FreeBSD istemcileri arasında Ipsec tünelleme

Bir tür ip güvenlik çeşidi olan Ipsec, ip paket seviyesine kadar olan Internet trafiğini şifreleme yöntemidir.Bu da SMTP gibi güvenliği düşük olan protokolleri güvenli haline getirebilme sansını bize veriyor.Ipv6 açısından ne kadar Ipv4 tarafında gibi gözükse de Ipsec Ipv6 ile çok rahat bir şekilde çalışabiliyor. Makalede İngilizce terimlerine karşılık olarak kullandığım kelimeleri açıklayıcı olması bakımından belirterek başlayalım:

```
access point   =   erişim noktası

transport mode =   taşıma modu

tunel mode     =   tünel modu

tunellıng      =   tünelleme

key lifetimes  =   anahtar süreleri

pre-shared key =   ön paylaşım anahtarı

host           =   istemci
```

Ipsec`in iki önemli bölümü Authentication Header (AH) ve Encapsulating Security Protocol (ESP)`dir.AH gönderilen bilginin gerçekten bilgi olduğunu ve gönderen kişinin gerçekten gönderen olduğunu ispat etmek için kullanılıyor.ESP ise gönderilen paketleri şifreleyerek paketin yol aldığı sure içinde kotu gözlerden uzak durmasını garantiliyor.

Hem ESP hem de AH tünel veya taşıma (Transport) seklinde kullanıla biliniyor.taşıma modu iki son nokta arasında güvenlik sağlarken, tünel modu ise VPN turu güvenlik sağlıyor.Taşıma modunda paketlerde yazılı bulunan başlık ve kaynak adresi imzalanıp şifreleniyor.Daha sonra varis noktasına gönderilen paketler alındıktan sonra onaylanıyor, şifreleri açılıp normal paketler haline getiriliyor.taşıma modu en çok olarak iki istemci arasında kullanılıyor.Yalnız kaynak IP`si önemli olduğundan NAT kullanılan ağlarda maalesef kullanılamıyor.

Tünel modunda AH veya ESP (Bazen her ikiside) tüm paketi şifreleyerek veya imzalayarak yeni bir paket haline getiriyorlar ve bu paket karsı tarafa yollanıp orada tekrar onaylanıyor ve şifresi acılıyor.şifresi açılıp onaylanan paket bundan sonra normal bir paket olarak işlem görüp en son hedefine gönderiliyor.Anlaşılabileceği gibi tünel modu iki router veya istemci ile router arasında kullanmak için mükemmel bir yöntem sunuyor.Orijinal paket kendi bütünlüğü içinde şifrelenip imzalandığı için tünel modunda ESP NAT kullanılan ağlar üzerinde kullanıla biliniyor.

Tünel modu ESP`si ağ üzerindeki güvenlik eksiği olan bölümleri güvenli hale getirmek için harika bir yöntem.Örneğin su anda bu yazıyı evimde bulunan kablosuz ağ aracılığı ile salonumda yazıyorum.Laptopum ve internet arasındaki trafik İlk olarak FreeBSD

kullandığım erişim noktam tarafından şifrelenip tünelleniyor.Böylece laptop`um ve erişim noktam (Access point) arasında bilgileri yakalayan herhangi bir kişi bu bilgileri anlayamıyor.

Ipsec kurulumu

IKE

İlk olarak her iki istemciyi IKE yani Internet Key Exchange kullanmaları için ayarlamamız gerekiyor.IKE Ipsec`in fazla sayıda olan şifre anahtarlarını istemciler arasında güvenli şekilde değiştirmesine olanak tanıyor.FreeBSD`de IKE FreeBSD 4.0 ve ustu versiyonlarında standart olarak gelen racoon deamon`u tarafından idare ediliyor. Uyumluluk konusunda emin olmak için her iki istemci üzerinde ayni versiyon işletim sistemi kullanıla bilinir.

Racoon`u `/usr/ports/security/racoon` dosyası içinden yükleyebiliyoruz.Bunun icin tek yapmamiz gereken

```
kullanici$:su
password:
root$: cd /usr/ports/security/raccoon
root$:make & install & clean
```

Racoon programının konfigürasyon dosyası standart yükleme ile `/usr/local/etc/racoon/racoon.conf` olarak belirleniyor ve yanında bir de pke.txt dosyası ile geliyor.`.conf` dosyası üzerinde çok fazla oynama yapmamıza gerek olmamasına rağmen `life time xxx <min|sec|hour|>` bölümünde bulunan anahtar süreleri (Key lifetimes) ile oynamak isteyebilirsiniz.Kısaca söylemek gerekirse daha kısa anahtar süreleri `ları size daha güvenli bir ortam sağlayacaktır.Bunun yanında daha uzun anahtar süreleri çok fazla işlemci gücü yiyebilir veya kurulu olan bağlantıları engelleyebilir.

Burada bizim için en önemli dosya pke.txt dosyası.Burada önemli olan bu dosyanın `chmod 600` ve süper kullanıcıya ait olması gerekmektedir.Eğer bunlar sağlanamazsa racoon programı bu dosyayı okumayı reddedecektir.Bu dosya racoon`un ilk bağlantıyı kurması için gerekli olan daha önceden paylaşılmış anahtarları içermekte.

