

CONTEXT-BASED ACCESS CONTROL(CBAC)

Taner KOÇ – İTÜ/BİDB 2009

CBAC'in Özellikleri

Cisco IOS Firewall'un mümkün kıldığı bir özellik olarak CBAC, uygulama katmanı bilgisini temel alarak TCP ve UDP paketlerini filtreleme işlemi gerçekleştirir. Uygulamaya özel protokollerde, multimedya uygulamalarında ve birden fazla kanala ihtiyaç duyan uygulamalarda CBAC, application katmanı filtrelemesi sağlar.

CBAC'in trafik filtreleme,trafik denetleme,ihlal tespiti,denetim kaydı ve uyarı verme gibi dört ana işlevi vardır.

Trafik Filtreleme:

CBAC, güvenlik duvarı vasıtasıyla ağ içerisinde başlatılmış bağlantılara izin verecek şekilde konfigüre edilebilir. Bunu ACL içerisinde geçici açılımlar oluşturarak yerine getirir. CBAC filtreleme işlemini yalnızca network ve transport katmanı bilgisine göre değil aynı zamanda application katmanı bilgisine göre filtreleme yapar.

Trafik Denetleme:

CBAC'te uygulama katmanında denetlenen paketlerin oturum bilgileri incelenir. Bu bilgiler sayesinde SYN-flood atağı gibi ataklar önlenir. TCP paketlerin sequence numaralarını denetleyerek, bu numaraların kabul edilebilir bir aralıkta olup olmadıklarını belirler ve şüpheli paketleri çöpe atar. Böylece SYN-flood atağında olduğu gibi atak yapan kişinin gönderdiği bağlantı istekleri reddedilerek sistemin kaynaklarını kullanılamaz duruma getiren DoS atağı engellenmiş olur.

İhlal Tespiti:

Belli türde ataklar belli türde karakteristiklere sahiptirler. Syslog mesajlarının denetlenmesi sırasında eğer bu ataklardan biri CBAC tarafından tespit edilirse bağlantılar sıfırlanır ve syslog mesajı düşülür.

Denetim Kaydı ve Uyarı Verme:

CBAC gerçek zamanlı denetim kayıtları ve uyarılar oluşturarak güvenlik duvarındaki durum hakkında bilgiler edinilmesini sağlar. Böylece kaynak ve hedef makinalarla, kullanılan portlarla, taşınan verinin miktarıyla ve daha birçok özelle ilgili bilgi edinilebilir.

Not:CBAC yalnızca yönetici tarafından belirtilmiş protokollerin denetiminde görev alır. Bunun dışındaki protokollerin durumu mevcut ACL'ler tarafından kontrol edilir. Sadece güvenlik duvarından geçen ataklar tespit edilebilir. Cisco IOS firewall'dan geçmeyen ataklar tespit edilemez.

CBAC Nasıl Çalışır?

CBAC devrede değilken filtreleme işlemi ACL'ler tarafından network ve transport katmanlarında yapılmaktadır. Fakat CBAC bir oturumun bütün dinamik uygulamalarını tanıyabilir. Aktif bağlantılarla ilgili durum tabloları oluşturur. Bu durum tabloları vasıtasıyla dinamik ACL'ler oluşturulur ve paket geçişleri bu sayede denetlenmiş olur.

Güvenlik duvarının interfacelerindeki ACL'lerde içeriden dışarıya başlatılan trafik için geçici ACL'ler oluşturulur ve bu geçici ACL'ler yoluyla dönen trafik bu başlatılan trafiğin bir parçasıysa ve onunla aynı özellikleri taşıyorsa içeri girmesine izin verilir. Aksi takdirde dışarıdan gelen trafik engellenmiş olur.

CBAC çalışma mekanizmasını birkaç adımda şu şekilde anlatabiliriz:

1-Korunan networkümüzden trafik oluşturulduğunda routerımızdan geçerken bir ACL mevcutsa ACL işletilir. ACL pakete izin vermiyorsa paket çöpe atılır, eğer izin veriyorsa CBAC denetim kuralları uygulanır.

2- CBAC denetim kuralları uyarınca Cisco IOS Software bağlantıyı inceleyebilir. Eğer başlatılan trafiği denetlenmiyorsa paketin geçmesine izin verilir, bilgi toplanmaz. Eğer başlatılan trafik denetleniyorsa bir sonraki adıma geçilir.

3- Bağlantı mevcut durum tablosuyla karşılaştırılır. Eğer bağlantı önceden mevcut değilse girdi eklenir. Eğer bağlantı önceden mevcutsa bağlantı için zaman sıfırlanır.

