

Laboratuvar Ortamında Wazuh(OVA) Kurulumu ve Özellik İncelemesi

İçindekiler

LABORATUVAR ORTAMINDA WAZUH(OVA) KURULUMU VE ÖZELLİK İNCELEMESİ	1
LABORATUVAR ORTAMI KURULUMU	2
Wazuh(OVA) ve Agent Kurulumu	2
ÖZET	5
WAZUH ÖZELLİKLERİ	6
File Integrity Monitoring	6
Yapılandırılma-Uygulama	6
Active Response	8
Security Configuration Assessment	10
Yapılandırılma-Uygulama	11
Vulnerability detection	12
Yapılandırılma-Uygulama	13
Log Data Collection	14
Yapılandırılma-Uygulama	15
Malware Detection	16
Yapılandırılma-Uygulama	17
Troubleshootings	19
Sonuç	19
REFERANSLAR	20

Okan UZUN

okanuzun42@gmail.com

LABORATUVAR ORTAMI KURULUMU

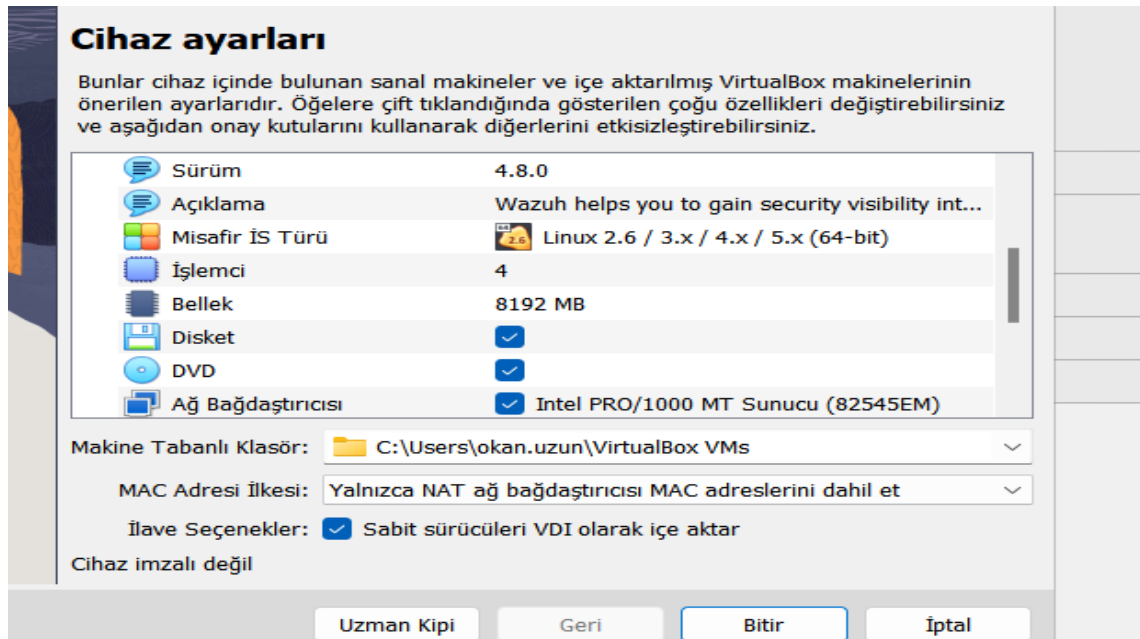
Laboratuvar ortamı kurulması adına bilgi teknolojileri alanında çalışan çoğu kişinin de bileceği sanallaştırma uygulamasından yararlanıldı; Virtualbox. Ortam oluşturulması için Wazuh(OVA) dosyası, Kali Linux, Debian ve Windows 8.1 gibi işletim sistemleri kullanıldı.

Bu içeriğin laboratuvar ortamı kurulumunda yalnızca Wazuh(OVA) kurulumundan bahsedilecektir. Diğer işletim sistemlerinin kurulumu için <https://www.howtogeek.com/796988/how-to-install-linux-in-virtualbox/>, <https://www.wikihow.com/Install-Windows-8-in-VirtualBox> sayfaları ziyaret edilebilir.

Wazuh(OVA) ve Agent Kurulumu

Wazuh, açık kaynak kodlu bir güvenlik bilgi ve olay yönetimi platformudur. Wazuh, bilgisayar ağlarında güvenlik olaylarını izlemek, tehditleri tespit etmek ve güvenlik açıklarını yönetmek için kullanılır. OVA (Open Virtual Appliance) dosyası, sanallaştırma ortamlarında (örneğin, VMware, VirtualBox gibi) kullanılmak üzere hazırlanmış bir sanal makine imajıdır. Bu dosya genellikle bir sanal makine için disk imajını, yapılandırma ayarlarını ve diğer gerekli bileşenleri içerir. Wazuh'un OVA dosyası, Wazuh platformunu kolayca sanal bir makine üzerinde çalıştırmak isteyen kullanıcılar için hazırlanmıştır.

- 1- İlk olarak [Virtual Machine \(OVA\) - Installation alternatives \(wazuh.com\)](https://wazuh.com/docs/quickstart/quickstart-ova) sitesinden ilgili dosyası indirilir. İndirilen bu dosyaya tıklanarak VirtualBox otomatik açılır ve kurulum ayarları için hazır hale gelir.



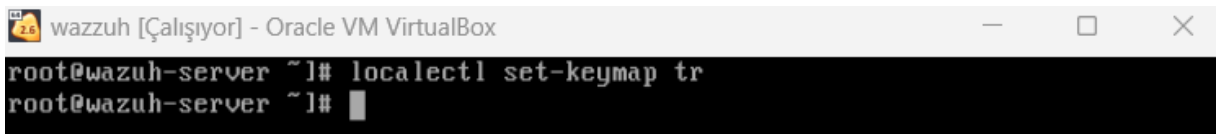
Şekil 1

- 2- Kurulum için 4096Mb ana bellek ve 2 işlemci fazlasıyla yeterli olacaktır. Görüntü belleğinin 128MB olması ve grafik denetleyicisinin “VMSVGA” olarak seçili olması gerekmektedir. Ayrıca kullanılacak diğer işletim sistemleriyle aynı ağda olması adına ağ ayarından NAT Network seçildi.