Örnek zamanı.Mason (10.0.0.2), tünel sunucusu ve gluon (10.0.0.77) arasında tünel kurmayı deneyelim.Hepsinde kullanılacak olan ön-paylaşım anahtarı Macplusbsdcool.Mason`da bulunan pke.txt dosyası :

```
10.0.0.77     macplusbsdcool
```

şeklinde görüzükürken, Gluon`da:

```
10.0.0.2      macplusbsdcool
```

Şeklinde olmalı.Şimdi her iki istemcide deamon`ları başlatalım:

```
/usr/local/sbin/racoon -f \
/usr/local/etc/racoon/racoon.conf -l /var/log/racoon.log
```

Kurallar

Şu anda her iki istemci`de anahtarları değiştirmeye hazır olmalarına rağmen bunu yapmaları gerektiğini bilmiyorlar.Bunun için son olarak her iki istemci üzerinde , istemcilerin Ipsec kurallarını ne zaman ve nasıl kullanmaları gerektiğini belirtecek Ipsec kurallarını belirlememiz gerekecek.Bunun için `setkey` komutunu kullanacağız.Bu açacağımız tünelin devamlı acık kalmasını isteyeceğimiz için bütün kuralları bir dosyaya koyup `setkey`in bu dosyadan okumasını isteyeceğiz:

```
pico /etc/rc.conf
setkey -f /etc/ıpsec.rules
```

Tünelleme için istemci paketlerinin mason üzerinden güvenli hale getirilip oradan hedefe ulaştırılmasını istiyoruz.Bunun için gluon üzerindeki konfigürasyon dosyası:

```
spdadd 10.0.0.77 0.0.0.0/0 any -P out ıpsec
esp/tunnel/10.0.0.77-10.0.0.2/require;
spdadd 0.0.0.0/0 10.0.0.77 any -P in ıpsec
esp/tunnel/10.0.0.2-10.0.0.77/require;
```

Şeklinde olurken Masonda ise:

```
spdadd 0.0.0.0/0 10.0.0.77 any -P out ıpsec
esp/tunnel/10.0.0.2-10.0.0.77/require;
spdadd 10.0.0.77 0.0.0.0/0 any -P in ıpsec
esp/tunnel/10.0.0.77-10.0.0.2/require;
```

Olmalı.Satırlar sanırım yeterince açık; `spdadd` yeni bir Ipsec kuralı eklemek istediğinizi, `any` kuralın bütün protokolleri için gerekli olduğunu, out ve in paketin hareket ettiği yönü belirtiyor.Son olarak `esp/.../require` kernelin Ipsec`i tünel modunda kullanması gerektiğini ve bunu yazılı olan birinci ip den ikinci ip`ye yapmasını belirtiyor.

Bu senaryonun Ipv6`da çalışması için gerekli olan tek şey Ipv4 yerine Ipv6 adreslerinin kullanılması olacaktır.Her iki makine da `setkey` komutunu kullanarak kuralları etkinleştirin.

Sıra Testte

Test için yapmamız gereken çok basit.Şu bizim eski ve her derde deva dostumuzu kullanacağız.

```
ping 10.0.0.1
```

Bağlantınızın Ipsec`i kullanıp kullanmadığını anlamak için ise `tcpdump`ı istemci üzerinde kullanıp yakaladığımız paketlere bir göz atalım:

```
tcpdump –a –vv –i tun0

13:58:39.045831 gluon > mason: ESP(spi=0x099e806d,seq=0x723)
13:58:39.046566 mason > gluon: ESP(spi=0x0d8d15b6,seq=0x6b2)
13:58:39.187444 gluon > mason: ESP(spi=0x099e806d,seq=0x724) (frag
64877:1480@0+)
```

Bu şekilde bir çıktı alıyorsanız her şey yolunda demektir.

Eğer ESP dışı herhangi bir trafik görüyorsanız ise bir şeyler yanlış gidiyor demektir.Bu durumda İstemci üzerinde varis adresinin `0.0.0.0/0` olduğundan emin olun.Eğer hiçbir ESP trafiği göremiyorsanız ise iki makine üzerinde belirlediğimiz kuralları kontrol edip doğruluğundan ve pke.txt dosyalarında yazılan anahtarların ayni olduğundan emin olun.Eğer yine bir problem bulamıyorsanız son çare olarak `/var/log/racoon.log` dosyasına bir göz atin.

Bir öneri daha.Eğer makinelerden birini boot ettiyseniz ve bağlantı sorunu bundan sonra başlamış ise iki istemci üzerindeki anahtarları `setkey –F` komutu ile temizleyin.Tekrar başlatma işlemi muhtemelen anahtarların değişikliğe uğramasına yol açmıştır.

Sonuç

Şu anda çalışan bir Ipsec tüneliniz olması ve bilgilerinize kimsenin müdahale edemeyeceği bir ortamın keyfini gönül rahatlığıyla yaşayabilirsiniz

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

Sorularınız için : dionypheles@gmx.net

Kaynaklar

Mike DeGraw-Bertsch `in “Ipsec tunelling between FreeBSD hosts” adlı yazısından çevrilmiştir.

Yazar`in sayfasına http://www.onlamp.com/pub/au/188, orijinal metne ise http://www.onlamp.com/pub/a/bsd/2001/12/10/ipsec.html adresinden ulaşabilirsiniz.