4- Yeni bir girdi ekleniyorsa çıkış interface'ine giriş yönünde dinamik ACL eklenir. Bu ACL başlatılan trafiğin dönen paketlerine izin verir. Bu açılım mevcut oturum açıkken sadece aktiftir. Dinamik ACL'le NVRAM'e kaydedilmez.

5- Oturum sonlandığında dinamik ACL ve durum tablosundaki dinamik bilgi silinir.

CBAC Konfigürasyonu:

Konfigürasyon yapılırken uygulanacak adımla şöyledir:

1-İç veya dış bir interface seçilir:Oturumun başlatılacağı interface iç interface olarak seçilir. Dönen paketlerin denetleneceği interface dış interface'tir.İstenirse CBAC birden fazla interface'te iki yönde konfigüre edilebilir.

2-Interface'e IP ACL'ler uygulanır:ACL ile güvenlik duvarımızın denetleyeceği trafiğe dışarı yönde izin verilir. Dış networkten korunacak networkümüze girecek trafiği filtrelemek için extended ACL uygulanır ve bu ACL üzerinde CBAC dinamik açılımlar yapar.

3-Denetleme kuralları tanımlanır:Yönetici tarafından interface'te hangi uygulama katmanı protokolünün denetleneceği belirlenmelidir.Denetleme kuralı global konfigürasyon modunda konfigüre edilir. Konfigürasyonu şu şekilde yapılır:

```
Router(config)# ip inspect name inspection_name protocol [alert {on | off}] [audit-trail {on | off}] [timeout seconds]
```

4-Denetleme kuralı interface'e uygulanır: Denetleme kuralı konfigüre etmenin son adımı denetleme kuralını interface'e atamaktır. Konfigürasyonu şu şekilde yapılır:

```
Router(config-if)# ip inspect inspection_name {in | out}
```

Denetleme kuralı atanırken dikkat edilemesi gereken nokta içeriden başlatılacak trafiğin geçeceği interface'te ACL iç yönde uygulanır ve aynı şekilde denetim kuralı da iç yönde uygulanır. Diğer interface'lerde denetlenecek trafiği engelleyecek ACL iç yönde uygulanır.

Denetleme kurallarını sıfırlamak için konfigürasyon şu şekilde yapılır:

```
Router(config)# no ip inspect
```

CBAC Troubleshooting:

Uyarılar(Alerts):

Uyarılar CBAC operasyonunu ilgilendiren yetersiz router kaynakları, DoS atakları ve diğer tehditleri görüntüler.

Uyarıların kapatılması önerilmez, fakat uyarıları kapatmak için konfigürasyon şu şekilde yapılır:

```
Router(config)# ip inspect alert-off
```

Denetleme Kayıtları:

Kayıtlar geçerli ve geçersiz girişimler dahil denetlenecek bağlantıların izini sürer. Örneğin CBAC durum tablosuna bir ekleme veya çıkarma yaptığında mesajlar görüntüler. Kayıtlar bağlantılar hakkında istatistiksel bilgiler verir. İstenirse bu kayıtlar konsol ekranında görüntülenebileceği gibi dış bir syslog serverında da tutulabilir.

CBAC aynı zamanda geçici ACL'leri , durum tablosunu ve CBAC operasyonunu görüntüleyebileceğimiz show komutlarına sahiptir.

Bunları görüntülemek için kullanılacak komut şu şekildedir:

```
Router# show ip inspect [parameter]
```

name inspection_name -> Sadece istenen isimdeki denetleme kuralını görüntüler.

config -> Bütün CBAC operasyonunu görüntüler.

interfaces -> Router interface'indeki aktive edilmiş denetleme kurallarını görüntüler.

sessions -> CBAC durum tablosundaki bağlantıların özetini görüntüler

sessions [detail] -> Durum tablosundaki bütün bağlantıları görüntüler

all -> Listedeki bütün özellikleri görüntüler

Daha detaylı troubleshooting yapmak için yönetici debug komutlarını da kullanabilir.

Bunun için kullanılacak komut şu şekildedir:

```
Router# debug ip inspect protocol parameter
```

tcp -> tcp denetleme olaylarını görüntüler

udp -> udp denetleme olaylarını görüntüler

icmp -> icmp denetleme olaylarını görüntüler

application_name -> uygulamaya yönelik olayları görüntüler

events -> paketlerin işlenmesi ile ilgili CBAC olaylarını görüntüler

object-creation -> durum tablosuna yapılan girişlerle ilgili olayları görüntüler

object-deletion -> durum tablosunda yapılan silinmelerle ilgili olayları görüntüler

function-trace -> CBAC'in çağırdığı yazılım fonksiyonları ile ilgili bilgiyi görüntüler

timers -> Zamanlama ile ilgili bilgileri görüntüler

detailed -> Bütün CBAC süreçleri hakkındaki bilgileri görüntüler.