Şekil 2

- 3- Yukarıdaki ayarlamaların ardından terminal ekranı açıldı, “wazuh-user” kullanıcı adı ve “wazuh” şifresiyle oturum açıldı. İlk olarak klavye Türkçe olarak ayarlandı. Daha sonra ara yüz için ip adresi öğrenildi. Agent kurulumları için web ara yüzüne erişim sağlandı.



- 4- Agent kurulumu için NAT Network üzerindeki işletim sistemlerin herhangi birinden ara yüzü erişimi sağlanır. Burada Debian üzerinde işlem yapılmıştır. Dashboard erişimi olduktan sonra “add agent” kısmına gidilir. Burada ilgili işletim sistemi seçilir. Burada Wazuh seçilen işletim sistemine göre çalıştırılması gereken komutları oluşturur.

LINUX

☒ RPM amd64 ☐ RPM aarch64

☐ DEB amd64 ☐ DEB aarch64

WINDOWS

☐ MSI 32/64 bits

macOS

☐ Intel ☐ Apple silicon

For additional systems and architectures, please check our [documentation](#).

Server address:

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FQDN).

Assign a server address

10.0.2.16

☒ Remember server address

Şekil 3

- 5- Agent kurulumu için ilgili komutlar Wazuh tarafından oluşturuldu. Komutları çalıştırırken Windowsda PowerShell üzerinde “Administrator” olarak Linux da ise “root” olarak çalıştırılması gereklidir.

4 Run the following commands to download and install the agent:

```
curl -o wazuh-agent-4.8.0-1.x86_64.rpm https://packages.wazuh.com/4.x/yum/wazuh-agent-4.8.0-1.x86_64.rpm && sudo WAZUH_MANAGER='10.0.2.16' rpm -ihv wazuh-agent-4.8.0-1.x86_64.rpm
```

Requirements

- You will need administrator privileges to perform this installation.
- Shell Bash is required.

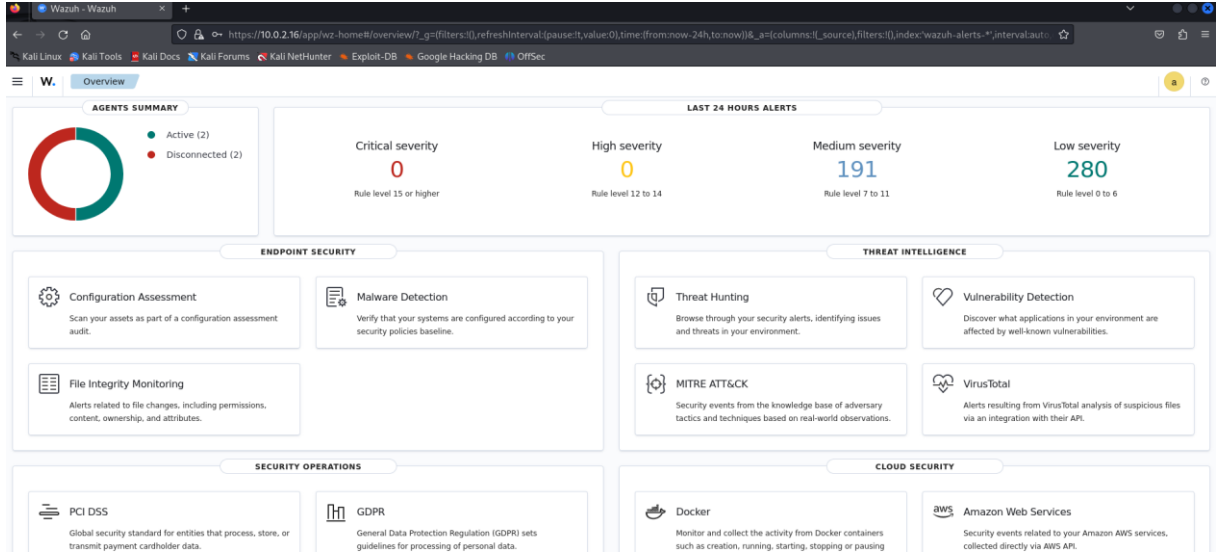
Keep in mind you need to run this command in a Shell Bash terminal.

5 Start the agent:

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
```

Şekil 4

- 6- Daha sonra agent durumlarının kontrolü için dashboard ekranına gidilir. Burada “Active” kısmında olanlar Windows ve Debian agentlarıdır. Dashboard ekranında yazılan farklı Wazuh özellikleri görülmektedir.



Şekil 5

ÖZET

Laboratuvar ortamı kurulumu için, öncelikle Virtualbox kullanılarak sanallaştırma yapıldı ve Wazuh'un OVA dosyası indirilerek Virtualbox'a yüklendi. Bu dosya, Wazuh'u sanal bir makine üzerinde çalıştırmak için hazırlanmıştı. Kurulum sırasında, sanal makine için 4096MB RAM ve 2 işlemci ayrıldı; görüntü belleği 128MB olarak ayarlandı ve Grafik denetleyicisi olarak "VMSVGA" seçildi. Ağ ayarları NAT Network olarak yapılandırıldı, böylece Wazuh diğer işletim sistemleriyle aynı ağda çalışabildi.

Sanal makine başlatıldıktan sonra terminal ekranı açılarak "wazuh-user" kullanıcı adı ve "wazuh" şifresiyle oturum açıldı. Web ara yüzüne erişim sağlandı ve agent kurulumları için gerekli adımlar atıldı. Debian üzerinde agent kurulumu gerçekleştirilerek, Wazuh'un sağladığı komutlar kullanılarak işletim sistemlerine agentlar kuruldu.

Son olarak, dashboard üzerinden agentların durumları kontrol edildi. Aktif olarak çalışan agentlar (Windows ve Debian) belirlendi ve dashboard üzerinde Wazuh'un çeşitli özellikleri incelendi.

WAZUH ÖZELLİKLERİ

File Integrity Monitoring

File Integrity Monitoring(FIM), agentın kurulu olduğu uç cihazlar(endpoint) üzerindeki dosyalarda gerçekleşen oluşturma, silme ve düzeltme durumlarını izleyen modüldür. Oluşan herhangi bir dosya eyleminde Wazuh dashboard kısmında görüntülenme sağlanır. Bu sayede kurumlar önemli dosyalar üzerinde gerçekleşecek olan eylemleri tespit edebilir.

FIM modülü periyodik olarak belirlenmiş olan dosyalar üzerinde izleme yapar. Bu izleme üzerine Wazuh agent belirlenmiş dosyanın eylemlerini raporlar. Herhangi bir tutarsızlıkta da uyarı verir.

Wazuh FIM modülü silme, oluşturma ve değiştirme FIM olaylarını toplayabilmek için 2 adet veri tabanı tutar. İlki uç cihazların hafızasında tutulan yerel [SQLite](#) tabanlı veri tabanıdır. Bu veri tabanı Windows işletim sisteminde "C:\Program Files (x86)\ossec-agent\queue\fim\db" yolunda tutulurken Linux da ise "/var/ossec/queue/fim/db" yolunda tutulur. Bir diğer veri tabanı ise Wazuh sunucusu üzerinde tutulan agent veri tabanıdır. Burada agent ID numarasına göre veri tabanı belirlenir. Bu veri tabanı da "/var/ossec/queue/db" yolunda tutulur.

FIM modülü, Wazuh agent ve Wazuh sunucu veri tabanını birbirleriyle senkronize eder. FIM sürekli olarak kendisine izlemesi için verilen dosya envanterini inceler. Senkronizasyon mekanizması Wazuh sunucusuna Wazuh agent üzerinden gelen dosya durum değişikliği durumlarını günceller.

Yapılandırılma-Uygulama

Wazuh FIM modülünün incelenmesi için Debian işletim sistemi üzerinden incelenmektedir. Hangi dosyaların izlenmesi gerektiği config dosyasında belirlenmelidir. Default ayarlarında çoğu izin seçili olarak gelmektedir.

- 1- Debian terminalde "/var/ossec/etc/ossec.conf" dosyası açılır. Config dosyaları XML diliyle oluşturulmuştur. [XML](#) dosyasında "<syscheck>" kısmında "<directories>" içine izlenmesi

istenilen dosya yolu belirtilir. Dosya değişikliği kim tarafından yapıldığını izlemek adına “whodata”, belirtilen dizinlerin altındaki tüm dosyalar izlemek adına da “check_all” parametreleri “yes” olarak ayarlanır.

