Dosya Bütünlüğünün İzlenmesi

Dosya Bütünlüğü İzleme (FIM), sistem ve uygulama dosyalarının bütünlüğünü izlemek için kullanılan bir güvenlik işlemidir. FIM, hassas varlıkları izleyen her kuruluş için önemli bir güvenlik savunma katmanıdır. Hassas veriler, uygulama ve cihaz dosyalarını izleyerek, düzenli olarak tarayarak ve bütünlüklerini doğrulayarak koruma sağlar. Kuruluşların sistemlerindeki kritik dosyalardaki değişiklikleri tespit etmelerine yardımcı olarak verilerin çalınması veya tehlikeye atılması riskini azaltır. Bu işlem, kaybedilen üretkenlik, kaybedilen gelir, itibar hasarı ve yasal ve düzenleyici uyumluluk cezalarında zamandan ve paradan tasarruf sağlayabilir.

Wazuh, dosya bütünlüğü izleme için yerleşik bir yeteneğe sahiptir. Wazuh FIM modülü dosyaları ve dizinleri izler ve bir kullanıcı veya işlem izlenen dosyaları oluşturduğunda, değiştirdiğinde ve sildiğinde bir uyarı tetikler. İzlenen dosyaların kriptografik toplam kontrolünü ve diğer özniteliklerini depolayan bir temel tarama çalıştırır. Bir kullanıcı veya işlem bir dosyayı değiştirdiğinde, modül toplam kontrolünü ve özniteliklerini temelle karşılaştırır. Bir uyumsuzluk bulursa bir uyarı tetikler. FIM modülü, aracılar ve yönetici için FIM yapılandırmasına bağlı olarak gerçek zamanlı ve zamanlanmış taramalar gerçekleştirir.

Wazuh FIM yeteneğinin bazı faydaları şunlardır: değişiklik yönetimi, tehdit tespiti ve yanıtlama ve düzenlemelere uyum.

Değişim Yönetimi

Wazuh FIM yeteneği, değişiklik yönetimi süreçlerinin doğru çalıştığını doğrulamak için olmazsa olmaz bir araçtır. Bu Wazuh yeteneği, dosyaların değişip değişmediğini, nasıl ve ne zaman değiştiğini ve kimin veya neyin onları değiştirdiğini görmek için incelemenize olanak tanır. Wazuh FIM modülü, temel bilgileri dosyanın en son sürümündeki bilgilerle karşılaştırır. Bu karşılaştırma, kritik dosyalardaki değişikliklere ve güncellemelere ilişkin görünürlük sağlar. Örneğin, bunu uygulamalardaki yanlış güncellemeleri veya yapılandırma dosyalarında yapılan yetkisiz değişiklikleri tespit etmek için kullanabilirsiniz.

Tehdit Tespiti ve Yanıtlama

FIM'i tehdit algılama ve yanıtlama için diğer Wazuh yetenekleriyle birleştirebilirsiniz. FIM yeteneği dosya bütünlüğünü izler, izin değişikliklerini algılar ve kullanıcı ve dosya etkinliklerini izler. Algılanan tehditlere hızlı yanıtlar için ayrıntılı uyarılar sağlar.

Mevzuata Uygunluk

FIM yeteneği, kuruluşların veri güvenliği, gizlilik ve veri saklama için düzenleyici gereklilikleri karşılamalarına yardımcı olur. Kritik dosyaları değişiklikler açısından izlemek, PCI DSS, HIPAA ve GDPR gibi düzenlemeler için önemli bir gerekliliktir.

Nasıl Çalışır?

FIM modülü belirli yollarda periyodik taramalar çalıştırır ve gerçek zamanlı olarak belirli dizinlerdeki değişiklikleri izler. Wazuh aracılarının ve yöneticisinin yapılandırmasında hangi yolların izleneceğini ayarlayabilirsiniz.

FIM, dosyaların toplam kontrol değerlerini ve diğer özniteliklerini yerel bir FIM veritabanında depolar. Bir tarama sırasında, Wazuh aracısı, FIM modülünün izlenen yollarda bulduğu tüm değişiklikleri Wazuh sunucusuna bildirir. FIM modülü, bir dosyanın toplam kontrol değerlerini, depolanan toplam kontrol değerleri ve öznitelik değerleriyle karşılaştırarak dosya değişikliklerini arar. Tutarsızlıklar bulursa bir uyarı oluşturur.

