FreeBSD Erişim Kontrol Listeleri

Unix dosya izinleri çok esnek ve nerdeyse bütün erişim problemlerini çözebiliyorlar.Peki ya çözemedikleri?Her dosyayı başka bir kullanıcı ile paylaşmak için yeni bir grup mu oluşturmak zorundasınız?Devamlı `sudo` ve `calife` ile root olup komut satırında güvenlik riski oluşturmak yerine neden root olmayan bazı kullanıcıların sadece root tarafından değiştirilebilen dosyalar üzerinde değişiklik yapabilmelerine olanak sağlamıyoruz?

ACL adini verdiğimiz Access Control Lists (Erişim Kontrol Listeleri) bu sorunları çözüyor.Bunlar standart Unix /user/group/other gibi hazır izinlerden daha fazla esneklik sağlıyorlar.Su ana kadar ACL`ler IRIX , Solaris (ve Windows NT) gibi ticari Unix çeşitlerinde bulunmakta idi.Su anda ise TrustedBSD sayesinden artık FreeBSD 5.0 surumu ve sonrasında da kullanılabilmekte.

ACL Kullanımı

ACL`ler dosya sistem superblogu`nda bulunan ve `tunefs` ile ayarlanabilen bir seçenek yardımı ile kullanılır hale getirilebiliniz.

Superblok seçeneğinin ayarlanması

`tunefs` komutu sadece okunabilir veya mount edilmemiş dosya sistemleri üzerinde kullanılabilinir.Tabiki bunu sağlayabilmek ve `sbin/tunefs -a enable /fs` komutunu kullanabilmek için (fs burada dosya sistemi bağlantı noktasını belirtmekte) için tek-kullanıcı (single user) mod`unu kullanarak boot etmeniz gerekiyor.(Eğer makinenin /, /usr vs. klasörlerine giriş hakkiniz yoksa ki bu daha çok co-located dediğimiz makinelerde oluyor, tunefs komutunu /etc/rc dosyasına ekleyerek ACL`leri bir sonraki boot`ta kullanılabilir hale getirebilirsiniz.

Eğer UFS2 dosya sistemini kullanıyorsanız isiniz daha da kolay.ACL`ler `GENERIC` kernel`de hali hazırda bulunan `options UFS_ACL` parametresi ile kullanılabilir durumdadır.Bilgisayarınızı tekrar başlatın.Hepsi bu kadar.Eğer UFS1 kullanıyorsanız isiniz biraz daha zor olacak.

UFS1 için ek konfigürasyon

Diğer birçok Freebsd 5.0 kullanıcısı gibi UFS1 kullanıyorsanız biraz daha uğraşmanız gerekecek.(FreeBSD 5.1 ve sonrası artık UFS2 dosya sistemi ile gelmekte).ACL`ler UFS` sisteminde bulunmayan genişletilmiş özniteliklerin üzerine kurulmuş durumdadır.genişletilmiş öznitelikleri kullanabilmek için kernel`inize `options UFS_EXTATTR` ve `options UFS_EXTATTR_AUTOSTART` parametrelerini ekleyip, kernel`ı tekrar derlemeniz gerekmekte.Kernel`ı `make depend && make && make install` komutları derliyoruz.Sistemi tekrar başlatmadan önce dosya sistemleri için genişletilmiş öznitelikleri çalıştırmamız gerekiyor.

Örneğin, `/var` dosya sisteminde öznitelikleri çalışır duruma getirebilmek için:

```
% mkdir -p /var/.attribute/system
% cd /var/.attribute/system
% extattrctl initattr -p /var 388 posix1e.acl_access
% extattrctl initattr -p /var 388 posix1e.acl_default
```

burada yapmanız gereken tek şey `/var` bağlantı noktasını istediğiniz dosya sistemi ismi ile değiştirmek.Bunu yaptıktan sonra bilgisayarı yeniden başlatırsanız artık ACL`ler kullanıma hazır olacaktır.

ACL kullanımı

ACL`leri kullanılabilir duruma gitirdiniz.Peki ya simdi?