```
GNU nano 7.2 /var/ossec/etc/ossec.conf *
</sca>

<!-- File integrity monitoring -->
<syscheck>
  <disabled>no</disabled>

  <!-- Frequency that syscheck is executed default every 12 hours -->
  <frequency>432</frequency>
  <scan_on_start>yes</scan_on_start>

  <!-- Directories to check (perform all possible verifications) -->
  <directories check_all="yes" whodata="yes">/etc,/usr/bin,/usr/sbin</directories>
  <directories check_all="yes" whodata="yes">/bin,/sbin,/boot</directories>
```

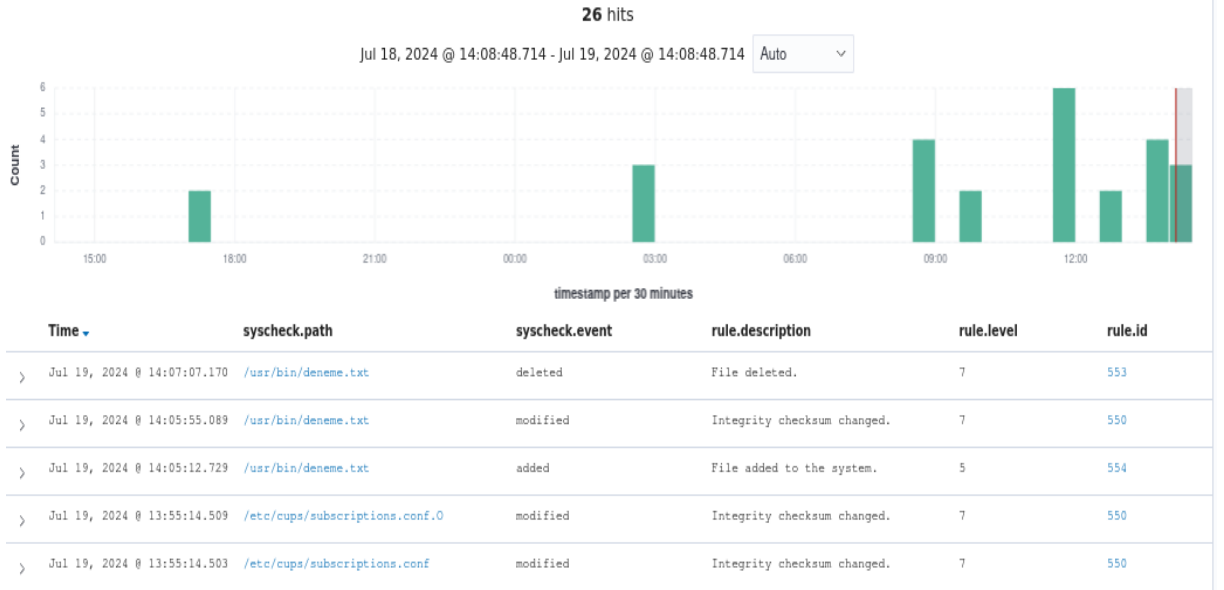
Şekil 6

- 2- Terminal üzerinde basit dosya işlemleri(oluşturma, düzeltme ve silme) işlemleri yapıldı.

