BAB V
HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil sistem yang sudah dikembangkan dan dilakukan oleh peneliti, beserta dengan pembahasannya seperti berikut.

5.1 Hasil

Pengembangan SIEM berbasis Wazuh untuk mencegah akses pada URL berbahaya berhasil dikembangkan menggunakan dataset yang berisi URL-URL berbahaya dalam sebuah file yang berformat CSV. Dataset tersebut berisi parameter URL dan Type yang kemudian dilatih oleh model LLM untuk melakukan prediksi serta pencegahan jika URL terdeteksi berbahaya/*malicious*. Hasil yang dicapai setelah proses pelatihan model membuktikan bahwa URL yang bersifat berbahaya/*malicious* dapat dideteksi pada SIEM dan diblokir aksesnya.

Penelitian ini memanfaatkan Large Language Model berbasis BERT untuk memprediksi dan memblokir akses pada URL berbahaya. Umumnya, pemantauan dilakukan secara manual sehingga pemeriksaan akses pada URL juga belum bisa secara otomatis. Sehingga pengembangan pada penelitian ini bertujuan untuk memastikan pemblokiran pada URL berbahaya dilakukan secara otomatis. Berikut adalah hasil pelatihan dan pengujian yang telah dilakukan.

Gambar 5.1 Hasil pengujian Akurasi dan Confusion Matrix pada model LLM

Berdasarkan gambar diatas, hasil pengujian adalah sebagai berikut;

Tabel 5.1 Tabel hasil pengujian Confusion Matrix

Confusion Matrix	Predicted Benign	Predicted *Malicious*
True Benign	1287	16
True *Malicious*	11	686

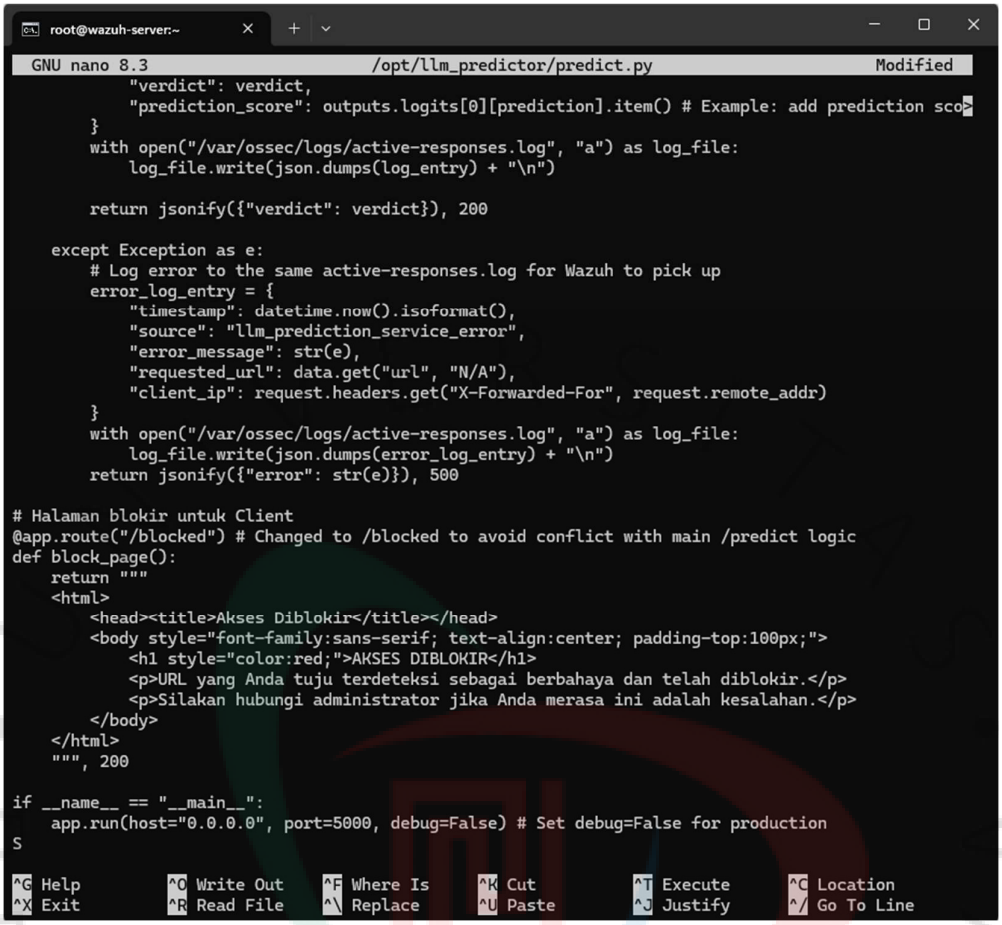
Hasil pengujian menunjukkan:

- **True Positive/True *Malicious***:
  686 URL berbahaya berhasil dideteksi sebagai berbahaya.
- **True Negative/True Benign:**
  1287 URL aman dan dideteksi sebagai aman.
- **False Positive/Predicted *Malicious***:
  16 URL aman tapi salah diklasifikasikan sebagai berbahaya.
- **False Negative/Predicted Benign**:
  11 URL berbahaya tapi lolos sebagai aman.

Tabel 5.2 Tabel hasil pengujian Evaluation Metrics

Evaluation Metrics	
Accuracy	0.9865 / 98%
Precision	0.9772 / 97%
Recall	0.9842 / 98%
F1-Score	0.9807 / 98%

Hasil pengujian menunjukkan hasil accuracy mencapai 98%; dari seluruh URL yang diuji, hasil klasifikasi antara URL berbahaya dan aman mencapai 98%. Hasil precision mencapai 97%; dari seluruh URL berbahaya yang diuji, 97% menunjukkan bahwa URL benar-benar berbahaya. Hasil recall mencapai 98%; 98% menunjukkan model berhasil memprediksi dan mendeteksi seluruh URL yang benar-benar berbahaya. Hasil F1-Score mencapai 98%; model berhasil menyeimbangkan antara metrik True Positive, True Negative, False Positive dan False Negative sebesar 98%.



