Wazuh Yöneticisi

Wazuh yöneticisi veri analizi ve uyarılardan sorumludur. Uyarıları syslog, e-postalar veya entegre harici API'ler aracılığıyla iletebilir. Wazuh'un veri analizini nasıl gerçekleştirdiği hakkında daha fazla bilgi için veri analizi belgelerine bakın.

Wazuh yöneticisi, çeşitli işlevlerden sorumlu olan çeşitli hizmetler ve bileşenlerden oluşur. Bunlara yeni Wazuh aracılarını kaydetme, güvenlik olaylarını toplama, günlükleri kod çözme, kuralları değerlendirme ve uyarı verme dahildir. Ayrıca Wazuh aracısının kimliklerini doğrulama ve Wazuh aracısı ile Wazuh sunucusu arasındaki iletişimleri şifreleme gibi diğer işlevlerden de sorumludur.

Acente Kayıt Hizmeti

Temsilci kayıt hizmeti, Wazuh temsilcilerini Wazuh yöneticisine kaydetmek için kullanılır. Kayıt hizmeti, Wazuh temsilcilerinin kaydını basitleştirir ve Wazuh yöneticisiyle güvenli bir şekilde iletişim kurmak üzere doğru bir şekilde kimlik doğrulaması yapılıp yapılandırılmalarını sağlar.

Bir uç noktaya bir Wazuh aracı yüklendiğinde ve başlatıldığında, kayıt sürecini başlatmak için otomatik olarak Wazuh yöneticisiyle iletişime geçer. Wazuh yöneticisi, Wazuh aracısıyla iletişimini şifreleyen benzersiz bir kimlik doğrulama anahtarı üretir. Kayıt süreci için parola kimlik doğrulaması, Wazuh yöneticisi kimlik doğrulaması ve Wazuh aracı kimlik doğrulaması gibi ek güvenlik önlemleri yapılandırabilirsiniz. Kayıt süreci hakkında daha fazla bilgi için Wazuh aracı kaydıyla ilgili belgelere bakın.

Yapılandırma

Aşağıdaki blok , Wazuh sunucusunun dosyasındaki `<auth>`varsayılan aracı kayıt hizmeti yapılandırmasıdır :`/var/ossec/etc/ossec.conf`

```
<auth>
  <disabled>no</disabled>
  <remote_enrollment>yes</remote_enrollment>
  <port>1515</port>
  <use_source_ip>no</use_source_ip>
  <force>
    <enabled>yes</enabled>
    <disconnected_time enabled="yes">1h</disconnected_time>
    <after_registration_time>1h</after_registration_time>
    <key_mismatch>yes</key_mismatch>
  </force>
```

```
  <purge>yes</purge>
  <use_password>no</use_password>
  <ciphers>HIGH:!ADH:!EXP:!MD5:!RC4:!3DES:!CAMELLIA:@STRENGTH</ciphers>
  <!-- <ssl_agent_ca></ssl_agent_ca> -->
  <ssl_verify_host>no</ssl_verify_host>
  <ssl_manager_cert>etc/sslmanager.cert</ssl_manager_cert>
  <ssl_manager_key>etc/sslmanager.key</ssl_manager_key>
  <ssl_auto_negotiate>no</ssl_auto_negotiate>
</auth>
```

Nerede:

