Bsd Güvenlik Duvarları: IPFW

FreeBSD sisteminiz standart olarak ip paketlerini incelemek için kullanabileceğiniz iki program ile geliyor.İp paketlerini incelemek için her ikisinin de kendine özel yöntemleri olduğu için her ikisinin de kullanılışını göstermeye çalışacağım.Sadece bir tanesini kullanabileceğiniz için `ipfw` ile başlayacağım ve daha sonra `ipfilter `a geçeneğim Her zamanki gibi makalede İngilizce terimlerine karşılık olarak kullandığım kelimeleri belirterek başlayalım:

```
rule set =  kurallar zinciri

firewall =  güvenlik duvarı

log      =  günlük
```

`Ipfw`yi kullanmaya başlamadan önce kernel konfigürasyon dosyanıza birkaç dosya ekleyip, kernel`ınızı tekrar derlemeniz gerekecek.Eğer kernel derleme konusunda takıldığınız yerler varsa FreeBSD el kitabinin kernel derleme hakkındaki bölümüne göz atabilirsiniz.

`Ipfw` ile kullanılabilecek birçok seçenek bulunmakta.Bu yüzden LINT dosyamızı inceleyerek başlayalım.Eğer LINT dosyanız `/usr/src/sys/i386/conf` içinde bululuyorsa `MAKE LINT` komutu ile bunu yaratabilirsiniz.Doğru bolumu bulmak için “/” işaretini kullanarak bir arama yaparak başlayalım:

```
cd /usr/src/sys/i386/conf
more LINT
/IPFIREWALL

# IPFIREWALL enables support for IP firewall construction,
# in conjunction with the 'ipfw' program.  IPFIREWALL_VERBOSE
# sends logged packets to the system logger.
# IPFIREWALL_VERBOSE_LIMIT limits the number of times a
# matching entry can be logged.
#
# WARNING:  IPFIREWALL defaults to a policy of "deny ip
# from any to any" and if you do not add other rules during
# startup to allow access, YOU WILL LOCK YOURSELF OUT.  It
# is suggested that you set firewall_type=open in /etc/rc.conf
# when first enabling this feature, then refining the firewall
# rules in /etc/rc.firewall after you've tested that the new
# kernel feature works properly.
#
# IPFIREWALL_DEFAULT_TO_ACCEPT causes the default rule (at boot)
# to allow everything.  Use with care, if a cracker can crash
# your firewall machine, they can get to your protected machines.
# However, if you are using it as an as-needed filter for
# specific problems as they arise, then this may be for you.
# Changing the default to 'allow' means that you won't get stuck
# if the kernel and /sbin/ipfw binary get out of sync.
```

Minimum olarak `ipfw`yi kullanabilmek için, kernel`a her ip paketini inceleyip belirtilen bir kurallar zinciri ile karşılaştırmasını söyleyen `IPFIREWALL` seçeneğini kernel`ınızda bulundurmanız gerekiyor.Doğabilecek hatalar ve sorunlar için her zaman günlük (log) tutma seçeneğini ise `IPFIREWALL_VERBOSE` ile belirleyebiliyoruz.Ayrıca Kernel`a ve dolayısıyla işlemciye binen yükü hafifletmek ve olabilecek Denial of Service saldırılarını engellemek için kernel tarafından tutulacak paket log`larını da sınırlamanız gerekecek.Bunu da `IPFIREWALL_VERBOSE_LIMIT` seçeneği ile yapacağız.