Gambar 5.2 Konfigurasi predict.py

Gambar diatas adalah file predict.py yang berfungsi untuk melakukan prediksi dan menentukan sifat URL dari dataset yang telah dilatih. Jika user/pengguna terdeteksi mengakses URL berbahaya, maka akses akan dialihkan pada halaman blokir. Jika user/pengguna mengakses URL aman, maka akses akan dilanjutkan tanpa adanya halaman blokir

```
  GNU nano 8.3              /etc/systemd/system/llm_predict_service.service
[Unit]
Description=URL Prediction LLM Service (BERT)
After=network.target

[Service]
User=root
Group=root
WorkingDirectory=/opt/llm_predictor/
ExecStart=/usr/bin/python3 predict.py
Restart=always
StandardOutput=append:/var/log/llm_predict_service.log
StandardError=append:/var/log/llm_predict_service_error.log

[Install]
WantedBy=multi-user.target
```

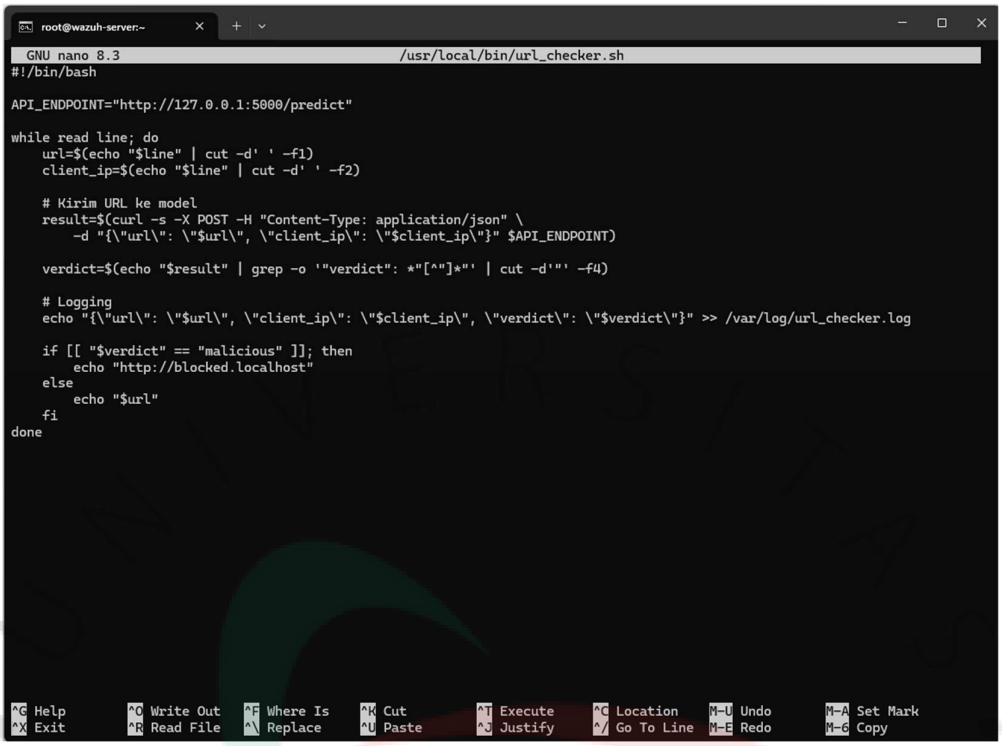
Gambar 5.3 Konfigurasi llm_predict_service.service

File llm_predict_service.service pada gambar diatas digunakan untuk mengaktivasi layanan/service yang menuju pada predict.py, sehingga fungsi untuk melakukan prediksi atau memeriksa sifat URL yang telah dilatih dapat dilakukan secara otomatis dengan perintah “curl”.

```
  GNU nano 8.3                          /etc/squid/squid.conf
# Port proxy
http_port 3128
visible_hostname squid-wazuh

# Format log untuk Wazuh (custom format)
logformat wazuh_format %{X-Forwarded-For}>h %>a %ui %un [%tl] "%rm %ru HTTP/%rv" %>Hs %<st
access_log /var/log/squid/access.log wazuh_format

# ACL jaringan internal (Kali Linux)
acl localnet src 192.168.100.0/24
acl localhost_access src 127.0.0.1/32

# ACL untuk host blokir (blocked.localhost)
acl block_page_host dstdomain blocked.localhost

# Jalankan URL rewriter
url_rewrite_program /usr/local/bin/url_checker.sh
url_rewrite_children 5 startup=1 idle=1 concurrency=0

# Aturan akses
http_access allow block_page_host
http_access allow localnet
http_access allow localhost_access
http_access deny all
```