- `<disabled>`Wazuh aracısının Wazuh yöneticisine kaydolma ve kimlik doğrulama işlemini etkinleştirir veya devre dışı bırakır. Varsayılan değer 'dir `no`. İzin verilen değerler `yes`ve 'dir `no`.
- `<remote_enrollment>`Wazuh yöneticisinin varsayılan olarak 1515 numaralı bağlantı noktasında TLS şifrelemesi kullanarak yeni Wazuh aracılarından gelen bağlantıları kabul etmesini sağlar. Varsayılan değer 'dir `yes`. İzin verilen değerler `yes`ve 'dir `no`.
- `<port>`Bağlantıları dinlemek için TCP bağlantı noktası numarasını belirtir. Varsayılan değer 'dir . İzin verilen değer ve `1515`arasındaki herhangi bir bağlantı noktası numarasıdır .`65535`
- `<use_source_ip>`istemcinin kaynak IP adresinin mi yoksa "herhangi biri"nin mi kullanılacağını tanımlar. İzin verilen değerler ve'dir `yes`. `no`Değer hayır olduğunda, kayıt için kullanılan kaynak IP değişse bile Wazuh aracısı Wazuh yöneticisine bağlanabilir. Ancak değer evet olduğunda, kaynak IP adresi değişse bile Wazuh aracısı Wazuh yöneticisine bağlanamaz.
- `<force>`Wazuh aracısının etiketi içinde yeniden kaydı için yapılandırılacak seçenekleri belirtir. Yeniden kaydın başarılı olması için tüm koşulların karşılanması gerekir. Aşağıdaki seçenekler, seçeneğin ayarlarını tanımlar `force`:
  - `<enabled>`yinelenen bir ad veya IP adresi varsa bir Wazuh aracısının eklenmesinin zorlanıp zorlanmayacağını belirtir. Eğer öyleyse `enabled`, aynı ad veya IP adresine sahip eski Wazuh aracısını kaldıracaktır. Varsayılan değer 'dir `yes`. Olası değerler `yes` ve 'dir `no`.
  - `<disconnected_time>`yalnızca ayarda yapılandırılan değerden daha uzun süre bağlantısı kesilmiş olan Wazuh aracıları için bir değiştirme yapılıp yapılmayacağını belirtir. Varsayılan değer `1h`(bir saat)'tir. İzin verilen değer sıfırdan büyük veya sıfıra eşit herhangi bir sayıdır. `s`, `h`, `m`, ve gibi soneklerin `d`saniye, saat, dakika ve günü temsil etmesine izin verir. Öznitelik ayarı `enabled`varsayılan değerine sahiptir `yes`, yani değiştirme yalnızca belirtilen bağlantı kesme süresi aşıldıktan sonra gerçekleşir. Etkin özniteliğin `yes`ve olmak üzere iki olasılığı vardır `no`.
  - `<after_registration_time>`Wazuh aracısı değişiminin yalnızca Wazuh aracısı kaydının ayarda yapılandırılan değerden büyük olması durumunda gerçekleştirileceğini belirtir. Varsayılan değer 'dir `1h`. İzin verilen değer sıfırdan büyük veya ona eşit herhangi bir sayıdır. `s`, `h`, `m`, ve gibi soneklerin `d`saniye, saat, dakika ve günü temsil etmesine izin verir.
  - `<key_mismatch>`Wazuh aracısının elinde tuttuğu anahtar, yönetici tarafından kaydedilen anahtardan farklı olduğunda Wazuh aracısının değiştirilmesinin gerçekleştiğini tanımlar. Varsayılan değer 'dir `yes`. Olası değerler `yes`ve 'dir `no`.

- `<purge>`Wazuh aracıları kaldırıldığında istemci anahtarlarının silinip silinmeyeceğini belirtir. Değer olduğunda `no`, kaldırılan Wazuh aracıları kaldırılmış olarak işaretlenen istemci anahtarları dosyasında kalır. Değer olarak ayarlandığında `yes`, istemci anahtarları dosyası temizlenir. Varsayılan değer 'dir `yes`. Olası değerler `yes`ve 'dir `no`.
- `<use_password>`paylaşımlı parola kimlik doğrulamasının kullanımını belirler. Değer olduğunda `no`, bu seçenek devre dışıdır. Değer olarak ayarlandığında `yes`, dosyadan paylaşımlı bir parola okunur `/var/ossec/etc/authd.pass`. Bu dosya mevcut değilse, rastgele bir parola oluşturulur ve `/var/ossec/logs/ossec.log`Wazuh sunucusundaki dosyada saklanır.
- `<ciphers>`SSL kullanarak ağ iletişimi için şifrelerin listesini ayarlar. Varsayılan değer `HIGH:!ADH:!EXP:!MD5:!RC4:!3DES:!CAMELLIA:@STRENGTH`.
- `<ssl_agent_ca>`istemcileri doğrulamak için kullanılan CA sertifikasına giden yolu belirtir. Wazuh kurulum dizini altındaki bağıl yol veya tam yol olarak adlandırılabilir. Olası değer herhangi bir geçerli yoldur.
- `<ssl_verify_host>`CA sertifikası belirtildiğinde kaynak ana bilgisayar doğrulamasını açar ve kapatır. İstemci kaynak IP adresi Ortak Ad alanı kullanılarak doğrulanacaktır. Varsayılan değer 'dir `no`. İzin verilen değerler `yes`ve 'dir `no`.
- `<ssl_manager_cert>`sunucu SSL sertifikasına giden yolu belirtir. Wazuh kurulum dizinindeki bağıl yol veya tam yol olarak adlandırılabilir. Varsayılan değer etc/sslmanager.cert'dir. Olası değer herhangi bir geçerli yoldur.
- `<ssl_manager_key>`sunucunun SSL anahtarına giden yolu belirtir. Wazuh kurulum dizininin altındaki bağıl yol veya tam yol olarak adlandırılabilir. Varsayılan değer etc/sslmanager.key'dir. Olası değer herhangi bir geçerli yoldur.
- `<ssl_auto_negotiate>`SSL/TLS yönteminin otomatik olarak seçilip seçilmeyeceğini değiştirir. Varsayılan olarak yalnızca TLS v1.2'ye izin verilir. olarak ayarlandığında `yes`, sistem istemciyle en güvenli ortak yöntemi müzakere eder. Yöneticinin TLS v1.2'yi desteklemediği eski sistemlerde, bu seçenek otomatik olarak etkinleştirilir. Varsayılan değer 'dir `no`. İzin verilen değerler `yes`ve 'dir `no`.

