

FreeBSD Ipuclari

FreeBSD isletim sistemi artik sadece bir isletim sistemi olmaktan cikip bir meta haline gelmis durumda. Her sekilde onu kullanip “Bunu da yapiyor ama nasil?” seklinde sormaktan bir turlu kurtulamasakta sonucta istedigimizi yaptirip yeni islemlere geciyor ve yeni yontemler ogreniyoruz.Iste bir islemi gerceklestirme yolunda karsınıza cikan bu tur “kritik nokta”lari bu yazi icinde birlestirmeye calistim.

Beastie ve Beerware

Arada bir FreeBSD sistemimin derinliklerinde acaba kacirdigim yeni bir dokuman veya eklenecek yeni bir klasor var mi diye gezinmek bana o kacirdigim yenilikleri yakalama firsati vermekte. Bu sefer `/usr/share/examples` bolumunden basladigim “goz gezdirme” islemi sirasinda gozume takilan ilk klasor `BSD_daemon` klasoru idi:

```
# ls /usr/share/examples/BSD_daemon
FreeBSD.pfa      beastie.eps      eps.patch
README          beastie.fig      poster.sh
```

Klasor icinde bulunan `README` dosyasi klasorde bulunan her bir dosyayi taker taker aciklamakta.Okumaya basladigimizda eski bir dost ve o cok sevdigimiz lisansi ile karsilasiyoruz. [Poul-Henning Kamp](#) ve tabiki onun unlu [Beerware](#) lisansi.

`README` bunun yaninda `fig2dev` yazilimini kullanarak nasil GIF ve PNG yaratilacagini gostermekte.Sistemimde olmadigi icin bu yazilim hakkında Google`da arama sonucu bunun `transfig` yaziliminin bir parcasi oldugunu gosterdi.Hemen yazilimi yukleyip resim dosyalarini olusturabilmek icin asagidaki komutlari verdim

```
# pkg_add -r transfig
# fig2dev -L gif -g '#f0f0f0' -t '#f0f0f0' beastie.fig beastie.gif
# fig2dev -L png beastie.fig beastie.png
```

Sonuc olarak cikan resimler Beastie`nin guzel basilmis goruntuleri idi.

Script`lere hareket kazandirmak

Daha sonra `dialog` alt klasorune gectim:

```
# ls /usr/share/examples/dialog
README      infobox      msgbox       textbox
checklist   inputbox     prgbox       treebox
ftreebox    menubox     radiolist    yesno
```

Daha önce, shell script`lere interaktif menüler eklemenize yardımcı olan `dialog` ile çalıştım. İnternet üzerinde çalışan birkaç örnek arasına da ancak sadece menübox oluşturmayı gösteren birkaç tanesini bulabildim, ki bu `dialog`'un sadece bir seçeneği idi. Her ne kadar `man dialog` detaylı şekilde konuyu açıklasa bile yeni bir şey öğrenirken çalışan bir örnek üzerinde oynayarak kavramak daha mantıklı gelmekte.

Yardımcı olması açısından aşağıya birkaç dialog kutusu örneği ekleyelim. Bunlar Bourne shell script`leri oldukları için kolay okunabilmekte ve zahmetsizce çalışabilmekte. Her script`i denemek için ilk yapılması gereken işlem onların çalıştırılabilir hale getirilmesi olacaktır:

```
# chmod +x [a-z]*
```

Dikkat edin `chmod` komutunu kullanırken dosya isimleri yerine wildcard kullanarak çalıştırılabilir (+x) hale getirdim. Böylece okumak için kullanacağım `README` dosyası kendi izinlerini değiştirmeden çalıştırılabilir hale getirmek istediğim dosyaların her biri ufak harf ile yazılmış `[a-z]` ve sona getirdiğim `*` wildcard`i tüm bu harflerdeki dosyaların izinlerinin değiştirilmesini onayladı.

Ayrıca script`ler `/usr/share/examples` buldukları için tam bir yol belirtmeniz gerekecektir. Bunun yanında bu klasöre girip script isminden önce `./` koymanız dosyayı çalıştırmanızı sağlayacaktır:

```
# cd /usr/share/examples  
# ./yesno
```

Acil Durumlarda

Belli bir öğrenme süreci geçirdikten sonra, sisteminizi etkileyecek büyük hatalar yapmak çok az görülen bir durumdur. Buna rağmen böyle bir durumda acil çıkış kapılarından haberdar olmak işinize yarayacaktır.

Eğer FreeBSD 5.2 veya daha yüksek versiyonunu kullanıyorsanız `man rescue` ve sonra da `ls /rescue` dosya ve klasörlerine bir göz atın. `/bin` veya `/sbin` klasörlerine ulaşamadığımız ve şu unlu “Enter full pathname of shell or RETURN for /bin/sh:” hatası karşımıza çıktığında sistemimizi onarmak için gerekli olan dosyaların birer “temiz” kopyalarının burada saklanması gerçekten iç rahatlatıcı. `Man` dosyasını dikkatlice okursanız detaylı şekilde bu klasörü nasıl kullanabileceğinizi adım adım öğrenebilirsiniz.

