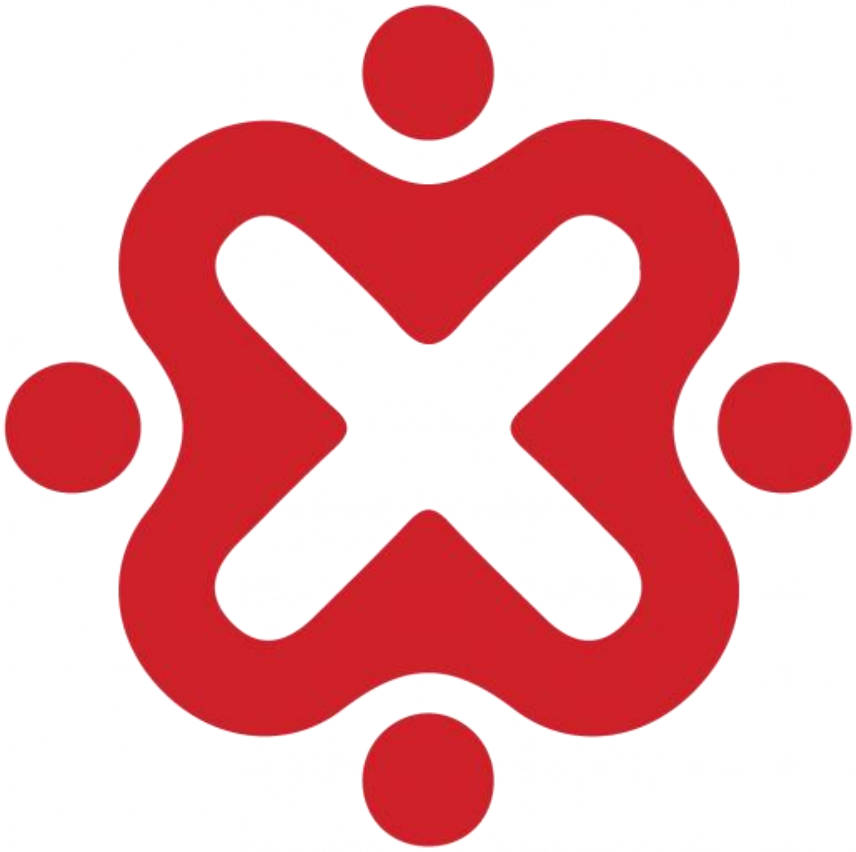
Instalasi Wazuh Single-Node Dan Penerapan Decoder Dan Ruleset



ID-Networkers

Indonesian IT Expert Factory

1. **Explanation**
   - **Wazuh**

     Wazuh adalah platform keamanan open-source yang berfungsi sebagai SIEM (Security Information and Event Management) sekaligus HIDS (Host-based Intrusion Detection System). Wazuh digunakan untuk mengumpulkan, memantau, menganalisis, dan merespons data keamanan dari berbagai sistem seperti server, endpoint, maupun layanan cloud. Dengan memasang agent di setiap host, Wazuh memantau integritas file, menganalisis log, mendeteksi intrusi, menemukan kerentanan, serta membantu organisasi memenuhi standar kepatuhan seperti PCI DSS atau ISO 27001. Semua data dianalisis oleh manager dan biasanya divisualisasikan melalui integrasi dengan Elastic Stack (ELK) sehingga memudahkan tim keamanan mendeteksi aktivitas mencurigakan dan meresponsnya secara otomatis. Karena bersifat gratis, fleksibel, dan mendukung banyak sistem operasi, Wazuh menjadi salah satu tools populer di dunia cybersecurity, khususnya bagi tim blue team dan Security Operation Center (SOC) untuk memperkuat sistem deteksi dan respons insiden.