Yapılandırma dosyasında değişiklik yaptığınızda, aşağıdaki komutu kullanarak komut satırı arayüzü üzerinden Wazuh yöneticisini yeniden başlatın:

Systemd

```
systemctl restart wazuh-manager
```

SysV Başlatma

```
service wazuh-manager restart
```

Ajan Bağlantı Hizmeti

Aracı bağlantı hizmeti, kalıcı ve güvenli bir iletişim kanalı kurmak ve sürdürmek için Wazuh aracılarından gelen olayları dinler. Wazuh aracısı, güvenlik verilerini analiz için Wazuh yöneticisine

göndermek için bu güvenli kanalı kullanır. Varsayılan olarak, hizmet `TCP`Wazuh aracısı ile Wazuh yöneticisi arasındaki iletişimi güvence altına almak için protokolü kullanır.

Yapılandırma

Aşağıdaki blok Wazuh sunucu yapılandırma dosyasındaki varsayılan bağlantı hizmeti yapılandırmasıdır `/var/ossec/etc/ossec.conf`:

```
<ossec_config>
  <remote>
    <connection>secure</connection>
    <port>1514</port>
    <protocol>tcp</protocol>
    <queue_size>131072</queue_size>
  </remote>
</ossec_config>
```

Nerede:

- `<connection>`kabul edilecek gelen bağlantının türünü belirtir. Varsayılan değer güvenlidir. İzin verilen değerler `secure`ve ' dir `syslog`.
- `<port>`olayları dinlemek için kullanılacak portu belirtir. Varsayılan port değeri `1514`güvenli bağlantı ve syslog bağlantısı içindir . İzin verilen değer ve `514`arasındaki herhangi bir port numarasıdır .`165535`
- `<protocol>`bağlantı için kullanılacak protokolü belirtir. Varsayılan değer 'dir `tcp`. İzin verilen değerler `tcp`ve 'dir `udp`.
- `<queue_size>`Uzak daemon kuyruğunun kapasitesini Wazuh aracı olaylarının sayısı olarak ayarlamanıza olanak tanır. Varsayılan değer 'dir . İzin verilen değer ile `131072`arasında bir tam sayıdır . Uzak kuyruk yalnızca Wazuh aracı olayları için kullanılabilir, syslog olayları için kullanılamaz. Bu seçenek yalnızca bağlantı güvenli olarak ayarlandığında çalışır. Bu yapılandırma ayarı hakkında daha fazla bilgi edinmek için Wazuh kuyruğu ile ilgili belgelerimize bakın .`1262144`

Değişiklikler yapıldıysa, değişiklikleri uygulamak için aşağıdaki komutla Wazuh yöneticisini komut satırı arayüzü üzerinden yeniden başlatın:

Systemd

```
systemctl restart wazuh-manager
```

SysV Başlatma

```
service wazuh-manager restart
```

Örneğin, bir Windows uç noktasındaki (IP adresi 192.168.71.125) bir Wazuh yöneticisine (IP adresi 192.168.71.203) bir Wazuh aracısının kaydı sırasında netstat kullanarak bağlantı hizmetinin çalışmasını doğrulayabilirsiniz. Ayrıca, herhangi bir Wazuh destekli uç noktada çalışan bir Wazuh aracısı, güvenlik olaylarını port üzerindeki Wazuh yöneticisine iletir . Yukarıdaki aracı bağlantı hizmeti yapılandırma`1514` bölümünde ayrıntılı olarak açıklanan yapılandırmayı kullanır .