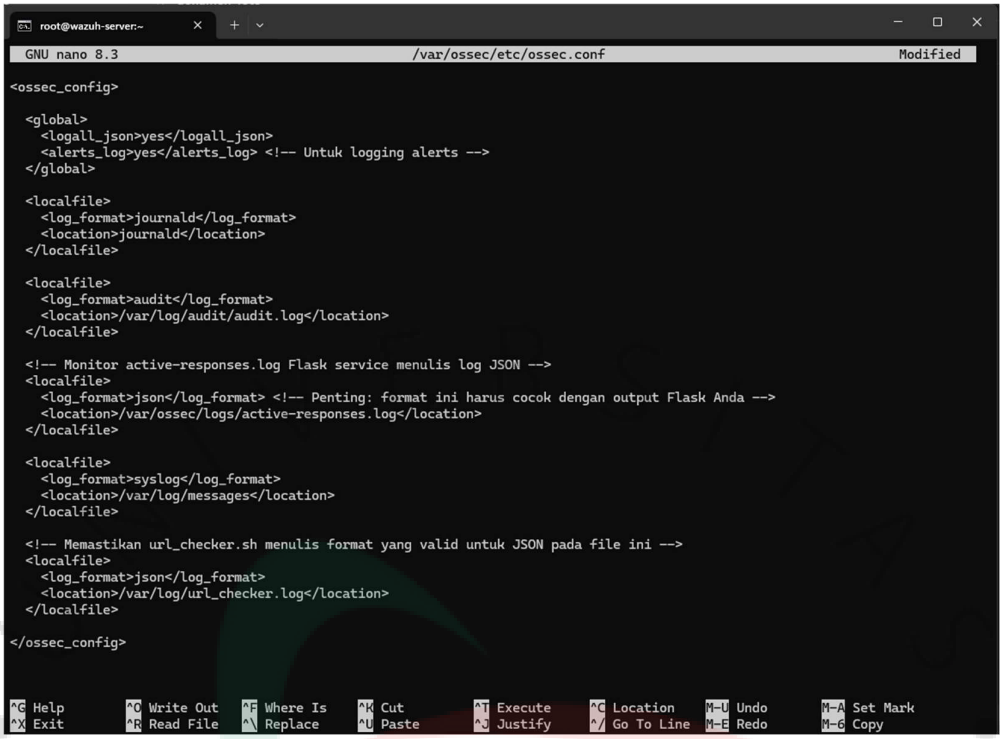
Gambar 5.4 Konfigurasi Squid Proxy

Gambar 5.4 merupakan konfigurasi Squid Proxy sebagai pemblokir jika ada terjadinya akses pada URL berbahaya. Squid Proxy juga berperan untuk melanjutkan log blokir agar dapat ditampilkan pada Wazuh, dan memanggil konfigurasi URL_checker.sh yang berfungsi untuk memeriksa hasil pemeriksaan URL melalui konfigurasi predict.py.



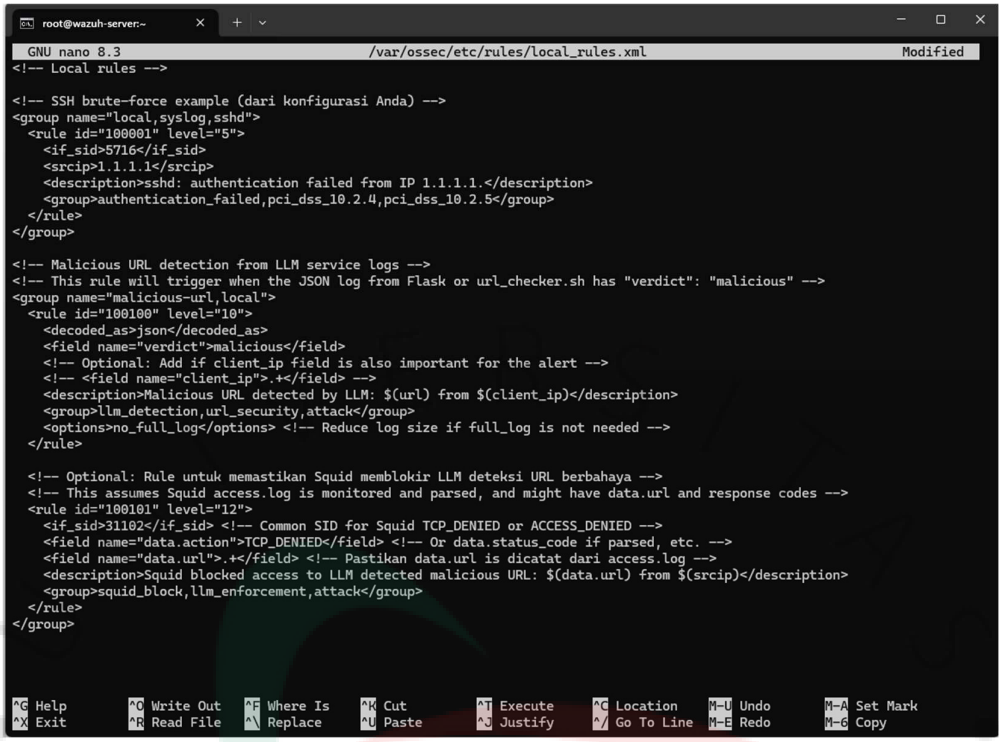
Gambar 5.5 Konfigurasi URL_checker.sh

Konfigurasi URL_checker.sh berfungsi sebagai penghubung/API antara predict.py dan squid.conf. Jika ada permintaan akses pada URL dan dilanjutkan kepada predict.py, maka URL_checker.sh melanjutkan kepada squid.conf untuk menentukan hasil prediksi URL yang telah dilakukan oleh predict.py. Jika predict.py menunjukkan hasil URL yang berbahaya, maka URL_checker.sh akan melanjutkan hasil URL tersebut sebagai berbahaya dan diblokir oleh Squid melalui konfigurasi squid.conf.