```
root@debian:/bin# touch deneme.txt
root@debian:/bin# nano deneme.txt
root@debian:/bin# nano deneme.txt
root@debian:/bin# rm deneme.txt
root@debian:/bin# ls
'['          gcov-tool-12      mountpoint      spd-say
7z           gcr-viewer       mpri-proxy     spdsend
7za          gdbus            mt              speaker-test
7zr          gdialog          mt-gnu         speech-dispatcher
aa-enabled   gdk-pixbuf-csource mtrace         splain
```

Şekil 7

- 3- İlgili agentın “FIM->Events” kısmına gidilerek dashboard üzerinde yapılan değişiklikler aşağıdaki gibi gözlemlendi. Wazuh FIM modülü anlık izleme, zamanlayıcı ve işlem önceliği vs. gibi özellikleri kullanılarak kurumun kullanımına bağlı olarak custom edilebilir.



Şekil 8

Active Response

Siber güvenlik analistleri genellikle olay yanıtı sırasında yüksek öncelik olayını incelemek veya eylemleri azaltma da problemler yaşayabilirler. Bu durum olayla ilgili geniş bir açıdan bakma konusunda zorluk çıkarabilir. Bu problemler de siber atağın hafifletilmesini zorlaştırır.

Belirli tetiklenmelere dayalı Active Response siber olayları yönetme konusunda güvenlik uzmanlarına kolaylık sağlar. Response olaylarının otomatize edilmesi yüksek öncelikli olayların tutarlı ve anlık olarak değerlendirilmesini sağlar. Bu, güvenlik uzmanlarının öncelik işlemi için ekstra efor sarf etmesini engellemeye çalışır.

Ayrıca Wazuh Active Response modülü tehditleri aktörlerini hafifletmeye yönelik olarak “[out-of-the-box](#)” response scripti kullanır. Kötü niyetli olarak görünen ağ erişimini engellemeye çalışır. Bu da güvenlik uzmanlarının iş yükünü azaltmaya yarayan diğer özelliktir. Active Response bu scriptleri izlenen endpoint üzerinde çalıştırır. Bu kısımda dikkatli olunması önemlidir zayıf uygulama sistemde zafiyet ortaya çıkarabilir.

Yapılandırılma-Uygulama

Active Response özelliğinin konfigürasyonları aşağıdaki adımlarla sağlanabilir.

- 1- Wazuh sunucusunda “/var/ossec/etc/ossec.conf” dosyasında “<command>” bloğunun olup olmadığı kontrol edilir, yoksa eklenir. Burada hazır olarak geldiği görülmektedir. Burada “name” bloğu komutun adını, “executable” bloğu active response scriptini ve “timeout_allowed” bloğu belirli bir süre sonra zaman aşımına izin verir.

```
GNU nano 2.9.8 /var/ossec/etc/ossec.conf Modified
<timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>restart-wazuh</name>
  <executable>restart-wazuh</executable>
</command>

<command>
  <name>firewall-drop</name>
  <executable>firewall-drop</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>host-deny</name>
  <executable>host-deny</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>
```

Şekil 9

- 2- Daha sonra Wazuh sunucusunda “/var/ossec/etc/ossec.conf” dosyasında yorum satırı halinde olan “<active-response>” bloğu doldurulur. Dosyada “<command>” bloğu ayarlanacak komutu(önceki adımda belirtildi.), “<location>” bloğu “local” parametresi için izlenen uç cihazı, “server” parametresi wazuh sunucusunu, “defined_agent” parametresi de ID numarasına göre izlenecek agentı gösterir. “level” parametresi ise tepkinin ne kadar ciddi ve önemli olduğunu, “rules_id” parametresi hangi scriptin çalıştırılacağını belirtir(Burada SSH brute force için 5763 kullanıldı). İşlemlerin ardından wazuh manager yeniden başlatılır.

```
GNU nano 2.9.8 /var/ossec/etc/ossec.conf Modified

<active-response>
  <command>firewall-drop</command>
  <location>defined-agent</location>
  <rules_id>5763</rules_id>
  <agent_id>003</agent_id>
  <level>10</level>
  <timeout>200</timeout>
</active-response>
```

Şekil 10

- 3- Yapılan Brute Force atağından sonra hedef tarafından engellendiği ve ping atılamadığı gözlemlendi.

```
(root@kali)~# hydra -t 4 -l debb -P /usr/share/wordlists/rockyou.txt 10.0.2.19 ssh -vvv
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (c) 2013-2023, see https://www.thc-ipv6.org/about/convicted.html
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-02 03:09:05
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking ssh://10.0.2.19:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://debb@10.0.2.19:22
[INFO] Successful, password authentication is supported by ssh://10.0.2.19:22
[STATUS] 42.00 tries/min, 42 tries in 00:01h, 14344357 to do in 5692:13h, 4 active
[STATUS] 28.00 tries/min, 84 tries in 00:03h, 14344315 to do in 8538:17h, 4 active
[ERROR] Received signal 2, going down ...
The session file ./hydra.restore was written. Type "hydra -R" to resume session.

(root@kali)~# ^C

(root@kali)~# ping 10.0.2.19
PING 10.0.2.19 (10.0.2.19) 56(84) bytes of data.
^C
--- 10.0.2.19 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3049ms
```

Şekil 11

Security Configuration Assessment

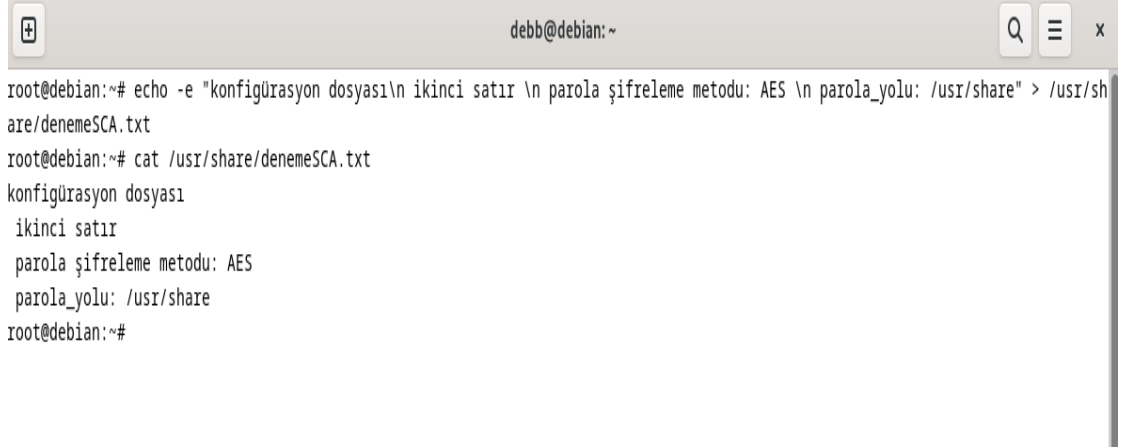
SCA(Security Configuration Assessment), bir sistemin uygulamalarının ve yapılandırılma ayarlarının kullanımına ait önceden tanımlanmış kuralları izleme sürecidir. Uç cihazları güvenli halde tutmanın en iyi yollarından bir tanesi de bu cihazın zafiyet yüzeyini azaltmaktır. SCA, efektif bir şekilde uç cihazdaki zayıflığı tespit etmeyi ve atak yüzeyini azaltmaya yarar. Wazuh SCA modülü tarama yaparak yanlış yapılandırılma ve sömürü gibi durumları inceleyerek bunlar hakkında iyileştirme çözümleri sunar.

SCA modülünün çalışma ilkesi yine bir veri tabanı işlemine dayanır. Her Wazuh agentı kendine ait SCA durumlarını kontrol ettiği SCA veri tabanı tutar. SCA modülü herhangi bir değişiklik veya kural üzerinde ihlal olması durumunda Wazuh sunucusuna bu değişiklik hakkında alert verir. Böylece gereksiz bir ağ trafiğinin de önüne geçilmiş olur.

SCA tarama sonuçları “Failed”, “Not Applicable” ve “Passed” olmak üzere 3 farklı olası durum vardır. Failed, belirli bir yapılandırma ayarının ilgili kurala uymadığını belirtir. Not Applicable, belirli bir güvenlik kuralının veya yapılandırma ayarının, mevcut sistem veya uygulama için geçerli olmadığını belirtir. Passed, tarama sırasında belirli bir güvenlik konfigürasyonunun veya uygulama ayarının, tüm önceden tanımlanmış güvenlik standartlarına veya kurallarına uygun olduğunu ifade eder

Yapılandırılma-Uygulama

- 1- İlgili dizin, "/usr/share" içinde test dosyası yaratıldı ve içeriği dolduruldu.