   - **Wazuh Decoder**

     Wazuh Decoder adalah bagian dari mekanisme analisis log di Wazuh yang berfungsi untuk membaca, mengenali, dan menguraikan format data log dari berbagai sumber. Decoder membantu Wazuh memahami isi log mentah yang dikirim oleh agent, kemudian mengubahnya menjadi informasi terstruktur agar bisa diproses lebih lanjut oleh ruleset. Setiap decoder memiliki pola (pattern) tertentu yang menyesuaikan format log, seperti log Apache, SSH, firewall, atau aplikasi custom. Dengan decoder, Wazuh dapat menangkap detail penting misalnya IP penyerang, status login, atau pesan error—lalu mencocokkannya dengan aturan deteksi. Decoder juga bisa dibuat atau disesuaikan sendiri jika organisasi menggunakan aplikasi dengan format log khusus. Intinya, decoder adalah jembatan antara data log mentah dan sistem deteksi Wazuh, sehingga memastikan setiap log dianalisis dengan benar dan akurat.

   - **Wazuh Ruleset**

     Wazuh Ruleset adalah kumpulan aturan deteksi yang digunakan untuk menganalisis log yang sudah diuraikan oleh decoder. Ruleset berfungsi sebagai filter pintar yang menentukan apakah data log berisi aktivitas normal, peringatan, atau indikasi serangan. Setiap aturan mendefinisikan pola tertentu seperti upaya login gagal berulang, modifikasi file sistem, atau koneksi dari IP mencurigakan dan jika pola tersebut cocok dengan data log, maka Wazuh akan menghasilkan alert. Ruleset dapat diatur berdasarkan prioritas (severity), sehingga tim keamanan bisa memutuskan tindakan yang tepat. Selain itu, ruleset di Wazuh bisa dikustomisasi atau dibuat sendiri agar sesuai dengan kebutuhan dan kebijakan keamanan organisasi. Dengan ruleset inilah Wazuh menjadi efektif dalam mendeteksi intrusi, penyalahgunaan sistem, maupun insiden lain secara otomatis dan real-time.

2. **Wazuh Installation**
   Saya menggunakan Ubuntu Server 22.04.05 LTS sebagai basenya dan langkah-langkah pastinya bisa kalian lihat di link ini :
   https://documentation.wazuh.com/current/quickstart.html

3. **Log Example**
   Saya diberikan 2 file log yaitu openstack_abnormal.log dan openstack_normal1.log
   Isi dari log tersebut adalah rangkumman dari serangkaian aktivitas yang diproses di suatu platform dalam scope tertentu

4. **Buat Decoder**
   Ini decoder yang saya buat

```
<decoder name="nova-api">
    <prematch type="pcre2">^nova</prematch>
    <regex type="pcre2">^(nova-api\.log\.\d{4}-\d{2}-\d{2}_\d{2}:\d{2}:\d{2})</regex>
    <order>header</order>
</decoder>

<decoder name="nova-compute">
    <prematch type="pcre2">^nova</prematch>
    <regex type="pcre2">^(nova-compute\.log\.\d{4}-\d{2}-\d{2}_\d{2}:\d{2}:\d{2})</regex>
    <order>header</order>
</decoder>

<!-- Anaknya -->

<decoder name="level-api">
    <parent>nova-api</parent>
    <regex type="pcre2">(INFO|DEBUG|ERROR|WARNING|CRITICAL)</regex>
    <order>level</order>
</decoder>

<decoder name="level-compute">
    <parent>nova-compute</parent>
    <regex type="pcre2">(INFO|DEBUG|ERROR|WARNING|CRITICAL)</regex>
    <order>level</order>
</decoder>
```

   Decoder akan membaca line dari log yang ada dan akan membaca 2 bagian yaitu headernya contohnya nova-compute.log.2017-05-14_21:27:09 2017-05-14 19:39:36.041
   Lalu bagian loglevel akan terbaca oleh child decodernya contohnya INFO, ERROR, CRITICAL dan lainnya