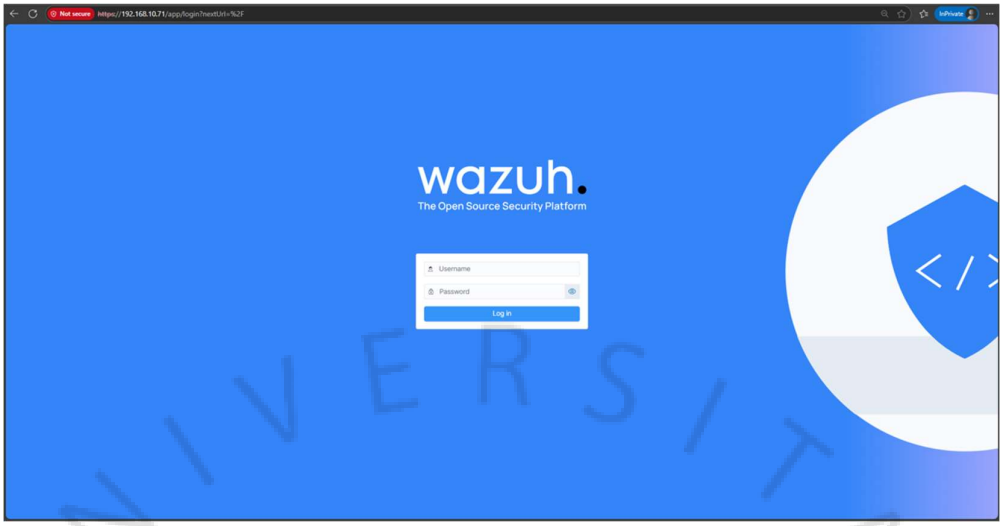
Gambar 5.6 Konfigurasi ossec.conf

Merujuk pada gambar 5.6, konfigurasi ossec.conf ditujukan agar Wazuh dapat menerima log dalam bentuk .json dan menunjukkan log pemblokiran yang dilanjutkan oleh konfigurasi URL_checker.sh.



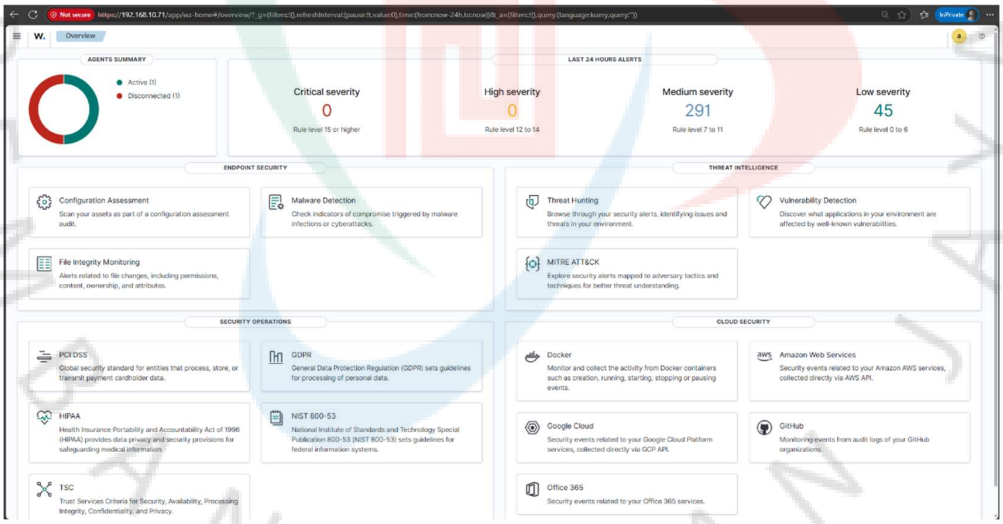
Gambar 5.7 Konfigurasi local_rules.xml

Konfigurasi local_rules.xml digunakan untuk membuat aturan khusus dan menulis log buatan/custom pada Wazuh jika terjadi adanya akses pada URL berbahaya. Nomor identifikasi aturan ini adalah 100100, dan ketika dimunculkan pada Wazuh, maka nomor identifikasi akan ditunjukkan beserta dengan pesan pemblokiran buatan/custom yang telah dibuat.



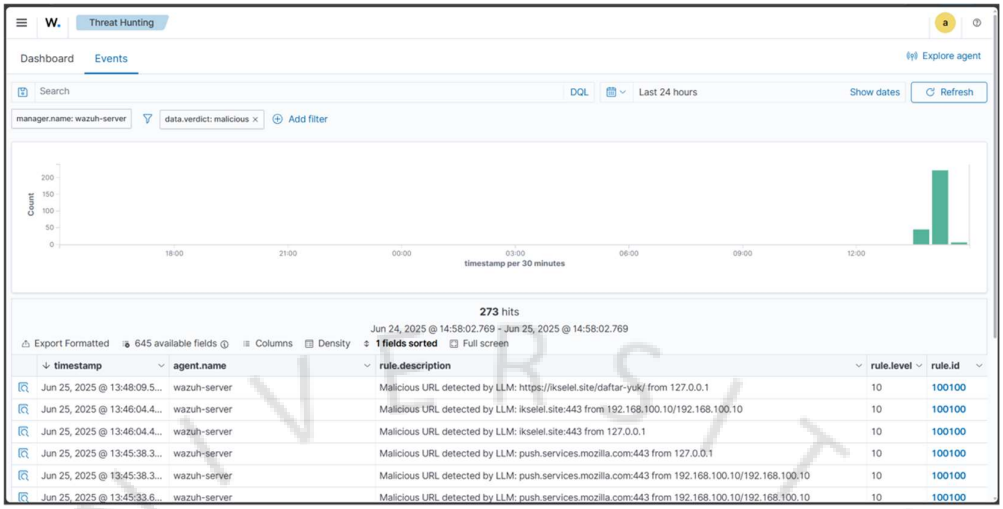
Gambar 5.8 Halaman login Wazuh

Halaman login Wazuh ditampilkan ketika mengakses alamat IP lokal yang dikonfigurasi secara otomatis. Kredensial bawaan yang terpasang pada Wazuh ini adalah; username: “admin” dan password “admin”.



