsiniz.
- *comp.os.linux.announce:* Linux işletim sistemi hakkında son haberler buraya yönlendiriliyor. Hemen her türlü yazılım, program hataları, Linux seminer ve konferans haberleri anında postalanıyor.
- *comp.os.linux.setup:* Linux kurulumu ve işletimi hakkında soru ve cevaplarını içerir.
- *comp.os.linux.hardware:* Linux'un desteklediği donanımlar ve bunlarla ilgili genel tartışma listesi
- *comp.os.linux.networking:* Linux'un ağ desteği (TCP/IP, SLIP, PPP vs.) konusunda tartışma listesi. Eğer bu grubu takip etmek istiyorsanızi, comp.os.linux.hardware'a da göz atın.
- *comp.os.linux.x:* X Window kurulumu ve kullanımı hakkında soru ve cevaplar. Linux'a özgü olmayan bir diğer X Window haber grubu da comp.windows.x.i386unix'tir.
- *comp.os.linux.advocacy:* Okumaktan en hoşlandığım haber grubu. Linux'un diğer işletim sistemleri üzerindeki üstünlükleri ve zayıflıkları burada tartışılıyor. Benim gibi MS-DOS sevmeyen birisi iseniz, burası sizi oldukça tatmin edecektir.
- *comp.os.linux.misc:* Diğer grupların konuları dışında kalan tartışmalar burada sürdürülüyor.

Tartışma Listeleri : E-posta atıp alabilme imkanınız varsa, haber gruplarının ayrı olarak listelere de üye olabilirsiniz. Listeler, tartışma gruplarına güzel bir alternatif teşkil ederler. Türkiye'de linux hakkında açılmış 4-5 tartışma listesi bulunmaktadır. Bunların arasında göze çarpan ve en çok kullanıcısı olan tartşma listesi Bilkent Üniversitesi'nde yeralan listedir. Bu listeye üye olmak için mesaj kısmında '*sub linux isim soyisim*' yazılı (isim ve soyisim yerine kendi isminizi ve soyadınızı kullanın) bir e-postayı *listserv@bilkent.edu.tr* adresine yollayın. En kısa zamanda listeye üyeliğiniz gerçekleşecektir.

RFC (Request For Comment) : Standartlar, tezler ve diğer resmiyet kokan pekçok bilgiye buradan erişilebilir. Başlangıç için önermiyorum. ODTÜ ftp adresinde, *ftp.metu.edu.tr/pub/rfc* `<ftp://ftp.metu.edu.tr/pub/documents/rfc>` dizini altında yansısı tutulmaktadır.

Aşağıda Türkiye'de ve yurtdışında çeşitli Linux kaynak dosyaları bulabileceğiniz ftp adresleri verilmiştir. Türkiye içinde ODTÜ, onlarca gigabyte yer kaplayan sunsite adresinin yansısının sadece bir kısmını tutabilmektedir. Bu yüzden zaman zaman yurtdışı adreslere bağlanmanızı hararetle salık veririm. *tsx-11.mit.edu* `<ftp://tsx-11.mit.edu>`, *sunsite.unc.edu* `<ftp://sunsite.unc.edu>` ve *nic.funet.fi* `<ftp://nic.funet.fi>` adresleri Linux'un ana adresleri olup en yeni yazılımlar önce bu adreslere gelir. İnternet trafiğini meşgul etmemek için size en yakın makinayı seçin.

Yurtiçi :

- *ftp.linux.org.tr* `<ftp://ftp.linux.org.tr>`
- *ftp.metu.edu.tr* `<ftp://ftp.metu.edu.tr/pub/linux>`
- *ftp.ehb.itu.edu.tr* `<ftp://ftp.ehb.itu.edu.tr/pub/linux>`
- *ftp.bilkent.edu.tr* `<ftp://bilkent.edu.tr/pub/linux>`
- *ftp.itu.edu.tr* `<ftp://ftp.itu.edu.tr/pub/linux>`
- *ftp.ankara.edu.tr* `<ftp://ftp.ankara.edu.tr/pub/linux>`

Yurtdışı :

- *tsx-11.mit.edu* `<ftp://tsx-11.mit.edu/pub/linux>`
- *sunsite.unc.edu* `<ftp://sunsite.unc.edu/pub/Linux>`
- *nic.funet.fi* `<ftp://nic.funet.fi/pub/OS/linux>`
- *wuarchive.wustl.edu* `<ftp://wuarchive.wustl.edu/systems/linux>`