`ipfw` standart olarak, kurallar zincirinizde belirtip izin verdiğiniz paketler dışındaki tüm paketleri düşürmeye programlanmış olarak gelecektir.Ben de size hangi paketlerin geçmenizi istediğiniz konusunda özgürlük sağlaması açısından bu seçeneği beğenenler arasındayım. Ne de olsa istediğim paketlerin kernel tarafından düşürüldüğünü fark ettiğimde her zaman için kurallar zinciri dosyam ile oynayıp bu paketlere izin verebilirim.Bunun için `IPFIREWALL_DEFAULT_TO_ACCEPT` seçeneğini ekleyerek standart olarak gelen `IPFIREWALL_DEFAULT_TO_DENY` seçeneğini değiştirmeyeceğim.

```
# IPDIVERT enables the divert IP sockets, used
# by ''ipfw divert''
```

Bu seçenek `natd` ile birlikte kullanılmak için tasarlanmış.Ben ise sadece tek bir makineyi korumak için programı kullanacağımdan bu seçeneği eklemeyeceğim.

```
# IPSTEALTH enables code to support stealth forwarding
# (i.e., forwarding packets without touching the ttl).
# This can be useful to hide firewalls from traceroute
# and similar tools.
```

Bu ilginç bir seçenek gibi görünüyor.Bunu da konfigürasyon dosyasına ekleyip güvenlik duvarımı test ettiğimde nasıl çalıştığına bakacağım.

```
# Statically Link in accept filters
options            ACCEPT_FILTER_DATA
options            ACCEPT_FILTER_HTTP
```

Bu makinede veya iç ağımda herhangi bir web sunucu çalıştırmadığımdan dolayı bu iki seçeneği eklemiyorum.

```
# The following options add sysctl variables for controlling
# how certain TCP packets are handled.
#
# TCP_DROP_SYNFIN adds support for ignoring TCP packets with
# SYN+FIN. This prevents nmap et al. from identifying the
# TCP/IP stack, but breaks support for RFC1644 extensions
# and is not recommended for web servers.
#
# TCP_RESTRICT_RST adds support for blocking the emission
# of TCP RST packets. This is useful on systems which are
# exposed to SYN floods (e.g. IRC servers) or any system
# which one does not want to be easily portscannable.
```

Güvenlik duvarı testim sırasında kullanabileceğim bu iki parametreyi de ekliyorum:

```
# ICMP_BANDLIM enables icmp error response bandwidth
# limiting.   You typically want this option as it will
# help protect the machine from D.O.S. packet attacks.
#
options         ICMP_BANDLIM
```

Bu seçenek ise standart kernel ile hazır olarak gelmekte.

```
# DUMMYNET enables the "dummynet" bandwidth limiter.
# You need IPFIREWALL as well. See the dummynet(4)
# manpage for more info. BRIDGE enables bridging between
# ethernet cards -- see bridge(4).
```

Bu bilgisayarda herhangi bir trafik düzenlemesi yapmayacağım için yine yukarıdaki seçenekleri kernel`a eklemiyorum.

Kernel`ı derlemeye geçmeden önce aşağıdaki satırları kernel konfigürasyon dosyasına ekliyorum.

```
#To enable IPFW with default deny all packets
options   IPFIREWALL
options   IPFIREWALL_VERBOSE
options   IPFIREWALL_VERBOSE_LIMIT=10

#To hide firewall from traceroute
options   IPSTEALTH

#To hide from nmap, remove if create web server
options   TCP_DROP_SYNFIN

#To hide from portscans
options   TCP_RESTRICT_RST
```

Kernel`im yeniden derlenirken ben de `/etc/rc.conf` dosyasına ekleyeceğim parametrelere göz atabilirim.Tekrar arama için “/” kullanacağım:

```
man rc.conf
/firewall

firewall_enable
   (bool) Set to NO if you do not want have firewall rules
   loaded at startup, or YES if you do.  If set to YES, and
   the kernel was not built with IPFIREWALL, the ipfw kernel
   module will be loaded.  See also ipfilter_enable.

firewall_script
   (str) If you want to run a firewall script other than
   /etc/rc.firewall, set this variable to the full path to
   that script.

firewall_type
   (str) Names the firewall type from the selection in
```

```
   /etc/rc.firewall, or the file which contains the local
   firewall ruleset.  Valid selections from /etc/rc.firewall,
   are ''open'' - unrestricted IP access; ''closed'' - all IP
   services disabled, except via lo0; ''client'' - basic pro-
   tection for a workstation; ''simple'' - basic protection
   for a LAN.  If a filename is specified, the full path must
   be given.
```