   File bisa dilihat di github saya di referensi

5. **Buat Ruleset**

Ruleset yang saya buat seperti ini

```
<group name="nova,">
  <rule id="100100" level="0">
    <decoded_as>nova-api</decoded_as>
    <description>Base Line nova-api</description>
  </rule>

  <rule id="100200" level="0">
    <decoded_as>nova-compute</decoded_as>
    <description>Base Line nova-compute</description>
  </rule>

<!-- INFO Masseh -->

  <rule id="100101" level="1">
    <if_sid>100100</if_sid>
    <field name="level">INFO</field>
    <description>INFO-Line</description>
  </rule>

  <rule id="100201" level="1">
    <if_sid>100200</if_sid>
    <field name="level">INFO</field>
    <description>INFO-Line</description>
  </rule>
```

```
<!-- Level Lain-->

  <rule id="100102" level="7">
    <if_sid>100100</if_sid>
    <field name="level">DEBUG|ERROR|WARNING|CRITICAL</field>
    <description>Bad Log Level Matched</description>
  </rule>

  <rule id="100202" level="7">
    <if_sid>100200</if_sid>
    <field name="level">DEBUG|ERROR|WARNING|CRITICAL</field>
    <description>Bad Log Level Matched</description>
  </rule>
</group>
```

Ruleset akan mengklasifikasikan log sesuai levelnya disini log INFO saya beri level 1, log DEBUG, ERROR, CRITICAL dan WARNING saya beri level 7 menurut referensi link ini

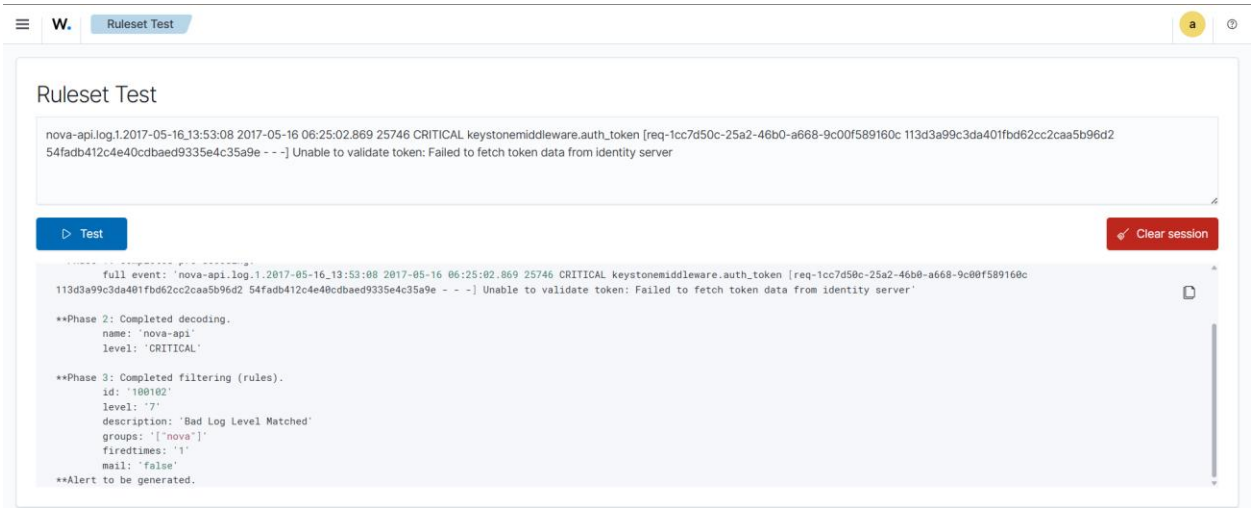
File ada di github saya juga, saya sertakan di referensi