Gambar 5.9 Dashboard Wazuh

Setelah melakukan login, akan muncul tampilan seperti pada gambar 5.9. Dalam penelitian ini, menu yang digunakan yaitu adalah “Threat Hunting”, guna untuk menunjukkan log hasil pemblokiran URL.



Gambar 5.10 Dashboard Threat Hunting

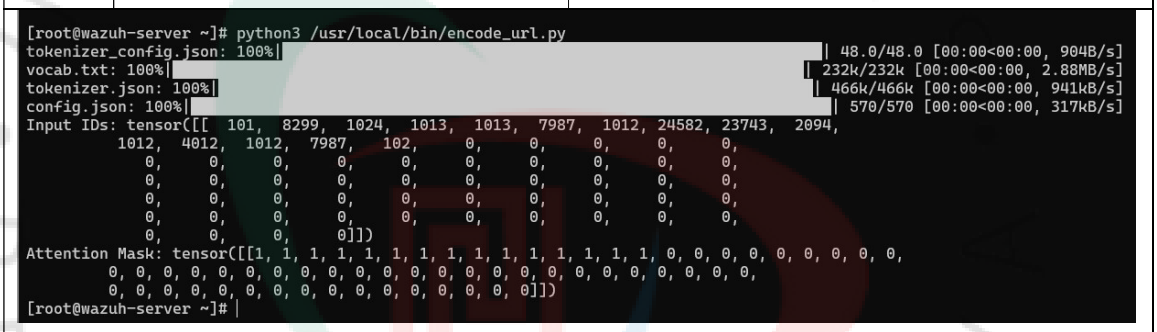
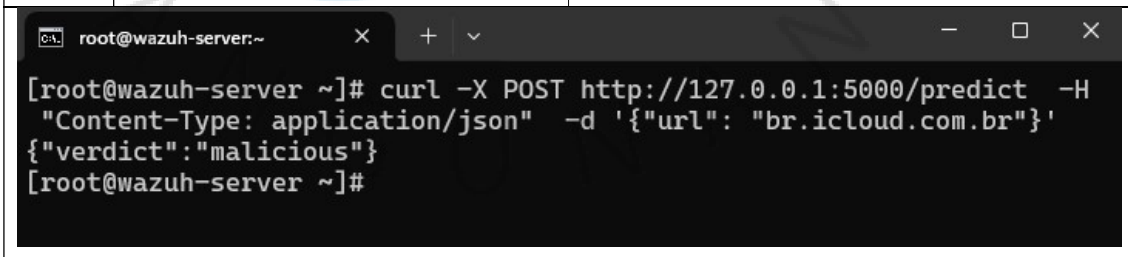
Gambar 5.10 adalah dashboard yang digunakan untuk memantau log, merujuk pada penelitian ini adalah log pemblokiran akses terhadap URL berbahaya. Gambar diatas menunjukkan adanya akses pada URL berbahaya dan telah ditampikan pada “rule.description” dan “rule.id” yang sesuai dengan konfigurasi local_rules.xml dan ossec.conf.

5.2 Uji Coba Hasil Pengujian

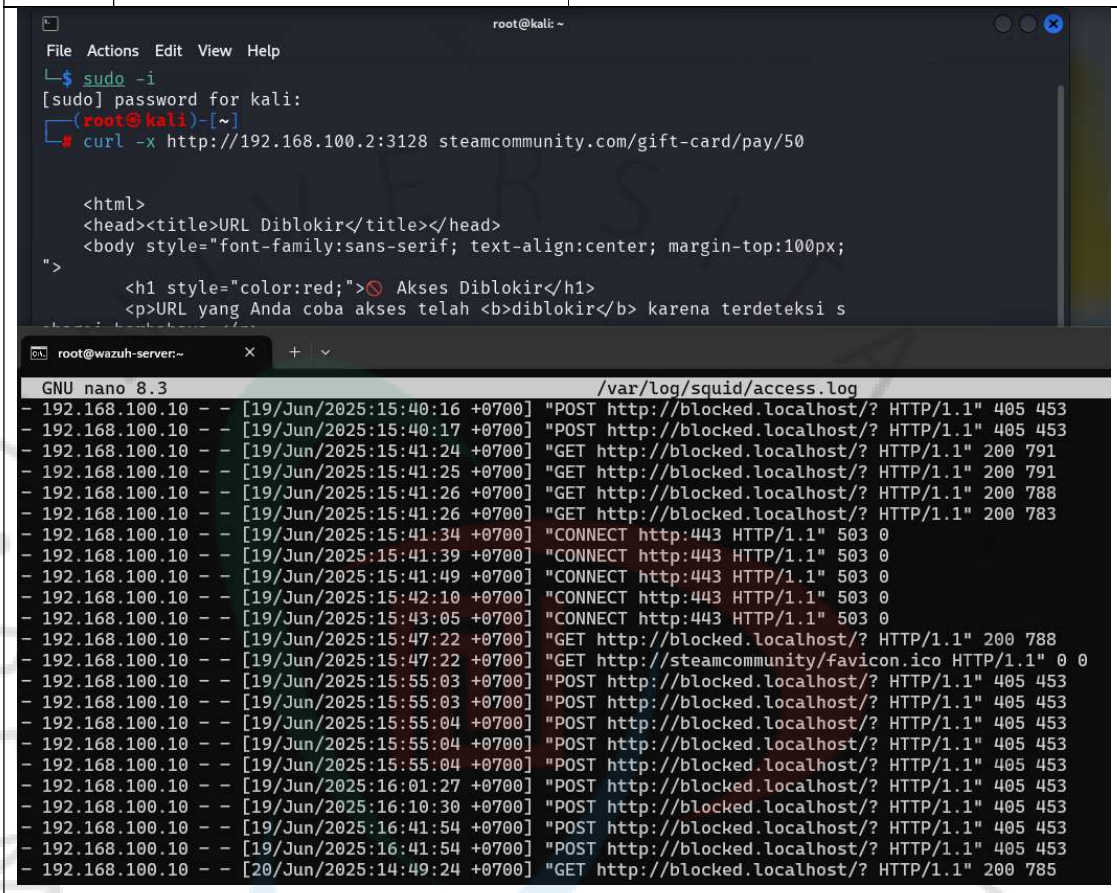
Pengujian pada sistem dilakukan dengan metode White Box dan metode Black Box yang sebelumnya telah dilampirkan pada sub-bab 4.5. Berikut adalah isi dari pengujian yang telah dilakukan.