Güvenlik duvarı kurallar zincirimin makine açılışımda yüklenmesini işediğimden dolayı `firewall_enable` seçeneğini rc.conf dosyama `firewall_enable=YES` olarak ekleyeceğim.Kendi kurallar zincirimi yaratacağım için bu dosyanın yerini de rc.conf dosyasında belirtmem gerekecek.Bunun için `firewall_type` seçeneğini kullanacağım.

```
firewall_quiet
   (bool) Set to YES to disable the display of ipfw rules on
   the console during boot.
```

Bu parametre ise, kurallar zincirinizde belirttiğiniz her uygulanan kuralın sistem açılısında gösterilmesi açısından YES kullanarak onaylanacak bir seçenek.

```
firewall_logging
   (bool) Set to YES to enable ipfw event logging.  This is
   equivalent to the IPFIREWALL_VERBOSE kernel option.

tcp_extensions
   (bool) Set to NO by default.  Setting this to YES enables
   certain TCP options as described by RFC 1323.  If you have
   problems with connections randomly hanging or other weird
   behavior of such nature, you might try setting this back to
   NO and seeing if that helps.  Some hardware/software out
   there is known to be broken with respect to these options.

log_in_vain   (bool) Set to NO by default.  Setting to YES
   will enable logging of connection attempts to ports that have
   no listening socket on them.

tcp_keepalive
   (bool) Set to YES by default.  Setting to NO will disable
   probing idle TCP connections to verify that the peer is
   still up and reachable.

tcp_drop_synfin
   (bool) Set to NO by default.  Setting to YES will cause the
   kernel to ignore TCP frames that have both the SYN and FIN
   flags set.  This prevents OS fingerprinting, but may break
   some legitimate applications.  This option is only avail-
   able if the kernel was built with the TCP_DROP_SYNFIN op-
   tion.
```

`TCP_DROP_SYNFIN` seçeneğini kernel`ıma eklediğim için bu seçeneği YES olarak değiştireceğim.İleride bir web sunucu eklemeyi düşünürsem diye yanına da bu seçeneği kaldırmam gerekeceğini belirten ufak bir notta alıyorum.

```
tcp_restrict_rst
   (bool) Set to NO by default.  Setting to YES will cause the
   kernel to refrain from emitting TCP RST frames in response
   to invalid TCP packets (e.g., frames destined for closed
   ports).  This option is only available if the kernel was
   built with the TCP_RESTRICT_RST option.

icmp_drop_redirect
   (bool) Set to NO by default.  Setting to YES will cause the
   kernel to ignore ICMP REDIRECT packets.

icmp_log_redirect
   (bool) Set to NO by default.  Setting to YES will cause the
   kernel to log ICMP REDIRECT packets.  Note that the log
   messages are not rate-limited, so this option should only
   be used for troubleshooting your own network.
```

Sonuçlarak tüm bu yazdıklarım sonunda karşıma şöyle bir rc.conf dosyası çıkıyor:

```
#required for ipfw support
firewall_enable="YES"
firewall_script="/etc/rc.firewall"
firewall_type="/etc/ipfw.rules"
firewall_quiet="NO"    #change to YES once happy with rules
firewall_logging_enable="YES"

#extra firewalling options
log_in_vain="YES"
tcp_drop_synfin="YES"  #change to NO if create webserver
tcp_restrict_rst="YES"
icmp_drop_redirect="YES"
```

Yeni kernel`im ile makineyi yeniden başlatmadan önce son birkaç not daha. `LINT` dosyası “YOU WILL LOCK YOURSELF OUT." “GERÇEKTEN BÜTÜN TRAFİĞİNİZİ ENGELLEYECEKSİNİZ” seklinde bir uyarı verirken GERÇEKTEN bunu kastetmektedir.Yeni bir kurallar zinciri yaratıp geçmesini istediğim paketleri belirlemedikten sonra sisteme hiçbir paketin girmesi veya sistemden çıkması mümkün olmayacaktır.Eğer internetten herhangi bir şey çekmek veya e-postalarınızı kontrol etmek isterseniz, sistemi yeniden başlatmadan önce son şansınız.