Wazuh FIM modülü, dosya oluşturma, değiştirme ve silme verileri gibi FIM olay verilerini toplamak için iki veritabanı kullanır. Biri, verileri şurada depolayan izlenen uç noktadaki yerel bir SQLite tabanlı veritabanıdır:

- `C:\Program Files (x86)\ossec-agent\queue\fim\db` Windows'ta.
- `/var/ossec/queue/fim/db` Linux'ta.
- `/Library/Ossec/queue/fim/db` macOS'ta.

Diğeri Wazuh sunucusundaki bir aracı veritabanıdır. Wazuh-db . daemon, Wazuh sunucusundaki her bir aracı için bir veritabanı oluşturur ve yönetir. Veritabanını tanımlamak için aracının kimliğini kullanır. Bu hizmet veritabanlarını . adresinde depolar `/var/ossec/queue/db`.



FIM modülü, Wazuh aracısını ve Wazuh sunucu veritabanlarını birbirleriyle senkronize tutar. Wazuh sunucusundaki dosya envanterini her zaman Wazuh aracısının erişebildiği verilerle günceller. Güncel bir Wazuh sunucu veritabanı, FIM ile ilgili API sorgularının servis edilmesini sağlar. Senkronizasyon mekanizması, Wazuh sunucusunu yalnızca Wazuh aracılarından gelen, kontrol toplamları ve değişen dosya öznitelikleri gibi bilgilerle günceller.

Wazuh aracısı ve yöneticisi varsayılan olarak FIM modülünü etkinleştirmiş ve önceden yapılandırmıştır . Ancak, izlenen yollar gibi FIM ayarlarını ortamınıza göre uyarladığınızdan emin olmak için uç noktalarınızın yapılandırmasını gözden geçirmenizi öneririz.

FIM Modülü Nasıl Yapılandırılır

FIM modülü Windows, Linux ve macOS işletim sistemlerinde taramalar çalıştırır. Hem genel ayarlar hem de uç noktanın işletim sistemine özgü ayarlar vardır. Bu ayarları ve desteklenen işletim sistemlerini bu kılavuzun Temel ayarlar bölümünde ele alıyoruz.

FIM modülünün dosyaların oluşturulmasını, değiştirilmesini ve silinmesini izlemesi gereken dizinleri belirtmeli veya izlemeniz gereken belirli dosyaları yapılandırmalısınız. Wazuh sunucusunda ve Wazuh aracı yapılandırma dosyalarında izlenecek dosyayı veya dizini belirtebilirsiniz. Ayrıca bu özelliği merkezi yapılandırma dosyasını kullanarak uzaktan da yapılandırabilirsiniz .

İzlenecek dosyaları ve dizinleri dizin seçenekleriyle ayarlamanız gerekir . Virgülle ayrılmış girdiler kullanarak veya birden fazla satıra girdiler ekleyerek birden fazla dosya ve dizin ekleyebilirsiniz. FIM dizinlerini, bir kabukta veya Komut İstemi (cmd) terminalinde kullandığınız şekilde * ve ? joker karakterlerini kullanarak yapılandırabilirsiniz. Örneğin, `C:\Users\*\Downloads`.

FIM modülü bir tarama çalıştırdığında, değiştirilmiş dosyalar bulursa ve değiştirilen dosya özniteliklerine bağlı olarak uyarıları tetikler. Bu uyarıları Wazuh panosunda görüntüleyebilirsiniz.

Aşağıda, FIM modülünün bir dosyayı ve dizini izlemek için nasıl yapılandırılacağını görebilirsiniz. `<FILEPATH_OF_MONITORED_FILE>`ve'yi `<FILEPATH_OF_MONITORED_DIRECTORY>`kendi dosya yollarınızla değiştirin.

1. Aşağıdaki ayarları Wazuh aracı yapılandırma dosyasına ekleyin ve dizin değerlerini kendi dosya yollarınızla değiştirin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

   ```
   <syscheck>
     <directories><FILEPATH_OF_MONITORED_FILE></directories>
     <directories><FILEPATH_OF_MONITORED_DIRECTORY></directories>
   </syscheck>
   ```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

> Not: Hem merkezi yapılandırmada hem de Wazuh aracısının yapılandırma dosyasında bir dizin belirtirseniz , merkezi yapılandırma öncelik kazanır ve yerel yapılandırmayı geçersiz kılar.