5.2.1 Hasil Pengujian White Box

Tabel 5.3 Pengujian White Box

<table>
<tr><th>No</th><th>Modul yang Diuji</th><th>Output yang Diharapkan</th></tr>
<tr><td>1</td><td>Tokenisasi URL</td><td>Format input_ids dan attention_mask valid dan sesuai</td></tr>
<tr><td colspan="3"> 
</td></tr>
<tr><td colspan="3">Hasil: Sesuai
Hasil sesuai dengan menunjukkan hasil tokenisasi pada URL yang dimasukkan untuk melakukan prediksi “benign” atau “malicious”.</td></tr>
<tr><td colspan="3"></td></tr>
<tr><td>2</td><td>Klasifikasi oleh Model BERT</td><td>Output label malicious atau benign</td></tr>
<tr><td colspan="3"> 
</td></tr>
<tr><td colspan="3">Hasil: Sesuai
Hasil sesuai dengan dimunculkannya hasil prediksi “verdict”:”malicious” terhadap URL yang benar-benar berbahaya.</td></tr>
</table>

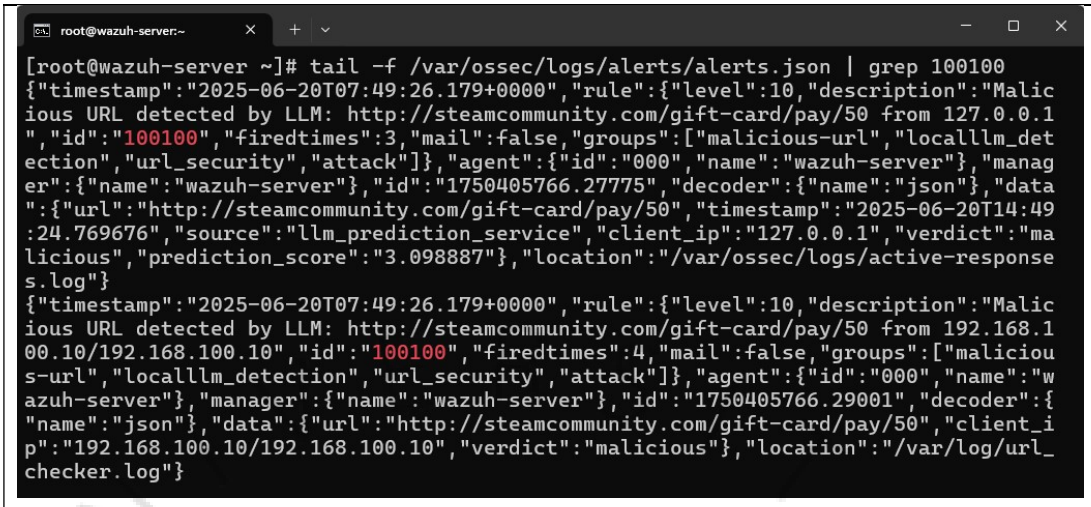
3	Flask API dan Squid Proxy	Flask API menerima respon dan Squid Proxy memblokir akses URL berbahaya



Hasil: **Sesuai**

Hasil sesuai dengan dimunculkannya hasil pemblokiran yang dilakukan oleh Squid Proxy dan halaman blokir oleh Flask API.

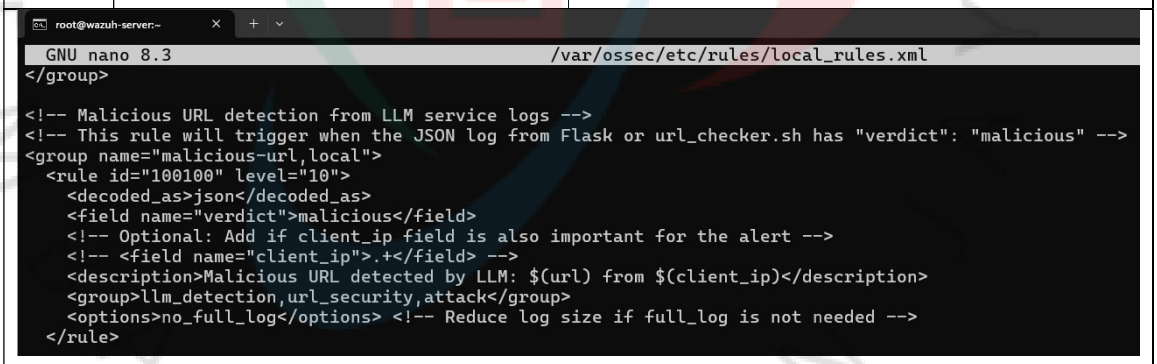
4	Logging ke Wazuh	Alert muncul pada log alerts Wazuh CLI



Hasil: **Sesuai**

Hasil sesuai dengan dimunculkannya nomor identifikasi aturan/rule 100100 dan URL berbahaya yang diakses.