```
debb@debian: ~  
root@debian:~# echo -e "konfigürasyon dosyası\n ikinci satır \n parola şifreleme metodu: AES \n parola_yolu: /usr/share" > /usr/share/denemeSCA.txt  
root@debian:~# cat /usr/share/denemeSCA.txt  
konfigürasyon dosyası  
ikinci satır  
parola şifreleme metodu: AES  
parola_yolu: /usr/share  
root@debian:~#
```

Şekil 12

- 2- SCA kural dosyasına , "/var/ossec/etc/SCAparola-Folder/kelime.yml", ilgili kurallar yazıldı. Bu işlemten önce "SCAparola-Folder" dizini oluşturuldu. Check ID 10000, "/usr/share/denemeSCA.txt" dosyasını tarar. Herhangi bir "parola_yolu" ifadesi bulursa kontrolün başarısız olduğunu vurgular.



```
debb@debian: ~  
GNU nano 7.2 /var/ossec/etc/SCAparola-Folder/kelime.yml *  
policy:  
  id: password_check  
  file: kelime.yml  
  name: "SCA parola işlemi"  
  requirements:  
    condition: any  
    rules:  
      - 'f:/usr/share/denemeSCA.txt'  
  checks:  
    - id: 10000  
      description: "parolanın bulunduğu yol belirtildi"  
      rationale: "parolanın bulunduğu yolll bir dosya içinde belirtmeli mi değil mi"  
      remediation: "parolanın bulunduğu yol herhangi bir dosya içerisinde tutulamaz bu dosya içeriğinin düzeltilmesi gerekir"  
      condition: none  
      rules:  
        - 'f:/usr/share/denemeSCA.txt -> r:^parola_yolu: /usr/share$'  
Değiştirilen tamponu kaydet?  
E Evet  
H Hayır  ^C iptal
```

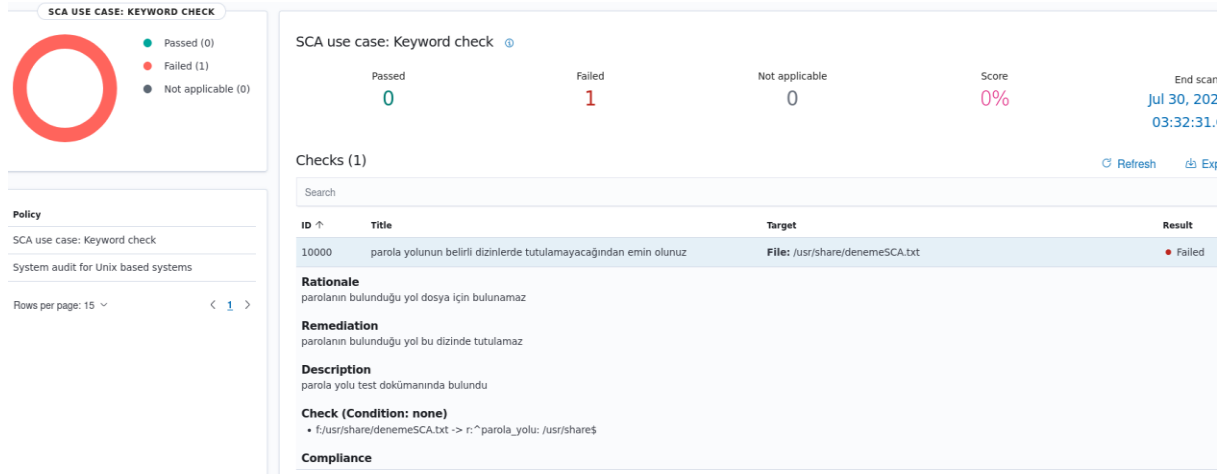
Şekil 13

- 3- Bu adımda Wazuh agentı üzerinde "/var/ossec/etc/ossec.conf" dosyası üzerinden kural tanımlaması ve aktifleştirilmesi yapılır.

```
debb@debian: ~  
GNU nano 7.2 /var/ossec/etc/ossec.conf *  
<packages>yes</packages>  
<ports all="no">yes</ports>  
<processes>yes</processes>  
  
<!-- Database synchronization settings -->  
<synchronization>  
  <max_eps>10</max_eps>  
</synchronization>  
</wodle>  
  
<sca>  
<policies>  
<policy enabled="yes">/var/ossec/etc/SCAparola-Folder/kelime.yml</policy>  
</policies>  
</sca>
```

Şekil 14

- 4- Wazuh agentı “systemctl restart wazuh-agent” komutuyla yeniden başlatılır. Ardından ara yüz üzerinden uygulamanın son çıktısı kontrol edilir.



Şekil 15

Vulnerability detection

Güvenlik açıkları, bilgisayar sistemlerinde tehdit aktörlerinin yetkisiz erişim sağlamak için kullanabileceği zayıflıklardır. Kötü amaçlı yazılımlar ve siber saldırganlar, bu açıkları kullanarak uzaktan kod çalıştırabilir, verileri sızdırabilir ve çeşitli kötü niyetli faaliyetlerde bulunabilirler. Bu nedenle, organizasyonların, kötü niyetli aktörlerin bu açıkları kullanmadan önce ağlarındaki güvenlik açıklarını hızla tespit edebilecek stratejilere ve güvenlik çözümlerine sahip olmaları kritiktir.

Wazuh Güvenlik Açığı Tespit modülü, kullanıcıların işletim sistemleri ve izlenen uç noktalardaki kurulu uygulamalarda mevcut güvenlik açıklarını tespit etmelerine yardımcı olur. Bu modül, çeşitli güvenlik açığı kaynaklarını kullanarak sistemlerdeki potansiyel zayıflıkları belirler ve kullanıcıları bilgilendirir.

Yapılandırılma-Uygulama

- 1- Zafiyet tespiti için Wazuh agentı uç cihazlardan ve onlara yüklenen yazılımların listesinin periyodik olarak wazuh sunucusuna iletir. Wazuh sunucusundaki yerel SQLite veri tabanı bu bilgileri tutar. Wazuh sunucusundaki “Vulnerability Detection” modülü , yazılım envanterinde bulunan verileri [CVE](#) dokümanı ile ilişkilendirir. Bu dizindeki “/var/ossec/etc/ossec.conf” dosyası varsayılan olarak aşağıdaki gibidir.