Daha önce hiç bir konfigürasyon dosyasını yanlış düzenlediniz veya sildiniz mi? Eğer bu dosya `/etc` içinde ise tamamen kaybolmuş sayılmaz. Sisteminiz bu dosyaların birer kopyalarını `/usr/share/examples/etc/` altında tutmakta.

Son olarak `rm` komutunu kullanmak istediginiz en son yer `/dev` klasorudur. Hic bir klasorkeki dosyalarin diger dosyalardan farkli olarak iki tane daha aciklama hanesine sahip oldugunu farkettiler mi? Su iki dosyayi karsilastiralim:

```
% ls /etc/rc.conf
-rw-r--r--  1 root  wheel  763 Dec  6 13:59 /etc/rc.conf
          ^^^

% ls /dev/fd0
crw-r-----  1 root  operator   4,  10 Dec 29 08:44 /dev/fd0
          ^^^^^^^
```

Dosya boyutunu byte olarak gosterme yerine aygit dosyasi bir buyuk numarayi takip eden ufak numara olarak aciklamakta. Bu numaralar aygitin cesidini belirterek kernel`a aygit ile nasil iletisime gecmesi gerektiiginin yolunu gostermektedir. Bu demek oluyor ki eger `/dev/fd0` dosyasini silerseniz o her zaman hayat kurtaran `touch /dev/fd0` komutu pek bi rise yaramayacaktır. Bunun yerine asagidaki sekilde `mknod` komutunu kullanabilirsiniz:

```
# mknod fd0 c 4 10
```

Burada `c` bunun bir karakter aygiti oldugunu belirtmekte. Dikkat ettiyseniz `c` daha onceki izin bolumunun hemen yaninda yer almakta idi. Peki bu ne anlama gelmekte? Eger boyle bir dosyayi sildiyseniz yeniden yaratmak icin tam olarak siralamayi ve kullanilan buyuk ve kucuk sayiyi bilmek zorundasiniz demektir. Eger elinizin altinda baska bir FreeBSD sisteminiz yok ise `/dev` klasorunun bir kopyasini almak veya ozelliklerini `ls -l /dev` komutu ile kaydetmek yerine bir hareket olacaktır.

Ne Okumali?

Gun gectikce artan [FreeBSD dokumanlari](#) `/usr/share/doc/en/articles` klasoru altinda bulunabilmekte.

Port`larimizin guncel kalmasini hepimiz `cvsup ports-supfile` komutu ile sagliyoruz peki ya dokumantasyonlar? Iste onlari da su sekilde guncel tutacagiz. CVS `supfile` dosyaniza asagidaki parametrenin bulundugundan emin olun:

```
doc-all tag=.
```

Kaynak dosyalari SGML oldugu icin bu port`un onlari HTML`e cevirmesine ihtiyaciniz olacak:

```
# pkg_add -r docproj-nojadetex
```

Bunun için sisteminizi [güncel tutmak için hazırlayacağınız](#) script`lerde aşağıdaki gibi bir kısım eklemeniz bu işi kolaylaştıracaktır:

```
echo "Updating docs"
cd /usr/doc
cp Makefile.orig Makefile
make install > /dev/null
```

Bu klasör içindeki yazıların bazıları yeni başlayanlar hatta FreeBSD ile tanışmamış kişiler için bile okunabilir derecede hazırlanmıştır. `/usr/share/doc/en/articles` klasöründe `new-users` ve `freebsd-questions` dokümanlarına göz atabilirsiniz. Eğer bu yazılara siz de katkıda bulunmak istiyorsanız fakat kod`lama yapamıyorsanız `contributing` bölümü tam size göre.

Hic BSD`nizin Unix aile ağacında nerede bulunduğunu görsel olarak görmek istediniz mi? Hic beklemeye gerek yok hemen `more /usr/share/misc/bsd-family-tree` komutunu deneyin. Peki yazılımınız için bir man dosyası da siz mi yaratmak istiyorsunuz? `/usr/share/misc` klasörü içinde bulunan `man.template` ve `mdoc.template` dosyaları işinizi çok kolaylaştıracaktır. Daha iyisi, `/usr/share/examples/mdoc` klasörü u cadet tam dokümantasyonu içeren man dosyasını örneklemektedir.

