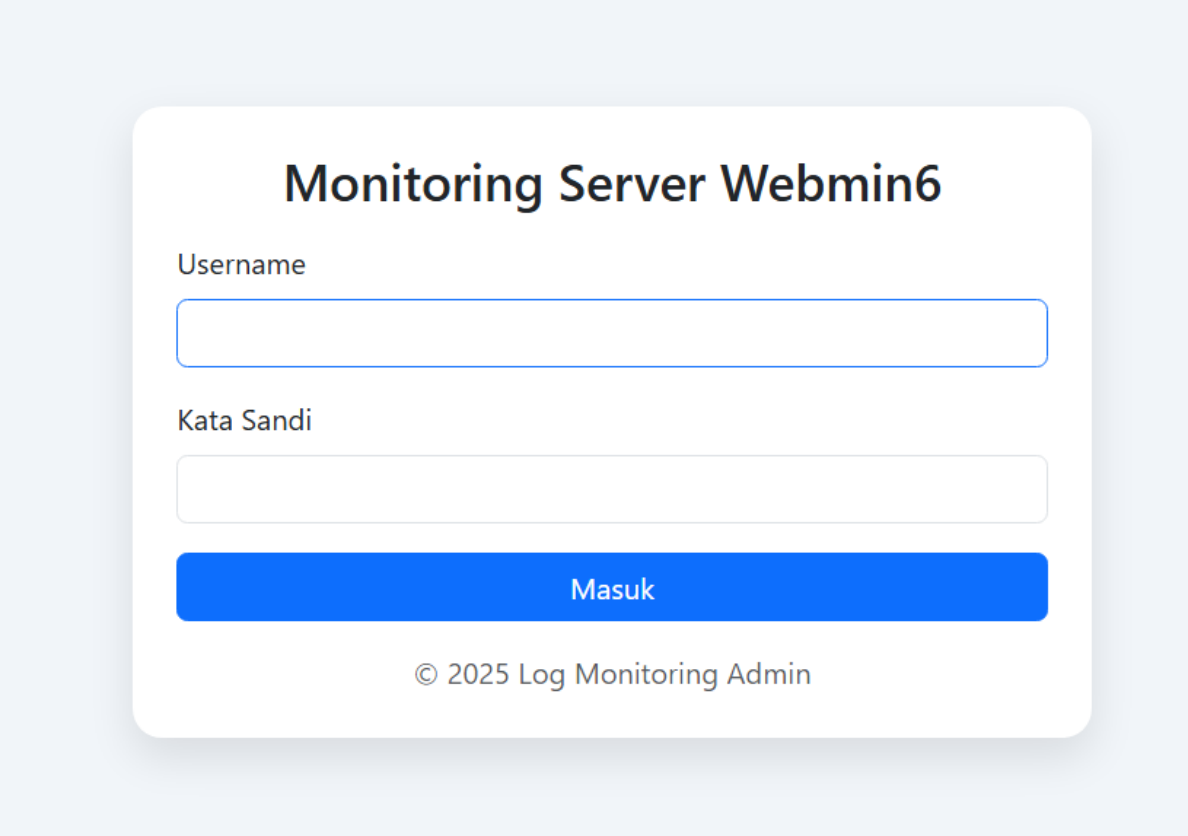
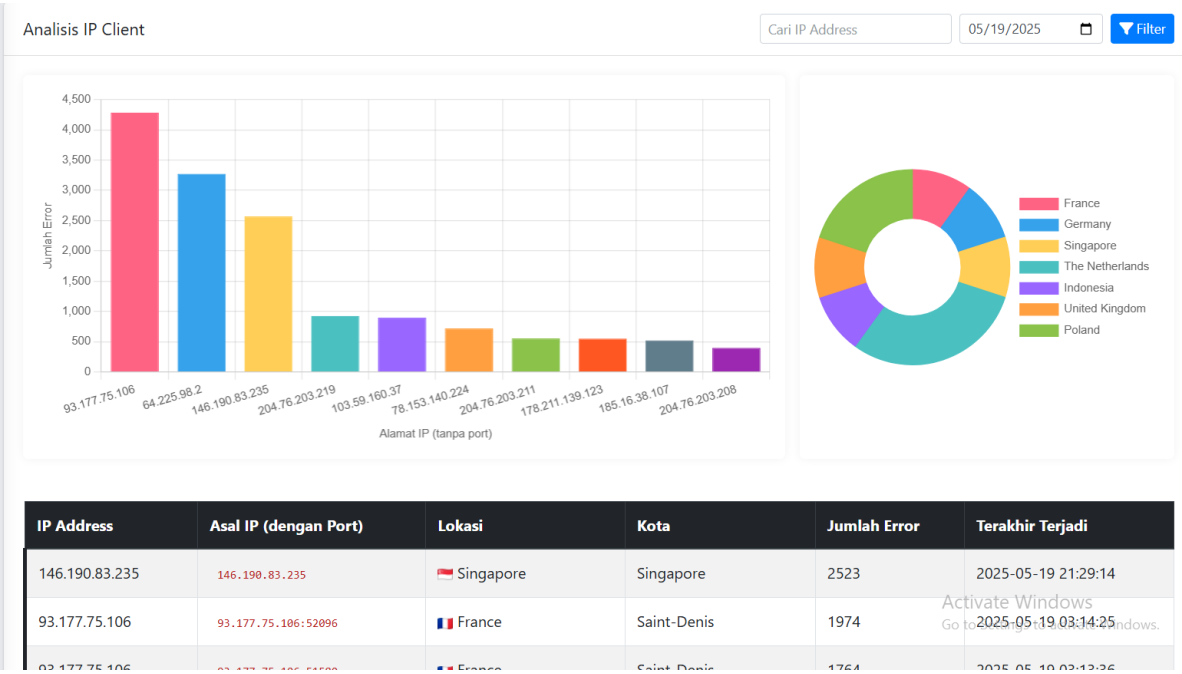
App Logwebmin