```
Dosya Makine Görünüm Giriş Aygıtlar Yardım
GNU nano 2.9.8 /var/ossec/etc/ossec.conf

<vulnerability-detection>
  <enabled>yes</enabled>
  <index-status>yes</index-status>
  <feed-update-interval>60m</feed-update-interval>
</vulnerability-detection>

<indexer>
  <enabled>yes</enabled>
  <hosts>
    <host>https://127.0.0.1:9200</host>
  </hosts>
  <ssl>
```

Şekil 16

- 2- Dashboard alanında uç cihaz aşağıdaki gibi gözlenir. Burada zafiyetin derecesi ve CVE kodu ile detaylı incelemeler yapılabilir.

1,125 hits						
agent.name	package.name	package.version	vulnerability.description	vulnerability.severity	vulnerability.id	
debb	avahi-daemon	0.8-10	A vulnerability was found in Avahi. A re...	Medium	CVE-2023-38473	
debb	avahi-daemon	0.8-10	A vulnerability was found in Avahi. A re...	Medium	CVE-2023-38472	
debb	avahi-daemon	0.8-10	A vulnerability was found in Avahi, whe...	Medium	CVE-2023-38469	
debb	avahi-daemon	0.8-10	A vulnerability was found in Avahi. A re...	Medium	CVE-2023-38471	
debb	avahi-daemon	0.8-10	A vulnerability was found in Avahi. A re...	Medium	CVE-2023-38470	
debb	libgnutls30	3.7.9-2+deb12u3	The SSL protocol, as used in certain co...	Medium	CVE-2011-3389	
debb	libbikid1	2.38.1-5+deb12u1	A flaw was found in the util-linux chfn a...	Low	CVE-2022-0563	
debb	gststreamer1.0-packagekit	1.2.6-5	A use-after-free flaw was found in Pack...	Low	CVE-2024-0217	
debb	gststreamer1.0-packagekit	1.2.6-5	A flaw was found in PackageKit in the w...	Low	CVE-2022-0987	
debb	python3.11-minimal	3.11.2-6+deb12u2	The email module of Python through 3....	Medium	CVE-2023-27043	

Şekil 17

Yukarıda, “/var/ossec/etc/ossec.conf”, dosyasında “<vulnerability-detection>” bölümü, zafiyet taramasını etkinleştirir ve zafiyet verilerinin düzenli olarak güncellenmesini sağlar. “<indexer>” bölümü, verilerin [Elasticsearch](#) veya benzeri bir veri deposuna indekslenmesini sağlar ve bu işlemi [SSL/TLS](#) ile güvence altına alır. Bu yapılandırma ayarları, Wazuh'un zafiyet tespiti ve veri indeksleme işlemlerinin düzgün bir şekilde çalışmasını sağlamak için kullanılır.

Log Data Collection

Wazuh log modülü uç cihazlardan, uygulamalardan ve ağ cihazlarından logları toplar. Gerçek zamanlı olarak toplanmış bu logları analiz eder ve uygun alanlara çıkarır. Wazuh analiz modeli belirli kurallara göre logları değerlendirir ve tüm uyarıları “/var/ossec/logs/alerts/alerts.log” ve “/var/ossec/logs/alerts/alerts.json” gibi dosyalara kaydeder.

Syslog, sistemlerin ve ağ cihazlarının çeşitli olayları ve log kayıtlarını merkezi bir sunucuya iletmek için kullanılan bir protokoldür. Bu protokol, hem Unix benzeri sistemlerde hem de diğer işletim sistemlerinde geniş bir uygulama yelpazesi sunar. Syslog, genellikle sistem performansını izlemek, hataları tespit etmek ve güvenlik olaylarını analiz etmek amacıyla kullanılır. Wazuh sunucusu uç cihazlar üzerindeki syslog loglarını toplayabilir. Syslog yapılandırılması aşağıda belirtilmiştir.

İlgili dosya, “/var/ossec/etc/ossec.conf”, dosyası açılır ve bir “remote” bloğu daha eklenerek aşağıdaki gibi doldurulur ve Wazuh yöneticisi yeniden başlatılır. Burada port numarası syslog port numarasını, “allowed-ips” ağ içindeki uç cihazların IP adresini ve “local_ip” Wazuh sunucu IP adres bilgisini belirtir.

```
<remote>
  <connection>secure</connection>
  <port>1514</port>
  <protocol>tcp,udp</protocol>
  <queue_size>16384</queue_size>
  <rids_closing_time>600</rids_closing_time>
  <connection_overtake_time>600</connection_overtake_time>
</remote>

<remote>
  <connection>syslog</connection>
  <port>514</port>
  <protocol>tcp</protocol>
  <allowed-ips>10.0.2.0/24</allowed-ips>
  <local_ip>10.0.2.20</local_ip>
</remote>
<!-- Policy monitoring -->

[root@wazuh-server ~]# systemctl restart wazuh-manager.service
[root@wazuh-server ~]#
```

Şekil 18

Windows işletim sistemine yüklenen uygulamalar da Wazuh tarafınan izlenebilir. Bu yüklemeler ara yüz ekranından takip edilerek detaylı incelemeler de yapılabilir.

Yapılandırılma-Uygulama

- 1- Wazuh sunucusunda “/var/ossec/etc/ossec.conf” üzerindeki ayarlar varsayılan olarak gelir. Diğer yapılandırma ise “/var/ossec/ruleset/rules/0585-win-application_rules.xml” dosyası üzerinden gerçekleşir. İlgili uygulama için, Wazuh ile Windows üzerine kurulan uygulamaların izlenmesi, XML dosyası aşağıdaki gibi hazır haldedir. Uygulama kurulumunda oluşacak alert 60612 numaralı alerttir.