Görsel `rc.conf`

[Freshmeat](#) veya [Freshports](#)`ta gezinirken yeni port`ları görüp denemek çoğu zaman sisteminize yeni özellikler katmak için idealdir. Bu sayfalarda gezinirken gözümüze çarpan bir port`u beraber deneyelim. İsmi [thefish](#) olan bu port`u yüklemeye şöyle başlayalım:

```
# pkg_add -r thefish
```

Bu port`u çalıştırdığımda port kendisini, sizin de `/stand/sysinstall``daki fareyi kullanmadığınız sadece ok tuşları ile hareket edebildiğiniz ekrandan hatırladığınız gibi `ncurses` versiyonunda çalıştırmakta. Üzerinde X yüklü olmayan sistemler için harika olan bu sistem GUI (Gnome-Kde vs.) kullanan kişiler için kullanışsız olmaktadır..

GUI versiyonu için bu port`u sistemden kaldırıp aşağıdaki versiyonunu kurmam gerekti:

```
# cd /usr/ports/sysutils/thefish
# make install clean
```

Bu port bana aşağıdaki seçenekleri vermekte:

```
[ ] CONSOLE      Disable X11 support
[x] GTK          Use GTK+ for the X11 interface
[ ] QT          Use Qt for the X11 interface
```

GTK+ ve QT kullanilabilecek widget set`leri arasinda. *Widget set`*leri bir pencerenin nasil gorunecegini, ornegin buyutuldugunde ufaltildiginda nasil olacagini, kontrol ederler. Eger Gnome kullaniyorsaniz GTK+, KDE kullaniyorsaniz QT widget set`ini kullaniyorsunuz demektir. Ben QT`yi tercih ettigim icin onu sectim. Eger *thefish`*in sayfasindaki ekran görüntülerine bakarsaniz QT menüsündeki butonların GTK+ menüsündekilerden daha farklı olduklarını göreceksiniz.

Bir kere *thefish* port`unu yüklediğinizde artık gerisi sizin yapacağınız konfigürasyonlara kalmakta.

Fikir mi Degistirdiniz?

Bir port`u standart gelen seçenekleri ile kurdüğünüz fakat daha sonra fikrinizi değiştirip bu seçenekleri güncellemek istediğiniz oldu mu? Örneğin QT ile *thefish* port`unu yükledikten sonra GTK+`ya çevirmek istediğinizi düşünelim. Böyle bir durumda port klasoru içinde *make* komutunu çalıştırdığınızda artık size seçim sansi vermediğini farkedeceksiniz. Bunun yerine doğrudan eski seçtiğiniz parametre ile yeniden yükleme yapmaya çalışacaktır.

Biraz araştırmadan sonra sunu buldum:

```
# more /var/db/ports/thefish/options

This file generated by 'make config'.
# No user-servicable parts inside!
# Options for thefish-0.6.3
_OPTIONS_READ=thefish-0.6.3
WITHOUT_CONSOLE=true
WITHOUT_GTK=true
WITH_QT=true
```

Bu durumda bu dosyayı kaldırabilir veya port klasoru için *make config* komutunu verirseniz port size tekrar seçme sansini verecektir.

RSS

Hergün takip ettiğiniz sayfalar vardır. RSS`nin farkında olmama rağmen bunu denemeye hiç şansım olmamıştı. Port`lar arasında ufak bir araştırma birçok RSS istemcisinin bulunduğunu göstermeye yetti. GUI bir istemci istediğimden dolayı */usr/ports/www/akregator* en başarılı olanlarından birisi olarak karşına çıktı.

Diğer RSS istemcilerinin önceden belirlenmiş listeleri olduğu için yerleri değiştirilmiş RSS sayfalarının bulunması durumunda tek tek sayfaları gezip RSS adreslerini toplamam gerekiyordu.

Bunların arasında [akregator](#) yazılımı hem RSS kullanımı hem de eklenmesi konusunda en kolay olanı idi. Normalde bu tür istemcilerde tam RSS yolunu yazmazsanız istemci hata verecektir. Fakat bu port'ta [slashdot.org](#) yazdığım da otomatik olarak RSS bölümü bulunacak ve orneğin <http://slashdot.org/index.rss> olarak saklanacaktır.

Son olarak

Umarım bu yazı size FreeBSD sisteminizin derinlikleri hakkında bir başlangıç olmuştur. Eğer siz de bu şekilde ufak ipuçları yakaladıysanız ve diğer kullanıcıların da bundan yararlanmasını isterseniz her türlü ipucu bilginizi hem Türkçe hem de İngilizce olarak hazırladığımız <http://ipucu.enderunix.org> adresine gönderebilirsiniz.

Ozgur Ozdemirci
ozgur@enderunix.org
<http://www.enderunix.org>
<http://www.enderunix.org/ozgur/blog>

[Dru Lavigne](#)'nin "FreeBSD Tips and Tricks For 2005" yazısından derlenmiştir. Orjinal metne http://www.onlamp.com/pub/a/bsd/2005/02/17/FreeBSD_Basics.html adresinden ulaşabilirsiniz.