İyi bir kurallar zinciri yaratmak bir sanattır desem sanırım yanılmış olmam.Eğer ilk defa bir güvenlik duvarı yaratmaya çalışıyorsanız ağ veya internet ile hiçbir işiniz olmadığı ve makineyi yeniden başlatıp yeni şeyler deneyebileceğiniz bir zamanı seçmeniz sizin için çok yararlı olacaktır.Kuralları yerleştirmeye başladıkça `ipfw` tarafından kullanılan mantığın sizin mantığınız gibi çalışmadığını göreceksiniz.

Ayrıca unutmamanız gereken diğer şey ise güvenlik duvarının bir kere kurup daha sonra unuttuğunuz bir şey olmadığıdır.İstediğiniz şeyleri istediğiniz gibi yapmadığını anladığınızda üzerinde çalışıp parametrelerini düzenlemeye hazır olmalısınız.Artık güvenlik duvarı kullanılan sisteminizi yeniden başlattıktan sonra aşağıda yazılı kuralları tamamlamak için zaman ayırmaniz gerekecek.

1 Sistematik olarak kurallarınızı kurallar zincirinize teker teker ve her birini deneyip sadece istediğiniz paketlere izin verdiğinden emin olarak ekleyin.

2. Günlük (log) tutulmasını isteyip istemediğinize karar verin ve tutulan günlükleri izleyin.Muhtemelen bunun sonunda tekrar kurallar zincirinize geri donup bazı ekleme ve çıkarmalar yaparak yanlışlıkla engellediğiniz paketlere tekrar izin vermeniz gerekecektir.

3. Güvenlik duvarınızın istediğiniz paketlere izin verip, istemediklerinizi düşürdüğünü anlayınca bir kez de kendiniz deneyip bundan emin olun.

Tamam simdi yeni kernel`ı yüklemek için sistemimi yeniden başlatabilirim.Bu arada sistem açılısındaki mesajları dikkatle izleyip aşağıdaki mesajların görüntülendiğinden emin olmam gerekiyor:

```
Flushed all rules.
00100 allow ip from any to any via lo0
00200 deny ip from any to any to 127.0.0.0/8
Firewall rules loaded, starting divert daemons:.
Additional routing options: tcp extensions=NO ignore ICMP redirect=YES
TCP keepalive=YES restrict TCP reset=YES drop SYN+FIN packets=YES.
```
<Bazı çıktılar silinmiştir.>
```
Additional TCP options: log_in_vain=YES.
```

Kurallar zinciri dosyamı yaratmadığım halde ilk üç satirin nasıl görüntülendiğini merak edebilirsiniz. `etc/rc.conf` dosyasına ekleme yaptığımda aşağıdaki satiri eklediğimi biliyorsunuz:

```
firewall_script="/etc/rc.firewall"
```

İşte bu dosya aşağıdaki satırları içermekte:

```
############
# Flush out the list before we begin.
#
${fwcmd} -f flush

############
# Only in rare cases do you want to change these rules
#
${fwcmd} add 100 pass all from any to any via lo0
${fwcmd} add 200 deny all from any to 127.0.0.0/8
```

İlk başta bulunan 100 ve 200 numaralı kuralların kullanılmasını tavsiye ettiği için kurallar zincirimi yaratırken 300`den başlayacağım.Fakat bunu yapmadan önce `ipfw` `un bütün paketleri engellediğinden emin olmak istiyorum.Bunun için `ipfw show` komutunu kullanacağım:

```
ipfw show
ipfw: socket: Operation not permitted
```