FIM modül analizinin yorumlanması

FIM analiz sonuçları, izlenen dosyalarda bir ekleme, değişiklik veya silme olduğunda Wazuh panosunda görünür. FIM sonuçlarını panonun üç farklı bölümünde görüntüleyebilirsiniz. FIM modülünden sonuçları görüntülemek için Wazuh panosunda **Dosya Bütünlüğü İzleme'ye** gidin . Sonuçlar aşağıdaki bölümlerdedir:

- Envanter
- Gösterge Paneli
- Olaylar

Envanter

Bu bölüm, FIM modülünün dizinlediği tüm dosyaların envanterini görüntüler. FIM veritabanı, dosya adı, son değişiklik tarihi, kullanıcı, kullanıcı kimliği, grup ve dosya boyutu dahil olmak üzere envanter bilgilerini içerir. Aşağıdaki görüntü, bir Ubuntu 22.04 uç noktasının dosya envanterini gösterir.

Envanter

Image not found or type unknown

FIM modülünün dosyayı en son ne zaman analiz ettiği ve dosya öznitelikleri gibi giriş ayrıntılarını görüntülemek için bir dosya girişine tıklayabilirsiniz. Ayrıca dosyayla ilgili FIM uyarılarını da görüntüleyebilirsiniz. Aşağıdaki görüntü dosya için bu bilgileri gösterir `/etc/resolv.conf`.

Giriş detayları

Image not found or type unknown

Dashboard

Gösterge paneli bölümü, Wazuh FIM modülünün aşağıdakilere ilişkin analiz sonuçlarına genel bir bakış sunar:

- Bir altyapı içindeki tüm etkenler.
- Bir altyapı içerisinde seçilmiş bir ajan.

Aşağıdaki görüntüde, izlenen tüm uç noktalar için FIM tarama sonuçlarının genel görünümüne ilişkin bir örnek görebilirsiniz.

Gösterge Paneli
Image not found or type unknown

Aşağıdaki görüntüde Ubuntu uç noktası için FIM tarama sonuçlarının genel görünümüne dair bir örnek görebilirsiniz.

FIM tarama sonuçlarına genel bakış
Image not found or type unknown

Events

Bu bölüm Wazuh FIM modülünün tetiklediği uyarıları gösterir. Burada, aracı adı, izlenen dosyanın dosya yolu, FIM olayının türü, uyarının açıklaması ve uyarının kural düzeyi gibi ayrıntıları görebilirsiniz.

Olaylar
Image not found or type unknown

Ayrıca, uyarıyı tetikleyen olay hakkında ek bilgileri görüntülemek için her uyarı girişini genişletebilirsiniz.

Genişletilmiş uyarı girişi
Image not found or type unknown

Temel Ayarlar

FIM yeteneğini Wazuh sunucusunda ve Wazuh aracısında yapılandırabilirsiniz. Hem Wazuh sunucusunda hem de Wazuh aracısında varsayılan bir FIM yapılandırması mevcuttur. Bu ayarları ihtiyaçlarınıza göre değiştirebilirsiniz.

FIM modülünü Wazuh sunucusunda ve Wazuh aracı yapılandırma dosyasında yapılandırabilirsiniz .

Ayrıca bu yeteneği merkezi yapılandırma dosyasını kullanarak uzaktan da yapılandırabilirsiniz . Tüm FIM yapılandırma seçeneklerinin listesi syscheck bölümünde mevcuttur.

Bu kılavuzda, Wazuh FIM modülünün desteklediği farklı yapılandırma seçeneklerini gösteriyoruz.

Gerçek Zamanlı İzleme

Bu `realtime`özellik yalnızca Windows ve Linux uç noktalarındaki dizinlerin gerçek zamanlı/sürekli izlenmesini sağlar.