```
/var/ossec/ruleset/rules/0585-win-application_rules.xml

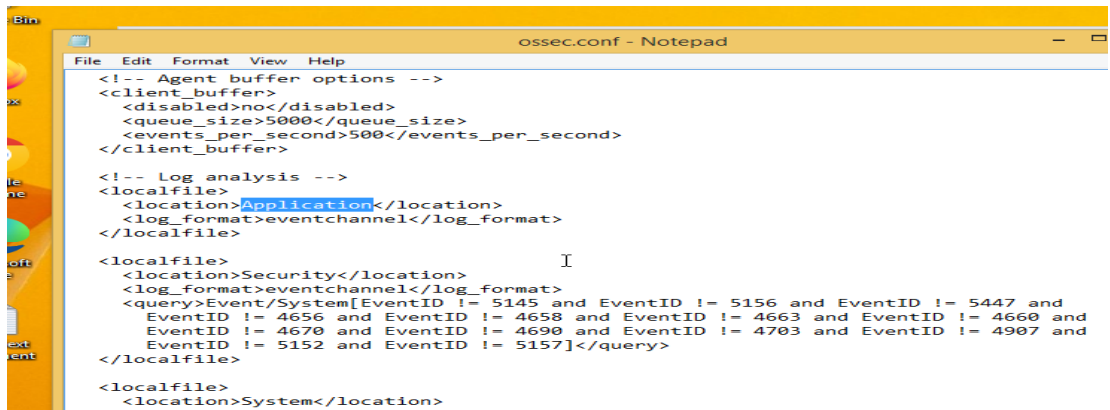
<description>Windows installer began an installation process.</description>
</rule>

<rule id="60611" level="3">
  <if_sid>60609</if_sid>
  <field name="win.system.eventID">^11724$|^1034$</field>
  <options>no_full_log</options>
  <description>Application uninstalled $(win.eventdata.data).</description>
</rule>

<rule id="60612" level="3">
  <if_sid>60609</if_sid>
  <field name="win.system.eventID">^11707$|^1033$</field>
  <options>no_full_log</options>
  <description>Application installed $(win.eventdata.data).</description>
</rule>
```

Şekil 19

- 2- Windows üzerindeki “C:\Program Files (x86)\ossec-agent\ossec.conf” dosyasındaki “<location>” bloğunun şekilde yapılandırılmış olduğu kontrol edildi. Bu Wazuh sunucusundaki XML dosyasıyla eş zamanlı olarak işlem yapmaktadır.



```
File Edit Format View Help
<!-- Agent buffer options -->
<client_buffer>
  <disabled>no</disabled>
  <queue_size>5000</queue_size>
  <events_per_second>500</events_per_second>
</client_buffer>

<!-- Log analysis -->
<localfile>
  <location>Application</location>
  <log_format>eventchannel</log_format>
</localfile>

<localfile>
  <location>Security</location>
  <log_format>eventchannel</log_format>
  <query>Event/System[EventID != 5145 and EventID != 5156 and EventID != 5447 and
    EventID != 4656 and EventID != 4658 and EventID != 4663 and EventID != 4660 and
    EventID != 4670 and EventID != 4690 and EventID != 4703 and EventID != 4907 and
    EventID != 5152 and EventID != 5157]</query>
</localfile>

<localfile>
  <location>System</location>
```

Şekil 14

- 3- Notepad++ adlı uygulamanın yüklendiği bilgisi Wazuh ara yüzünde aşağıdaki gibi gözlemlendi. Rule ID değeri de işlemin doğru olduğunun bir diğer göstergesi oldu. İlgili ID değerine gidilerek de detaylı bilgiler toplanabilir.



Şekil 21

Malware Detection

Malware Detection modülü zararlı yazılım veya dosyanın tespiti için kullanılmaktadır. Güvenlik araçları bilinen [Malware](#) imzasına bakarak zararlı yazılım-dosyayı tespit edebilir. Ancak bazen zararlı yazılım ve dosyalar sisteme enjekte olmuş olabilir. Wazuh geniş bir çerçeveye [Malware](#) varlığını ve anormal olayları ortaya çıkarabilir.

Wazuh FIM modülü izlenen uç cihazlar üzerindeki zararlı dosyaları tespit edebilir. Ancak tek başına FIM modülü yeterli olmayabilir. Bu eksiğin ortadan kaldırılması adına FIM modülünün threat tespit kurallarıyla kombine edilmesi gereklidir. FIM modülü [VirusTotal](#) gibi tehdit istihbaratı kaynağı kullanılarak daha etkili hale getirilebilir.

Wazuh FIM modülü dosyaları hash formatında tutar ve herhangi bir değişiklik, silme ve oluşturma gibi işlemlerde uyarı verir. Virustotal API entegrasyonu gerçekleştirildiğinde bu hash dışarı çıkarılır. Entegrasyon Virustotal [API](#) özelliğini kullanarak [HTTP POST](#) isteği gönderir. Bu çağrı Virustotal veri tabanı ile karşılaştırılmış dosya hashlerini karşılaştırmak için kullanılır. Entegrasyon istek sonucu olan JSON yanıtı alır. Bu yanıtlar “/var/ossec/logs/integrations.log” dosyasında tutulur. Virustotal API entegrasyonu ilgili uygulamada örneklendirilmiştir.

1- Wazuh sunucusunda API anahtarı “ /var/ossec/etc/ossec.conf” dosyasına Şekil 20 üzerinde olduğu gibi “integration” bloklarına kaydedilir.

Şekil 22

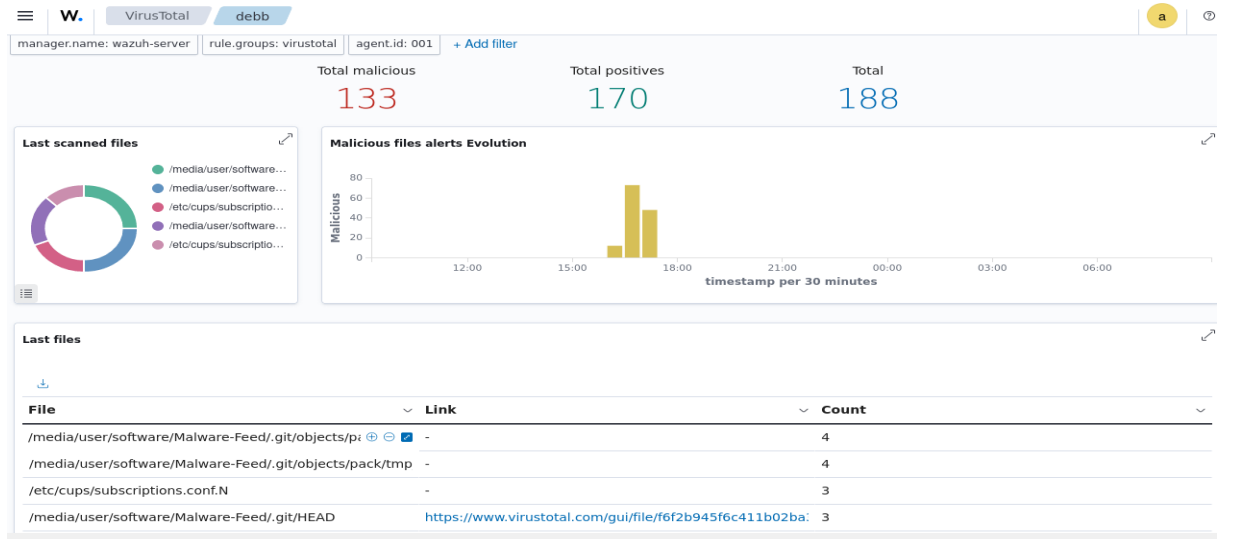
- Şekil 23

- 3- Malware dosyaları ilgili klasöre indirildi.