6. **Uji Coba**

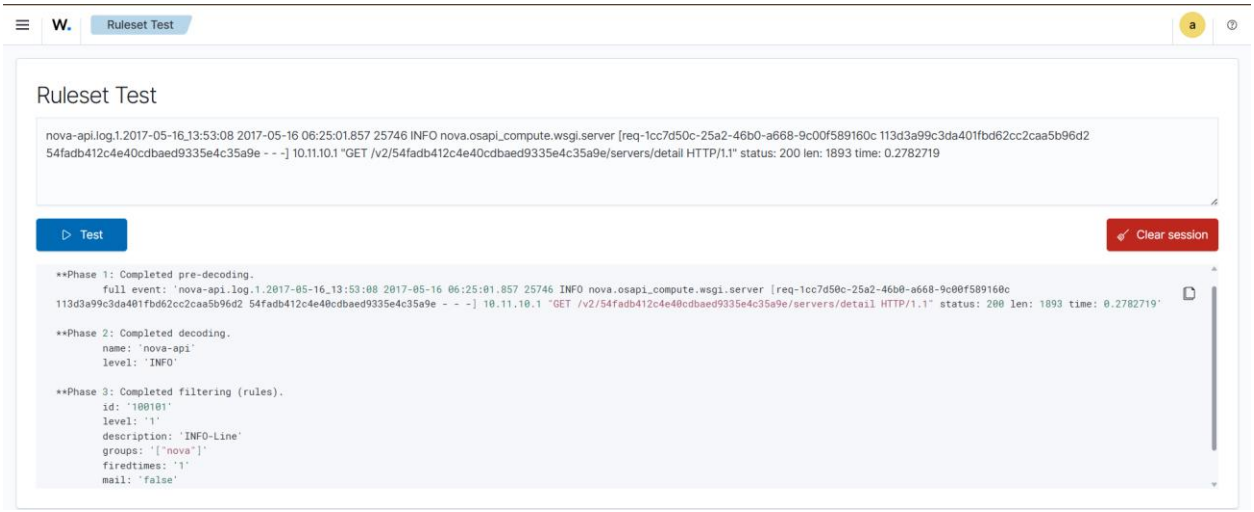
Decoder dan ruleset ini akan diterapkan di wazuh yang sudah saya instalasi dengan langkah seperti ini:

- Akses direktori /var/ossec/etc/decoders
- Ubah isi dari local_decoder.xml menjadi decoder buatan sendiri
- Akses lagi direktori /var/ossec/etc/rules
- Ubah isi dari local_rules.xml dengan rules yang sudah kita buat
- Restart wazuh-manager dengan **sudo systemctl restart wazuh-manager**

Setelah itu akses wazuh dashboard dan masuk ke bagian Server Management > Ruleset Test dan masukkan satu atau beberapa line dari file log openstack yang saya bahas tadi dan tunggu responnya



Saya mencoba memasukkan line log berlevel CRITICAL dan respon dari ruleset testnya menunjukan alert akan segera ter-generate. Artinya implementasi decoder dan ruleset diatas sudah bekerja



Saya mencoba memasukkan log berlevel INFO dan hasilnya menunjukan decoder dan ruleset sudah membaca dan mengklasifikasikan sesuai dengan konfigurasi

***Note**

Disini saya decoder dan ruleset yang saya buat hanya mengklasifikasikan berdasarkan loglevelnya saja, Log dengan level INFO pun bisa saja mengandung isi yang menunjukan indikasi bahaya. Kali ini saya hanya melakukan praktik awal untuk edukasi saja

7. Pengembangan

Development ruleset dan decoder ini bisa dikembangkan dengan

- Melengkapkan regex untuk membaca full line log yang diberikan
- Mengkategorikan tiap line log yang awalnya berbeda agar bisa membaca semua jenis kemungkinan lognya
- Dari point sebelumnya, bisa ditambahkan regex untuk membaca indikasi berbahaya per bagian dari lognya seperti HTTP Request, hasil message, dan bagian potensial lainnya
- Mengklasifikasikan ruleset level sesuai dengan standard yang ada

8. Referensi

https://documentation.wazuh.com/current/user-manual/ruleset/rules/rules-classification.html
https://github.com/newbieganas/wazuhtest
https://documentation.wazuh.com/current/user-manual/ruleset/ruleset-xml-syntax/index.html
https://documentation.wazuh.com/current/user-manual/ruleset/decoders/custom.html

Special Thanks To This Repo For My Learning Resource
https://github.com/ariafatah0711/idn_bootcamp/blob/main/task/week_9/2_fine_tunning_wazuh