Aplikasi ini adalah hasil pengembangan mandiri Dinas Kominfo Bartim untuk memonitoring dan memblokir aktivitas client yang mencurigakan pada server webmin (server apache).





IP Address	Asal IP (dengan Port)	Lokasi	Kota	Jumlah Error	Terakhir Terjadi
146.190.83.235	146.190.83.235	Singapore	Singapore	2523	2025-05-19 21:29:14
93.177.75.106	93.177.75.106:52096	France	Saint-Denis	1974	2025-05-19 03:14:25
93.177.75.106		France	Saint-Denis	1764	2025-05-19 03:13:36

Activate Windows
Go to Settings to activate Windows.

Apikasi ini telah diinstall pada server Apache dengan spesifikasi:

1. CPU 8x Intel(R) Xeon(R) Bronze 3206R CPU @ 1.90GHz (1 Socket)
2. RAM 32 GB
3. HDD 100GB
4. OS Linux Mint 20.2

Aplikasi dapat diunduh pada link **Cloud Repositori Bartim**.

Petunjuk Instalasi:

1. Upload file aplikasi pada /var/www/html/logwebmin

2. Buat konfigurasi virtual domain pada /etc/apache2/sites-available/000-default.conf

```
<VirtualHost *:80>
    ServerName log6.baritotimurkab.go.id
    DocumentRoot /var/www/html/logwebmin/public
    RewriteEngine on
    DocumentRoot /var/www/html/logwebmin/public
    RewriteCond %{SERVER_NAME} =www.log6.baritotimurkab.go.id [OR]
    RewriteCond %{SERVER_NAME} =log6.baritotimurkab.go.id
    RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
</VirtualHost>
```

3. Buat ssl dengan letsencrypt

4. Buat konfigurasi virtual domain ssl pada /etc/apache2/sites-available/default-ssl.conf

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerName log6.baritotimurkab.go.id
    DocumentRoot /var/www/html/logwebmin/public

    SSLEngine on
    SSLCertificateFile /etc/letsencrypt/live/log6.baritotimurkab.go.id/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/log6.baritotimurkab.go.id/privkey.pem
    Include /etc/letsencrypt/options-ssl-apache.conf

    ErrorLog ${APACHE_LOG_DIR}/logwebmin_error.log
    CustomLog ${APACHE_LOG_DIR}/logwebmin_access.log combined
</VirtualHost>
</IfModule>
```

5. Buat inisiasi fitur socket
nano /etc/systemd/system/ipblock.service

isi:

```
[Unit]
Description=IP Blocking Service
Requires=ipblock.socket

[Service]
Type=simple
ExecStart=/usr/local/bin/ipblock-daemon
User=www-data
Group=www-data
Restart=always
RestartSec=5
Environment="SOCKET_PATH=/var/www/html/logwebmin/storage/sockets/ipblock.sock"