```
root@debian: /media/user/software# ls
Malware-Feed
root@debian: /media/user/software# ls Malware-Feed/
2020.06.22_FBI-FLASH-MI-000124-MW
2020.07.16_CISA-WELLMAIL
2020.07.23_FBI-FLASH-AC-000129-TT
2020.07.27_CISA-Legacy_Malware_Targeting_QNAP_NAS
2020.07.28_FBI-FLASH-MI-000130-MW
2020.08.03_CISA-Chinese_RAT_TAIDOO
2020.08.19_CISA-North_Korean_RAT_BLINDINGCAN
2020.08.26_CISA-MAR-10301706_North_Korean_RAT_VIVACIOUSGIFT
2020.08.26_CISA-North_Korean_RAT_ECCENTRICBANDWAGON
2020.08.26_CISA-North_Korean_RAT_FASTCASH
2020.09.15_CISA-MAR-10297887_Iranian_Web_Shells
2020.09.17_FBI-FLASH-ME-000134-MW
2020.09.18_Checkpoint-Rampant_Kitten
2020.09.29_Symantec-Palmerworm Espionage Gang
```

Şekil 24

- 4- Sayfa yenilemesinin ardından uyarılar panelde oluşmaya başladı.



Şekil 15

Troubleshootings

Wazuh özelliklerinin incelenmesinde karşılaşılan problemler ve hata gidermeleri aşağıda belirtilmiştir.

Hata: “Wazuh services cannot be started”.

Çözüm: Log Dosyalarını Kontrol Etme ve Wazuh servislerini yeniden başlatma

Wazuh log dosyalarını kontrol ederek hataların ne olduğunu belirleyin. Log dosyaları genellikle “/var/ossec/logs/” dizininde bulunur.

“sudo tail -f /var/ossec/logs/ossec.log” komutu ile en son logları görüntüleyebilirsiniz.

“sudo systemctl restart wazuh-manager”, “sudo systemctl restart wazuh-agent”

Hata: “See systemctl status wazuh-agent.service” and “journalctl -xeu wazuh-agent.service” for details.”

Çözüm: Özellikle “ossec.conf” dosyası üzerinde yapılan hatalardan dolayı bu sorunla karşılaşıldı. Dolayısıyla bu dosyanın yapılandırılma işleminin kontrol edilmesi gerekir. Ayrıca “sudo journalctl -xeu wazuh-agent.service” komutuyla “wazuh-agent” ile ilgili hatalar da bu şekilde gözlemlenebilir.

Sonuç

Wazuh, ağ güvenliği ve olay yönetimi için kapsamlı bir çözüm sunar. File Integrity Monitoring (FIM) modülü, dosya değişikliklerini izleyerek anormal aktiviteleri raporlar. Active Response, otomatik olay yanıtları sağlayarak hızlı tepki mekanizmaları oluşturur. Security Configuration Assessment (SCA) modülü, sistem yapılandırmalarını değerlendirip zayıflıkları tespit eder. Vulnerability Detection ise mevcut güvenlik açıklarını tarar ve CVE veritabanıyla ilişkilendirir. Log Data Collection ve Malware Detection özellikleri, log verilerini toplar ve zararlı yazılımları tespit etmek için entegre tehdit istihbaratı kullanır. Bu özellikler, sistemin güvenliğini artırarak olası tehditleri etkili bir şekilde yönetir.

REFERANSLAR

- [1] (Stefan Stanković, A Review of Wazuh Tool Capabilities for Detecting Attacks Based on Log Analysis, 2022) https://www.etrans.rs/2022/zbornik/ICETRAN-22_radovi/068-RTI2.6.pdf
- [2] (Moiz, Majid, Basit, Ebrahim, & Abro, 2024) Moiz, S., Majid, A., Basit, A., Ebrahim, M., & Abro, A. A. (2024, Nisan 02). Security and Threat Detection through Cloud-Based Wazuh Deployment. <https://ieeexplore.ieee.org/abstract/document/10482206>
- [3] Wazuh dokümantasyon bileşenleri, <https://documentation.wazuh.com/current/gettingstarted/components/index.html>
- [4] Wazuh dokümantasyon genel bakış, <https://documentation.wazuh.com/current/>
- [5] Wazuh özellik incelemesi, <https://www.youtube.com/watch?v=i68atPbB8uQ>
- [6] Hata gidermeleri, <https://github.com/wazuh/wazuh/issues/22431>
- [7] Hata gidermeleri, https://www.reddit.com/r/Wazuh/comments/1dw23zp/edit_ossecconf_on_dashboa rd_is_not_working/
- [8] Hata gidermeleri, <https://stackoverflow.com/questions/70983686/wazuh-filebeat-elasticsearch-non-zero-metrics>