Görünüşe bakılırsa sadece süper kullanıcı güvenlik duvarı kurallarını görmeye yetkili.Bir de süper kullanıcı olarak deneyelim:

```
su
Password:
ipfw show
00100   0     0 allow ip from any to any via lo0
00200   0     0 deny ip from any to 127.0.0.0/8
65535 115 14092 deny ip from any to any
```

Emin olmak için başka bir yöntem deneyelim:

```
ping www.freebsd.org
ping: cannot resolve www.freebsd.org: Host name lookup failure
traceroute www.freebsd.org
traceroute: unknown host www.freebsd.org

lynx www.freebsd.org
Alert!: Unable to access document.
```

Hmm.İsim – ip çözünürlüğü görünüşe göre çalışmıyor.Bir de ip`yi ping`lemeyi deneyelim:

```
ping 24.141.116.1
PING 24.141.116.1 (24.141.116.1): 56 data bytes
ping: sendto: Permission denied
ping: sendto: Permission denied
^C
--- 24.141.116.1 ping statistics ---
2 packets transmitted, 0 packets received, 100% packet loss
```

En son ping komutunu süper kullanıcı olarak kullandığımdan dolayı artık `ipfw` `in bütün paketleri düşürdüğünden emin oldum.bütün ağ ve internet bağlantılarımı kapattığıma göre artık istediğim paketlere izin vermeleri için kullanacağım kurallar zincirimi yaratmaya başlayabilirim.

`Ipfw` tarafından okunan kuralları yaratmanın iki yolu bulunmakta:

- Eğer `ipfw add` komutunu kullanırsanız; kuralın uygulamaya geçirilmesi için sistemi yeniden başlatmanız gerekmeyecektir.Sistem yeniden başlatıldığında ise yazdığınız kurallar silinecektir.
- `ipfw` tarafından okunmasını söylediğiniz kurallar zinciri dosyanıza satırlar haline ekleme yapabilirsiniz.Yalnız bu kurallar sistem yeniden başlatılana kadar uygulamaya girmeyecektir.

Bu bilgisayarı kullanan tek kişi ben olduğumdan dolayı kuralları doğrudan dosyaya ekleyecek ve sistemi yeniden başlatacağım.Daha önceden hatırlarsanız aşağıdaki parametreyi `/etc/rc.conf` dosyasına eklemiştik:

```
firewall_type="/etc/ipfw.rules"
```

Öyle ise simdi `etc/ipfw.rules` dosyasını yaratmamız gerekiyor.

Kuralların yaratılması daha önce belirttiğim gibi bayağı bir uğraş gerektiriyor.Bunun için aşagidaki kaynaklardan yararlanarak kendi kurallarınızı yaratabilirsiniz:

man ipfw

http://www.robertgraham.com/pubs/firewall-seen.html

http://www.defcon1.org//html/Networking_Articles/Firewall-Ipfw/firewall-ipfw

http://www.interhack.net/pubs/fwfaq/

http://www.freebsddiary.org/firewall.html

http://www.bsdtoday.com/2000/December/Features359.html

http://www.freebsd-howto.com/HOWTO/Ipfw-HOWTO

http://www.daemonnews.org/200103/firewall.html

Gördüğünüz gibi bu konuda çok sayıda dokuman bulunmakta.Okudukça dikkatinizi çekecektir; kullanılacak kurallar zinciri her kişi ve ağ için değişim göstermekte.Sonuç olarak başka bir metinden alacağınız kuralları tamamen kullanamayacak buna eklemeler ve düzeltmeler ile size uygun hale getireceksiniz.

Özgür Özdemircili
http://www.enderunix.org
http://news.enderunix.org
http://haber.enderunix.org

Sorularınız için : dionypheles@gmx.net

Kaynaklar

Dru Lavigne`in “BSD Firewalls: IPFW” adlı yazısından çevrilmiştir.

Yazar`in sayfasına http://www.oreillynet.com/pub/au/73 , orijinal metne ise http://www.onlamp.com/pub/a/bsd/2001/04/25/FreeBSD_Basics.html adresinden ulaşabilirsiniz.