Dosyaları gerçek zamanlı olarak izlemek için FIM modülünü dizinler `realtime` seçeneğinin niteliğiyle yapılandırın. Nitelik için izin verilen değerler ve'dir ve yalnızca dizinlerle çalışır, tek tek dosyalarla değil. Gerçek zamanlı değişiklik algılama, zamanlanmış FIM modülü taramaları sırasında duraklatılır ve bu taramalar tamamlanır tamamlanmaz yeniden etkinleştirilir.`realtime``yes``no`

Aşağıda, FIM modülünün bir dizini gerçek zamanlı olarak nasıl izleyeceğini görebilirsiniz. `<FILEPATH_OF_MONITORED_DIRECTORY>`Kendi dosya yolunuzla değiştirin.

> Not: Gerçek zamanlı izleme için bir dizin belirtirken, Wazuh aracısını yeniden başlatmadan önce mevcut olmalıdır. Aksi takdirde, modül Wazuh aracısının sonraki yeniden başlatılmasında bulana kadar dizini yoksayar.

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`

   ```
   <syscheck>
     <directories realtime="yes"><FILEPATH_OF_MONITORED_DIRECTORY></directories>
   </syscheck>
   ```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`

Kayıt Dosyası Öznitelikleri

FIM modülünü belirli dosyaları ve dizinleri izleyecek şekilde yapılandırdığınızda, dosyaların meta verilerini kaydeder ve izler. FIM modülünün toplaması ve yoksayması gereken belirli dosya meta verilerini ayarlamak için dizinler seçeneğini kullanabilirsiniz. Dizinler seçeneği çeşitli öznitelikleri destekler.

Aşağıdaki tabloda FIM modülünün kaydettiği desteklenen öznitelikler açıklanmaktadır.

Bağlanmak	Varsayılan değer	İzin verilen değerler	Tanım
check_all	Evet	evet hayır	Aşağıdaki tüm özniteliklerin değerlerini kaydeder.
check_sum	Evet	evet hayır	Dosyaların MD5, SHA-1 ve SHA-256 karmalarını kaydeder. Aynı anda check_md5sum="yes", check_sha1sum="yes", ve kullanmakla aynıdır. check_sha256sum="yes"
check_sha1sum	Evet	evet hayır	Dosyaların SHA-1 karma değerini kaydeder.
check_md5sum	Evet	evet hayır	Dosyaların MD5 hash'ini kaydeder.
check_sha256sum	Evet	evet hayır	Dosyaların SHA-256 karma değerini kaydeder.
check_size	Evet	evet hayır	Dosyaların boyutunu kaydeder.
check_owner	Evet	evet hayır	Linux'ta dosyaların sahiplerini kaydeder.
check_group	Evet	evet hayır	Dosyaların/dizinlerin grup sahibini kaydeder. Windows'ta gidher zaman 0'dır ve grup adı boştur.
check_perm	Evet	evet hayır	Dosyaların/dizinlerin izinlerini kaydeder. Windows'ta, her kullanıcı veya grup için reddedilen ve izin verilen izinlerin bir listesi kaydedilir. NTFS bölümleriyle Linux ve Windows'ta çalışır.
check_attrs	Evet	evet hayır	Windows'daki dosyaların özniteliklerini kaydeder.
check_mtime	Evet	evet hayır	Bir dosyanın değiştirilme zamanını kaydeder.
check_inode	Evet	evet hayır	Linux'ta dosya inode'unu kaydeder.

Aynı özniteliği değiştiren seçenekler arasında bir çakışma olduğunda, yapılandırılan sonuncu öncelik kazanır. Örneğin, aşağıdaki yapılandırma seçeneği check_mtimeşu şekilde ayarlar yes:

```
<directories check_all="no" check_mtime="yes">/etc</directories>
```

Aşağıdaki yapılandırma, değişiklik zamanı kontrolü dahil tüm özniteliklerin kaydedilmesini devre dışı bırakır.

```
<directories check_mtime="yes" check_all="no">/etc</directories>
```

Aşağıda izlenen bir dosyanın SHA-1 karmasının kaydının nasıl devre dışı bırakılacağına dair bir yapılandırma örneğini görebilirsiniz. <FILEPATH_OF_MONITORED_FILE>Kendi dosya yolunuzla değiştirin.