5	Rule Matching dan Active Response	Penyesuaian Rules yang sudah ada dan penambahan rules untuk pemblokiran URL berbahaya dari Flask API dan Squid Proxy

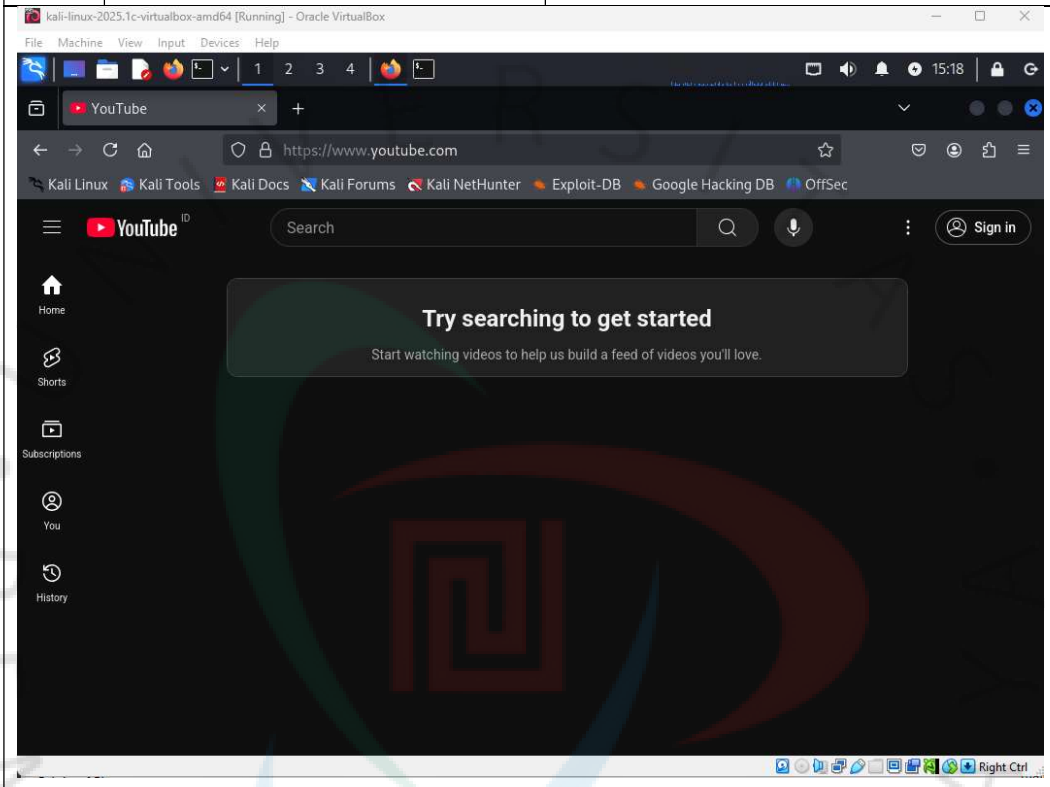


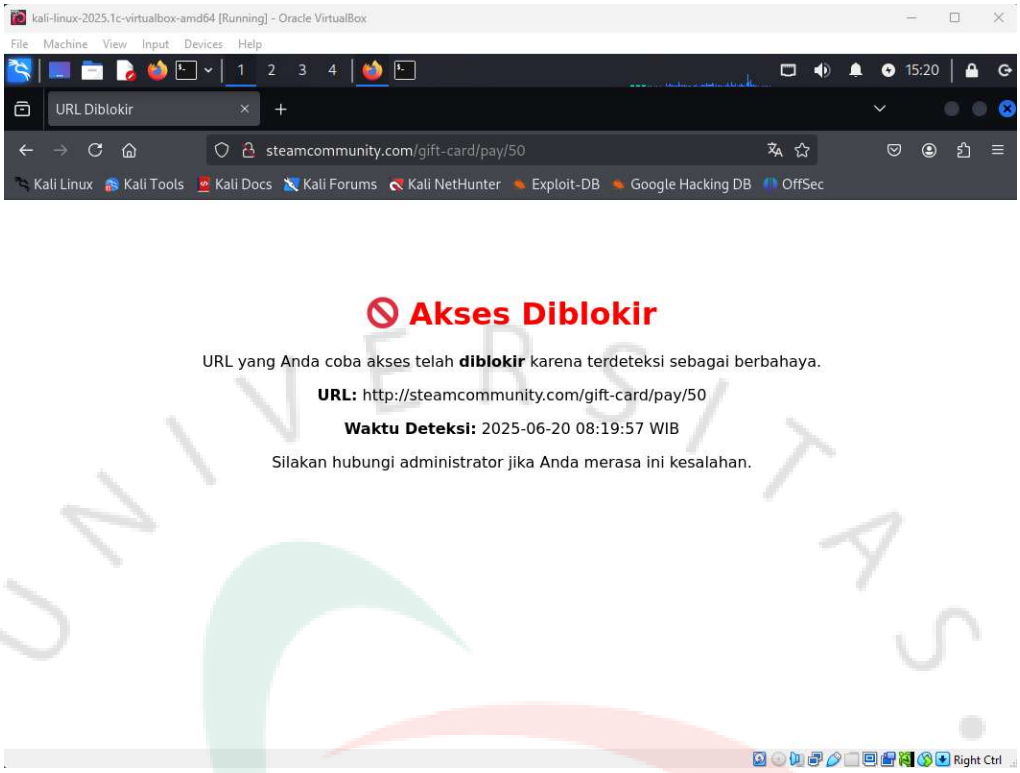
Hasil: **Sesuai**

Hasil sesuai merujuk pada konfigurasi yang ditunjukkan pada gambar dan berhasil dimunculkan pada Wazuh.