Wazuh yöneticisi ile Wazuh aracısı arasındaki bağlantı hizmetinin çalışmasını doğrulamak için aşağıdaki adımları gerçekleştirin:

1. Windows uç noktasında komut istemini başlatın ve uç noktadaki bağlantıları listelemek için şu komutları çalıştırın:`netstat -a`

   ```
   netstat -a
   ```

   Output

   ```
   C:\Users\Tony>netstat -a

   Active Connections

     Proto  Local Address          Foreign Address        State
       TCP    192.168.71.125:51787   a23-53-42-162:https    ESTABLISHED
     TCP    192.168.71.125:51788   a-0003:https           ESTABLISHED
     TCP    192.168.71.125:51789   a-0003:https           ESTABLISHED
     TCP    192.168.71.125:51790   a23-53-42-162:https    ESTABLISHED
     TCP    192.168.71.125:51791   192.168.71.203:1514    SYN_SENT
   ```

   `192.168.71.125`IP adresine sahip Windows uç noktasının bir TCP paketi gönderdiğini ve porttaki `SYN_SENT`IP adresine sahip Wazuh sunucusuyla bağlantı kurmayı beklediğini görebiliyoruz .`192.168.71.203``1514`

2. `netstat`Wazuh sunucusunun Windows 10 uç noktasıyla ne zaman bağlantı kurduğunu görüntülemek için komutu çalıştırın.

   ```
   netstat
   ```

   Output

   ```
   Active Connections

     Proto  Local Address          Foreign Address        State
     TCP    192.168.71.125:3389    192.168.71.1:25743     ESTABLISHED
     TCP    192.168.71.125:51572   a23-64-12-19:https     CLOSE_WAIT
     TCP    192.168.71.125:51573   192.229.221.95:http    CLOSE_WAIT
     TCP    192.168.71.125:51694   192.168.71.203:1514    ESTABLISHED
     TCP    192.168.71.125:51699   192.168.20.103:ms-do   SYN_SENT
     TCP    192.168.71.125:51701   192.168.20.101:ms-do   SYN_SENT
     TCP    192.168.71.125:51703   20.231.121.79:http     SYN_SENT
   ```

```
TCP    192.168.71.125:51704   192.168.20.125:ms-do   SYN_SENT
```

IP adresine sahip Windows uç noktasının , port üzerindeki `192.168.71.125`IP adresine sahip Wazuh sunucusuna bağlı olduğunu görebiliyoruz .`192.168.71.20`31514

Analiz Motoru

Wazuh analiz motoru, Windows olayları, SSH günlükleri, web sunucusu günlükleri ve diğerleri gibi çeşitli günlük türlerindeki verileri analiz eder. İşlenen bilgi türünü belirlemek için kod çözücüleri ve kod çözülen olaydaki belirli kalıpları belirlemek için kuralları kullanır. Bu kurallar, bir IP adresini engelleme ve kötü amaçlı yazılımları kaldırma gibi uyarıları ve yanıt eylemlerini tetikleyebilir.

Veri Kaynakları

Wazuh, çeşitli kaynaklardan günlükler toplayarak BT altyapınızın tüm yönlerinin kapsamlı bir şekilde izlenmesine olanak tanır. Bu, Wazuh'un karmaşık tehditleri tespit etmesini, güvenlik açığı riskini azaltmasını, güvenlik politikalarına uyumu sağlamasını ve belirlenen güvenlik olaylarına hızla yanıt vermesini sağlar. Aşağıda Wazuh tarafından desteklenen bazı yaygın veri kaynakları verilmiştir:

- **İşletim sistemi günlükleri : Wazuh,** Windows , Linux ve macOS gibi çeşitli işletim sistemleri tarafından oluşturulan günlükleri toplar . Syslog, auditd, uygulama günlükleri ve diğerleri dahil olmak üzere Linux uç noktalarından çeşitli günlükler toplayabilir. Windows uç noktalarında, Wazuh varsayılan olarak Sistem, Uygulamalar ve Güvenlik olay kanallarından Windows olay günlüklerini toplar. Wazuh, macOS birleşik günlük sistemi (ULS) kullanarak macOS uç noktalarındaki günlükleri toplar. macOS ULS, tüm sistem düzeylerinde günlüklerin yönetimini ve depolanmasını merkezileştirir.
- **Syslog olayları** : Wazuh , Linux/Unix sistemleri ve Wazuh aracısı kurulumu gerektirmeyen ağ aygıtları da dahil olmak üzere çeşitli syslog özellikli aygıtlardan günlükleri toplar.
- **Aracısız izleme** : Wazuh aracısız izleme yeteneği, aracı kurulumunu desteklemeyen uç noktaları izler. Uç nokta ile Wazuh sunucusu arasında bir SSH bağlantısı gerektirir. Bu yetenek, dosyaların, dizinlerin veya yapılandırmaların izlenmesini ve uç noktada komutların çalıştırılmasını sağlar.
- **Bulut sağlayıcı günlükleri : Wazuh,** AWS , Azure , Google Cloud ve Office 365 gibi bulut hizmet sağlayıcılarından doğrudan günlükleri ve olayları toplayarak bulut altyapısını izler . Bunlara EC2 örnekleri, S3 kovaları, Azure VM'leri ve daha fazlası gibi bulut hizmetlerinden gelen günlükler dahildir.
- **Özel günlükler : Wazuh'u** VirusTotal , Windows Defender , ClamAV ve daha fazlası dahil olmak üzere çeşitli uygulamalardan ve üçüncü taraf güvenlik araçlarından günlükleri toplayacak ve ayrıştıracak şekilde yapılandırabilirsiniz .

Kod Çözme

Kod çözme, farklı veri kaynaklarından gelen günlükler gibi yapılandırılmış veya yapılandırılmamış verileri, izleme ve uyarı için kullanılabilecek anlamlı bilgiler çıkarmak için analiz etme sürecidir. Wazuh'ta kod çözmenin temel amacı, ham verileri Wazuh yöneticisinin yorumlayabileceği ve işleyebileceği bir biçime dönüştürmektir. İki süreci içerir:

- **Ön kod çözme aşaması** : Bu aşamada, günlük analiz motoru günlük başlığından zaman damgası, ana bilgisayar adı ve program adı gibi syslog benzeri bilgileri çıkarır. Ön kod çözme aşaması günlük yapısını basitleştirir ve daha ileri analiz için hazırlar. Ön kod çözme sürecini göstermek için aşağıdaki örnek günlük girişini göz önünde bulundurun:

  ```
  Feb 14 12:19:04 192.168.1.1 sshd[25474]: Accepted password for Stephen from 192.168.1.133 port 49765 s
  ```

  Ön kod çözme aşamasını göstermek için Wazuh Logtest aracını kullanıyoruz. Wazuh sunucusunda aşağıdaki adımları gerçekleştirin:
  1. `/var/ossec/bin/wazuh-logtest`Wazuh sunucusunda komut satırından çalıştırın
  2. Yukarıdaki örnek günlüğü kopyalayıp yapıştırın ve enter'a basın.

  Ön kod çözme aşaması sonrasında elde edilen bilgiler aşağıda gösterilmektedir:

  ```
  Starting wazuh-logtest v4.8.0
  Type one log per line

  Feb 14 12:19:04 192.168.1.1 sshd[25474]: Accepted password for Stephen from 192.168.1.133 port 49765 s

  **Phase 1: Completed pre-decoding.
        full event: 'Feb 14 12:19:04 192.168.1.1 sshd[25474]: Accepted password for Stephen from 192.168.1.1
        timestamp: 'Feb 14 12:19:04'
        hostname: '192.168.1.1'
        program_name: 'sshd'
  ```

- **Kod çözme** : Bu aşamada, Wazuh analiz motoru günlükle eşleşen bir kod çözücü uygular. Kod çözücüler, günlüklerde bulunan kullanıcı adları, IP adresleri, hata kodları, URL'ler ve diğer ilgili bilgiler gibi alanları ayıklar. Aşağıdaki kod çözücüler örnek günlükle eşleşir. Bu kod çözücüler `/var/ossec/rulesets/decoders/0310-ssh_decoders.xml`Wazuh sunucusundaki dosyadadır:

  ```
  <decoder name="sshd">
    <program_name>^sshd</program_name>
  </decoder>

  <decoder name="sshd-success">
    <parent>sshd</parent>
    <prematch>^Accepted</prematch>
    <regex offset="after_prematch">^ \S+ for (\S+) from (\S+) port (\S+)</regex>
  ```

```
  <order>user, srcip, srcport</order>
  <fts>name, user, location</fts>
</decoder>
```