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:/var/ossec/etc/ossec.conf
   - Pencereler:C:\Program Files (x86)\ossec-agent\ossec.conf
   - macOS:/Library/Ossec/etc/ossec.conf

```
<syscheck>
  <directories check_sha1sum="no"><FILEPATH_OF_MONITORED_FILE></directories>
</syscheck>
```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control Restart`

> Not: İlk FIM taramasından sonra oluşturulan belirtilen dosyalar veya dizinler, bir sonraki planlanmış tarama sırasında izlenmek üzere eklenecektir.

Planlanmış Taramalar

FIM modülü taramalarının zamanlamasını değiştirmek için `<frequency>`Wazuh FIM modülünün seçeneğini yapılandırabilirsiniz. Bu seçenek, FIM taramaları arasındaki süreyi tanımlar. Alternatif olarak, scan_time ve scan_day seçeneklerini kullanarak taramaları haftanın belirli bir saatinde ve gününde çalışacak şekilde yapılandırabilirsiniz . Zamanlanmış taramalar, günlük dosyaları gibi sık güncellenen dosyaları izlerken uyarı taşmasını önler.

FIM modülü varsayılan olarak her 12 saatte bir *(43200 saniye) tarama çalıştırır. Aşağıdaki yapılandırma örneğinde, FIM modülünün her 15 dakikada bir (900 saniye)* tarama çalıştıracak şekilde nasıl ayarlanacağını görebilirsiniz .

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

```
<syscheck>
  <frequency>900</frequency>
</syscheck>
```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

   Alternatif olarak, scan_time ve scan_day seçeneklerini kullanarak taramaları planlayabilirsiniz . Bu seçenekleri kullanarak FIM'i yapılandırmak, FIM taramalarını iş saatleri dışında ayarlamanıza yardımcı olur.

Aşağıdaki yapılandırma örneği, belirtilen dizinlerin taramalarının her cumartesi *saat 22:00'da* nasıl çalıştırılacağını göstermektedir.

3. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

   ```
   <syscheck>
     <scan_time>10pm</scan_time>
     <scan_day>saturday</scan_day>
   </syscheck>
   ```

4. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

Dosya Değerlerindeki Değişiklikleri Bildir

Öznitelik `report_changes`, FIM modülünün bir metin dosyasında değiştirilen tam içeriği bildirmesine olanak tanır. Bu, izlenen bir dosyaya eklenen veya silinen metni kaydeder. Bu işlevi, dizin `report_changes` seçeneklerinin özniteliğini etkinleştirerek yapılandırabilirsiniz . Bu öznitelik için izin verilen değerler ve'dir . Windows, macOS ve Linux uç noktalarında hem dizinlerle hem de tek tek dosyalarla çalışır.`yesno`

Bu seçeneği etkinleştirdiğinizde özniteliği dikkatli kullanmalısınız `report_changes`. Wazuh, izlenen her dosyayı özel bir konuma kopyalayarak depolama kullanımını artırır. Dosyaların kopyasını şu adreste bulabilirsiniz:

- `/var/ossec/queue/diff/local/` Linux'ta.
- `Library/Ossec/queue/diff/local/`macOS'ta.
- `C:\Program Files (x86)\ossec-agent\queue\diff\local\`Windows'ta.

Aşağıda, FIM modülünün dosya değişikliklerini bildirecek şekilde nasıl yapılandırılacağını görebilirsiniz. `<FILEPATH_OF_MONITORED_FILE>`Kendi dosya yolunuzla değiştirin.

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

```
<syscheck>
  <directories check_all="yes" report_changes="yes"><FILEPATH_OF_MONITORED_FILE></directories>
</syscheck>
```

2. Yapılandırma değişikliklerini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

   `report_changes`Aşağıdaki yapılandırma örneğinde, dizindeki tüm dosyalar için özniteliğin nasıl kullanılacağını görebilirsiniz `<FILEPATH_OF_MONITORED_DIRECTORY>`. FIM modülünün dosyaya tam içerik değişikliklerini bildirmesini nasıl önleyeceğinizi görebilirsiniz . Kendi dosya yolunuzla `<FILEPATH_OF_MONITORED_DIRECTORY>/private.txt` değiştirin .`<FILEPATH_OF_MONITORED_DIRECTORY>`

   Seçeneği kullanırken , bir istisna oluşturmak için nodiff`report_changes` seçeneğini kullanabilirsiniz . Bu seçenek dosyanın değişikliklerini uyarır ancak Wazuh FIM modülünün bir metin dosyasında değiştirilen tam içeriği bildirmesini engeller. nodiff seçeneğini kullanmak, dosya içeriği değişikliklerini uyarılar aracılığıyla göndererek oluşabilecek veri sızıntısını önler.

3. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

```
<syscheck>
  <directories check_all="yes" report_changes="yes"><FILEPATH_OF_MONITORED_DIRECTORY></directorie
  <nodiff><FILEPATH_OF_MONITORED_DIRECTORY>/private.txt</nodiff>
</syscheck>
```

4. Yapılandırma değişikliklerini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

Hariç Tutmaların Eklenmesi

FIM modülünü, aşağıdaki iki yöntemden birini kullanarak belirli dosya ve dizinlere ilişkin uyarıları yoksayacak şekilde yapılandırabilirsiniz:

Yoksay Seçeneğini Kullanma

Bir yolu yoksaymak için yoksay seçeneğini kullanabilirsiniz . Satır başına bir dosya veya dizin girişine izin verir. Ancak, birden fazla yol için dışlamalar eklemek için birden fazla satır kullanabilirsiniz.

Bu örnekte, FIM modülünün bir dosya yolunu yoksayacak şekilde nasıl yapılandırılacağını görebilirsiniz. Bu ayrıca dosya uzantıları `.log`ve . için regex eşleşmesini de yoksayar. Kendi dosya yollarınızla `.tmp`değiştirin .`<FILEPATH_OF_MONITORED_FILE>`

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin:
   - Linux:`/var/ossec/etc/ossec.conf`
   - Pencereler:`C:\Program Files (x86)\ossec-agent\ossec.conf`
   - macOS:`/Library/Ossec/etc/ossec.conf`

   ```
   <syscheck>
     <ignore><FILEPATH_OF_MONITORED_FILE></ignore>
     <ignore type="sregex">.log$|.tmp$</ignore>
   </syscheck>
   ```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:
   - Linux:`systemctl restart wazuh-agent`
   - Pencereler:`Restart-Service -Name wazuh`
   - macOS:`/Library/Ossec/bin/wazuh-control restart`

Özel Kuralları Kullanma

Alternatif bir yöntem, uyarı seviyesi 0 kurallarını kullanmaktır. Bu yöntem, FIM modülü tarafından taranan belirli dosya ve dizinlerin uyarılarını yok sayar. Seviye 0 kuralları için uyarılar sessizdir ve Wazuh sunucusu bunları bildirmez.

`/var/www/htdocs/`Aşağıdaki yapılandırma örneğinde, bir Linux uç noktasındaki dizinin nasıl izleneceğini ve dosya için sessiz uyarıların nasıl kullanılacağını görebilirsiniz `/var/www/htdocs/private.html`.

Linux Uç Noktası

1. Wazuh aracı yapılandırma dosyasına aşağıdaki ayarları ekleyin `/var/ossec/etc/ossec.conf`:

   ```
   <syscheck>
     <directories>/var/www/htdocs</directories>
   </syscheck>
   ```

2. Herhangi bir yapılandırma değişikliğini uygulamak için Wazuh aracısını yönetici ayrıcalığıyla yeniden başlatın:

```
systemctl restart wazuh-agent
```

Wazuh Sunucusu

1. Wazuh sunucusundaki dizinde `fim_ignore.xml`dosyayı oluşturun :`/var/ossec/etc/rules/`

   ```
   touch /var/ossec/etc/rules/fim_ignore.xml
   ```

2. Dosyaya aşağıdaki kuralları ekleyin `fim_ignore.xml`:

   ```
   <group name="syscheck">
     <rule id="100345" level="0">
       <if_group>syscheck</if_group>
       <field name="file">/var/www/htdocs/private.html</field>
       <description>Ignore changes to $(file)</description>
     </rule>
   </group>
   ```

   Kural, `/var/www/htdocs/private.html`dosya için FIM uyarısını susturur.

3. Yapılandırma değişikliklerini uygulamak için Wazuh yöneticisini yeniden başlatın:

   ```
   systemctl restart wazuh-manager
   ```

Revision #4
Created 23 December 2024 18:24:24 by Ayşegül Sarıkaya
Updated 31 December 2024 17:54:16 by Ayşegül Sarıkaya