ACL`lere kısa bir bakış

ACL`leri görmek çok kolay. `ls –l` komutunu kullanıp liste aldığınızda ACL`ler + işareti taşıyacaktır.

```
-rw-rw-r--+ 1 rob  rob  0 Apr 19 17:27 acl-test
```

ACL`I görmek için `getfacl` komutunu kullanabiliriz:

```
$ getfacl acl-test
#file:acl-test
#owner:1000
#group:1000
user::rw-
user:nobody:rw-
group::r--
group:wheel:rw-
mask::rw-
other::r--
```

`User::`, `group::`, ve `other::` bölümleri standart UNIX dosya ve izin sisteminde kurulumla gelen ACL`ler olarak bulunsa da `nobody` ve `wheel` daha sonradan katılan izin sistemleridir.Bunların hepsi belirli kullanıcılar ve guruplar için izinleri belirlemekte.

ACL ekleme ve çıkarma

`Setfacl` komutu ACL`leri eklemek, silmek ve değiştirmek için kullanılır.birçok seçeneği olmasına rağmen birkaç tanesini bilmek ACL`lerinizi kontrol etmek için yeterli olacaktır.

İlk bilinmesi gereken şey ACL`lerin `getfacl` tarafından gösterilen şekilde belirlenmesi gerektiğidir.Örnek olarak `acl-test` üzerinde bulunan ACL`I silip yeni bir tane yaratalım:

```
$ setfacl -b acl-test
$ setfacl -m user:nobody:rw-,group:wheel:rw- acl-test
```

`-b` seçeneği standart, kullanıcı gurup ve other satırları dışındaki bütün ACL`leri kaldırır. `-m` seçeneği girilen yeni parametre ile ACL`yi değiştirir.Kullanılan parametreler kısaltılabilir.Örneğin `u:nobody:rw-,g:wheel:rw-`.Ayrıca `setfacl`yi hali hazırda gelen izinleri değiştirmek için kullanabilirsiniz.Örneğin `user::rw-` komutu bir dosya için `chmod u=rw` komutunu kullanmaya eşdeğerdir.

ACL`leri kaldırma ise nerdeyse aynidir. `setfacl -x u:nobody:rw-,g:wheel:rw-` komutu bulunan ACL`yi kaldırır. `–M ve –X` seçenekleri ise ufak harfli yazılışları ile ayni işlemi bu sefer bir dosyadan okuyarak yaparlar.

`acl-test` dosyasını düşünelim:

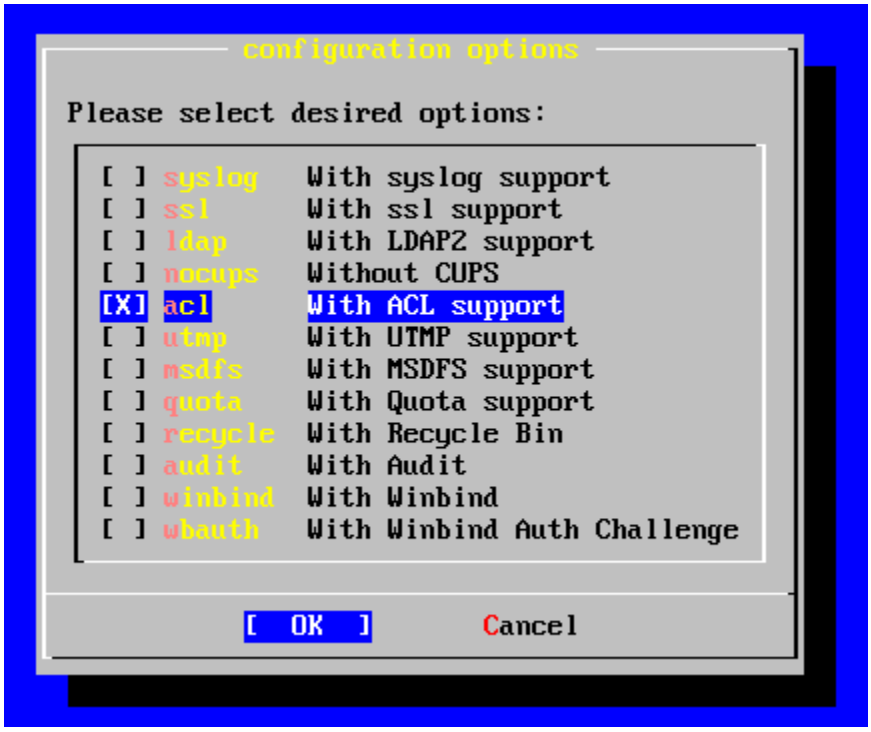
```
$ cat test-acl-list
u:nobody:rw-
# açıklama
g:wheel:rw-
$ setfacl -X test-acl-list acl-test
$ getfacl acl-test
#file:acl-test
#owner:1000
#group:1000
user::rw-
group::r--
mask::r--
other::r--
```