5.2.2 Hasil Pengujian Black Box

Tabel 5.4 Pengujian Black Box

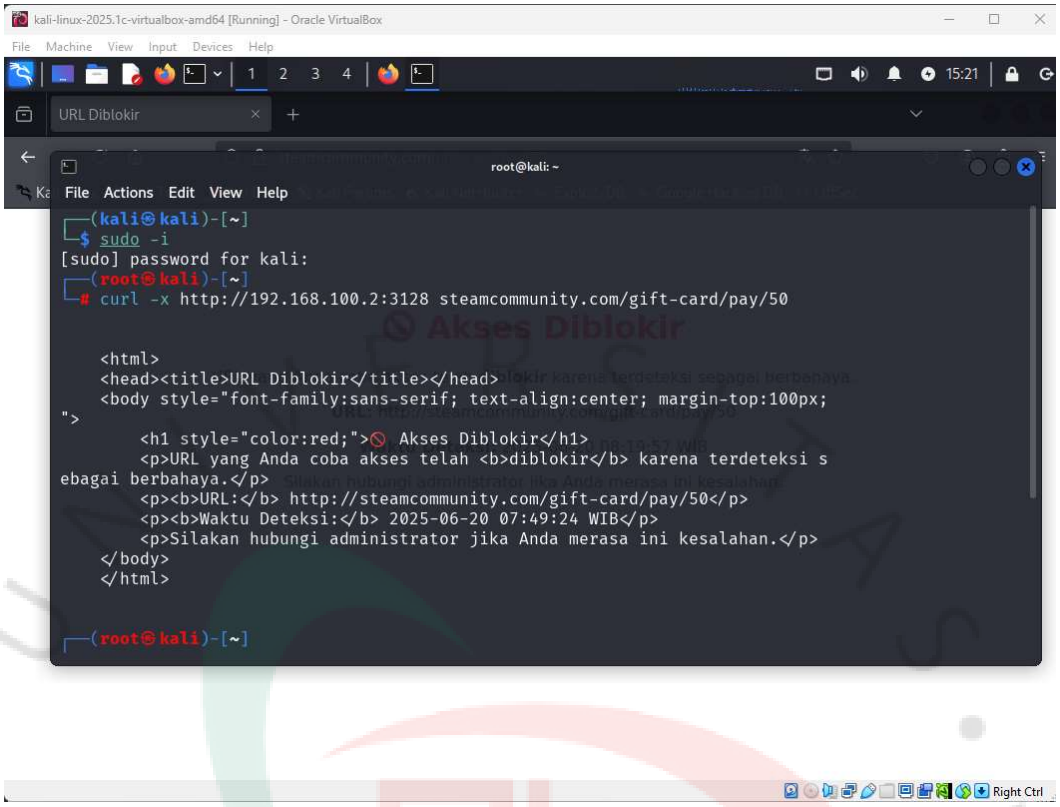
No	Skenario Pengujian	Hasil yang Diharapkan
1	Client mengakses URL aman	Halaman ditampilkan dengan normal (youtube.com)
		
Hasil: **Sesuai** Hasil sesuai dengan menunjukkan halaman web sah “youtube.com” yang bisa diakses.		
[redacted]		
2	Client mengakses URL berbahaya	Akses diblokir, halaman blokir Squid tampil



Hasil: **Sesuai**

Hasil sesuai dengan percobaan akses kepada URL berbahaya dan akses terblokir.

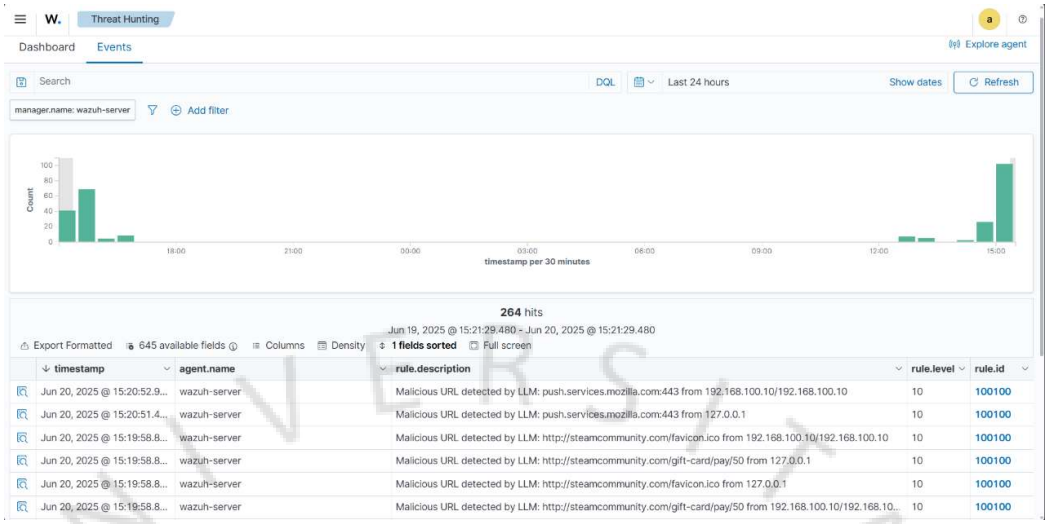
3	Flask API menerima dan merespons klasifikasi URL	Flask API menerima permintaan dan mengembalikan halaman Akses Diblokir



Hasil: **Sesuai**

Hasil sesuai dengan merujuk pada Flask API yang menunjukkan halaman pemblokiran.

4	Log dikirim ke Wazuh dan menampilkan alert	Alert muncul sesuai rule 100100 pada local_rules.xml



Hasil: **Sesuai**

Hasil sesuai ditunjukkan dengan adanya aturan/rule khusus yang dibuat untuk pemblokiran akses pada URL berbahaya.