Kod çözücü `sshd`program adıyla eşleşirken `sshd`, kod çözücü örnek günlükten , , ve `ssh-success`öğelerini çıkarır .`Stephen``192.168.1.133``49765`

Kod çözme aşamasını göstermek için Wazuh Logtest aracını kullanıyoruz. Wazuh sunucusunda aşağıdaki adımları gerçekleştirin:

1. `/var/ossec/bin/wazuh-logtest`Wazuh sunucusunda from komut satırını çalıştırın .
2. Yukarıdaki örnek günlüğü kopyalayıp yapıştırın ve enter'a basın.

Kod çözme aşaması sonucunda elde edilen bilgiler aşağıda gösterilmektedir:

```
Starting wazuh-logtest v4.7.5
Type one log per line

Feb 14 12:19:04 192.168.1.1 sshd[25474]: Accepted password for Stephen from 192.168.1.133 port 49765 s

**Phase 1: Completed pre-decoding.
        full event: 'Feb 14 12:19:04 192.168.1.1 sshd[25474]: Accepted password for Stephen from 192.168.1.1
        timestamp: 'Feb 14 12:19:04'
        hostname: '192.168.1.1'
        program_name: 'sshd'

**Phase 2: Completed decoding.
        name: 'sshd'
        parent: 'sshd'
        dstuser: 'Stephen'
        srcip: '192.168.1.133'
        srcport: '49765'
```

Kural Değerlendirmesi ve Uyarı

Günlük çözüldükten sonra, Wazuh yöneticisi bunu bir kural setiyle karşılaştırır. Wazuh kural setleri XML dosyalarında tanımlanır ve farklı izleme ihtiyaçlarına uyacak şekilde özelleştirilebilir. Bu kurallar, karşılandığında uyarıları tetikleyen koşulları belirtir. `5715`Aşağıdaki kural, önceki bölümdeki örnek günlükle eşleşir. Bu kural, `/var/ossec/ruleset/rules/0095-sshd_rules.xml`Wazuh sunucusundaki dosyadadır.

```
<rule id="5715" level="3">
  <if_sid>5700</if_sid>
  <match>^Accepted|authenticated.$</match>
  <description>sshd: authentication success.</description>
  <group>authentication_success,pci_dss_10.2.5,</group>
</rule>
```

Nerede:

- `<rule id="5715" level="3">`kural kimliğini `5715`ve kural düzeyini olarak belirtir `3`. Kural kimliği kural için benzersiz bir tanımlayıcıdır, düzey ise kural eşleştiğinde olayın önem düzeyini temsil eder.
- `<if_sid>5700</if_sid>`ID'li başka bir kurala bağımlılığı belirtir `5700`. Kural yalnızca daha önce eşleşmişse değerlendirilecektir `5700`.
- `<match>^Accepted|authenticated.$</match>`ile başlayan `Accepted`veya biten herhangi bir günlük girişiyle eşleşir `authenticated.`.
- `<description>sshd: authentication success.</description>`kuralın neyi algıladığını açıklar. Bu durumda, başarılı bir SSH kimlik doğrulamasını gösterir.
- `<group>authentication_success,pci_dss_10.2.5,</group>`kuralı `authentication_success`ve `pci_dss_10.2.5` gruplarına atar.

Varsayılan olarak, Wazuh sunucusu 2'nin üzerinde bir seviyeye sahip herhangi bir kural için uyarılar üretir. Bu senaryoda, kural seviyesi 3 olduğu için günlük bir uyarıyı tetikler ve bu Wazuh panosunda görünür olacaktır.

Varsayılan olarak desteklenmeyen günlükleri analiz etmek için özel kod çözücüler ve kurallar oluşturabilirsiniz. Özel kurallar ve kod çözücülerin nasıl oluşturulacağını öğrenmek için özel kurallar ve özel kod çözücüler belgelerine bakın.

Revision #9
Created 28 December 2024 00:34:13 by Ayşegül Sarıkaya
Updated 31 December 2024 13:26:28 by Ayşegül Sarıkaya