ACL`ler ve diğer Unix araçları

Maalesef çoğu Unix aracının ACL desteği bulunmamakta.Örneğin `tar` ACL yedeği alamaz.FreeBSD`de NFS ise ACL`leri görmezden gelir.Fakat `tar` veya `dump` ile yapılan tüm system UFS1 yedeklemeleri `.attribute` klasörlerinin de yedeklerini alacaktır.Ayrıca FreeBSD`nin `dump` komutu ACL`leri de içerir şekilde UFS2 dosya sistemini anlamaktadır. `Archivers/star` port`u da ACL`leri desteklemektedir.

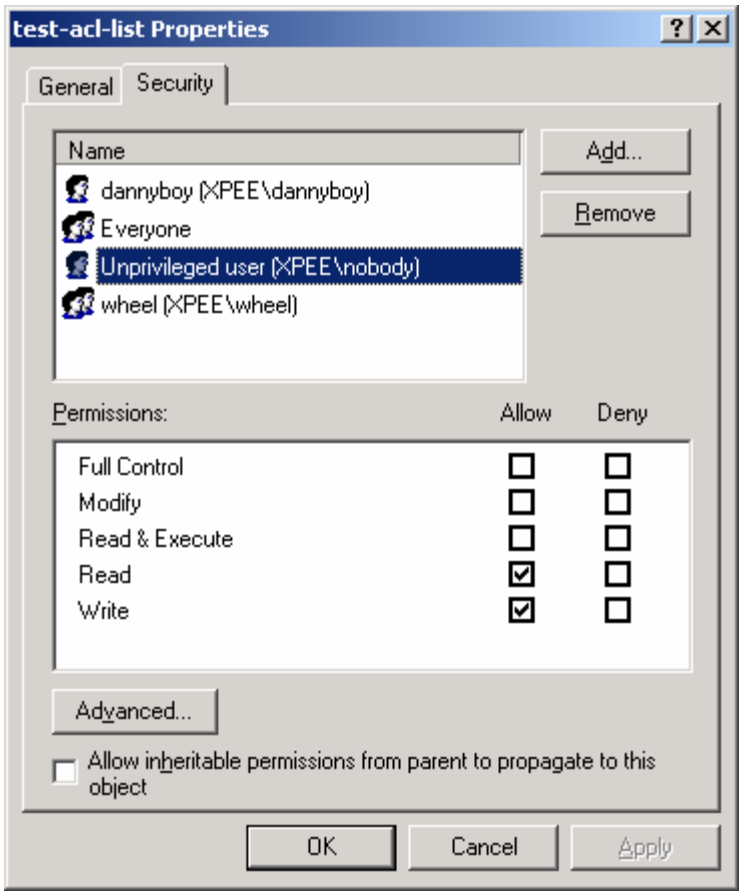
Samba ve Windows altında ACL kullanımı

Eğer Samba`yi ACL desteği ile derlerseniz, Samba tarafından paylaşılan dosyaları Windows ACL araçları ile değiştirebiliriniz. `/usr/ports/net/samba` klasörüne girip `` `make WITH_ACL_SUPPORT && make install` `` komutunu vererek bunu kolayca yapabilirsiniz.Kurulum sırasında da ayrıca aşağıda görüldüğü gibi diyalogdan istediğiniz özellikleri seçme sansınız olacaktır:



Resim1: Samba konfigürasyon seçenekleri

Samba`yi kurup çalıştırdıktan sonra ACL kullanılan dosya sisteminde bulunan bir dosya üzerinde gelip `Özellikler` ve daha sonra `Güvenlik` bölümünde ACL`leri inceleyebilir ve sanki bir Windows sunucusundaymis gibi bunları ayarlayabilirsiniz.



Resim 2. FreeBSD üzerinde bulunan bir dosyanın Windows 2000 istemcisi üzerinden ACL`lerinin ayarlanması.

Eğer ACL eksikliği yüzünden hala Windows 2000 sunucu kullanmaya devam ediyorsanız bundan sonra ciddi olarak FreeBSD üzerinde Samba kurup bunu kullanmaya başlamayı düşünebilirsiniz.

ACL`lerin çoğaltılması

Daha gelişmiş bir örnekle devam edelim.Örneğin `cool_widgets` klasörünüzü sadece Bob isimli kullanıcıya açmak istiyorsunuz.yapmanız gereken tek şey yeni bir ACL eklemek.Yeni eklenen dosyalar otomatik olarak içinde bulundukları bu klasörün ACL izinlerini almayacaklardır.Almalarını sağlamak üzere klasör için varsayılan ACL değerini belirlemeniz gerekiyor. `-d` seçeneğini `getfacl` veya `Setfacl` komutları ile kullanarak klasörün kendisinde değil fakat içeriğindeki varsayılan ACL değerini belirleyebiliriz.

```
$ mkdir cool_widgets
$ chmod o-rwx cool_widgets
$ ls -l
...
```

```
drwxr-x---  2 rob  rob   512 Apr 19 21:21 cool_widgets
...
$ getfacl -d cool_widgets
#file:cool_widgets
#owner:1000
#group:1000
```

Simdi ise varsayılan ACL değerini belirleyelim:

```
$ setfacl -d -m u:bob:rw- cool_widgets
setfacl: acl_calc_mask() failed: Invalid argument
setfacl: failed to set ACL mask on cool_widgets
```

Hmm burada bir sorunumuz var.varsayılan ACL`ler normal ACL`ler gibi çalışmamakta.varsayılan ACL`de `user::`, `group::`, ve `other::` girişlerini yapmadan önce belirli girişler yapamıyoruz.

```
$ setfacl -d -m u::rw-,g::r--,o::---,u:bob:rw- cool_widgets
$ setfacl -m u:bob:r-x cool_widgets
```

`Bob` için varsayılan olmayan `r-x` girişine dikkat edin: varsayılan ACL dizin içinde yaratılacak dosyaları eklerken dizinin kendisini etkilemez. `u:bob:rw-` ACL kuralı gurup kullanılmadan `cool_widgets` içinde yaratılacak tüm dosyalara uygulanacaktır.Daha sonra bu ACL`I kaldırmak isterseniz `setfacl` için kullanacağınız `–k` parametresi varsayılan ACL`ler için ise yararken `–b` seçeneği dosya ACL`leri için uygulanacaktır.

Sonuç

ACL`ler normal Unix izin sistemi ile çözülmesi imkansız olan erişim kontrol listeleri problemlerini kolayca çözüyorlar.Yeni gurup yaratma ve root olarak çok fazla işlem yapılmasını engelleyerek daha güvenli sunucular yaratıyor.

Özgür Özdemircili
www.siberhayat.com – siber yaşamlarımızın bir yansıması –

Sorularınız için : dionypheles@gmx.net

Kaynaklar

Daniel Haris`in FreeBSD Access Control Lists yazısından çevrilmiştir.

Yazar`in sayfasına http://www.oreillynet.com/pub/au/1265 , Orijinal metne ise http://www.onlamp.com/pub/a/bsd/2003/08/14/freebsd_acls.html adresinden ulaşabilirsiniz.