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

nano /etc/systemd/system/ipblock.socket
isi:

```
[Unit]

Description=IP Blocking Socket

[Socket]
ListenStream=/var/www/html/logwebmin/storage/sockets/ipblock.sock
SocketUser=www-data
SocketGroup=www-data
SocketMode=0660

[Install]
WantedBy=sockets.target
```

nano /usr/local/bin/ipblock-daemon (berikan permissions 0755 dan ownership www-data)
isi:

```
#!/usr/bin/python3

import socket
import os
import subprocess
import re

SOCKET_PATH = '/var/www/html/logwebmin/storage/sockets/ipblock.sock'
```

```
# Setup socket directory
os.makedirs(os.path.dirname(SOCKET_PATH), exist_ok=True)

# Remove old socket if exists
try:
    os.unlink(SOCKET_PATH)
except FileNotFoundError:
    pass

# Create and bind socket
sock = socket.socket(socket.AF_UNIX, socket.SOCK_STREAM)
sock.bind(SOCKET_PATH)
sock.listen(1)
os.chmod(SOCKET_PATH, 0o660)
os.chown(SOCKET_PATH, 33, 33)  # www-data user and group ID

def validate_ip(ip):
    """Validate IPv4 address format"""
    pattern = r'^(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-
5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)$'
    return re.match(pattern, ip) is not None

while True:
    conn, _ = sock.accept()
    try:
        data = conn.recv(1024).decode().strip()
        if not data:
            continue

        # Handle unblock command
        if data.startswith('unblock:'):
            ip = data.split(':')[1]
            if validate_ip(ip):
                try:
                    subprocess.run(['sudo', '/usr/local/bin/unblock-ip.sh', ip], check=True)
                    conn.send(b"OK")
                except subprocess.CalledProcessError as e:
                    conn.send(f"ERROR: Unblock failed - {str(e)}".encode())
            else:
                conn.send(b"ERROR: Invalid IP format for unblock")

        # Handle block command (original functionality)
        else:
            ip = data
            if validate_ip(ip):
                try:
```

```
                subprocess.run(['sudo', '/usr/local/bin/block-ip.sh', ip], check=True)
                conn.send(b"OK")
            except subprocess.CalledProcessError as e:
                conn.send(f"ERROR: Block failed - {str(e)}".encode())
        else:
            conn.send(b"ERROR: Invalid IP format for block")

    except Exception as e:
        conn.send(f"ERROR: Server error - {str(e)}".encode())
    finally:
        conn.close()
```

nano /usr/local/bin/block-ip.sh
isi:
```
#!/bin/bash
IP=$1
iptables -A INPUT -s $IP -j DROP
```

nano /usr/local/bin/unblock-ip.sh
isi:
```
#!/bin/bash
IP=$1
iptables -D INPUT -s $IP -j DROP
```

6. Berikan akses www-data untuk menjalankan shell block-ip.sh dan unblock-ip.sh
nano /etc/sudoers
isi:
```
#

# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification
```

```
# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo    ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

#fungsi tambahan blokir lewa app logwebmin
www-data ALL=(ALL) NOPASSWD: /usr/local/bin/block-ip.sh
www-data ALL=(ALL) NOPASSWD: /usr/local/bin/unblock-ip.sh
```

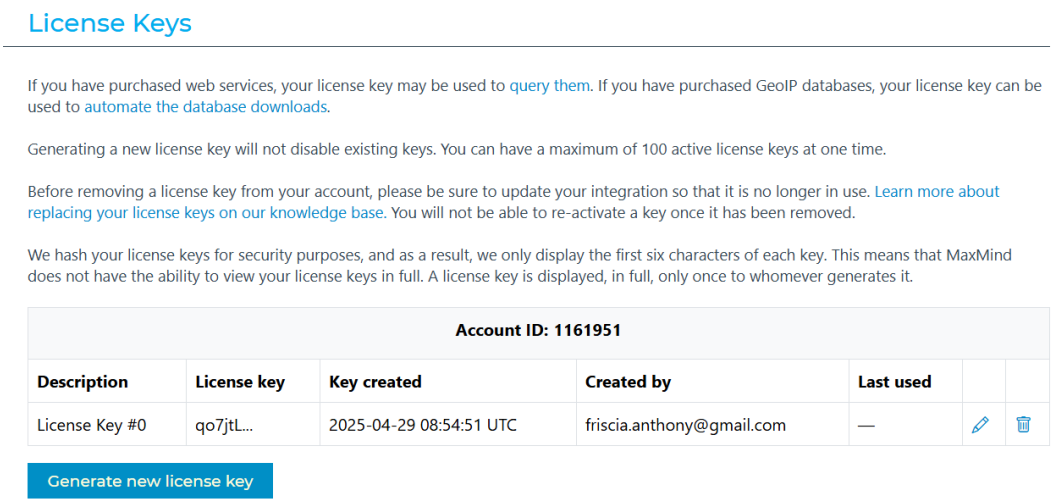
7. Jalankan service daemon

sudo systemctl daemon-reload
sudo systemctl restart ipblock.socket ipblock.service
sudo systemctl enable ipblock.socket

8. Verifikasi service

sudo systemctl status ipblock.service
journalctl -u ipblock.service -f

9. Pastikan anda telah mendaftar akun pada https://www.maxmind.com/ untuk mendapatkan license-key seperti pada contoh berikut:



agar dapat diinisiasikan pada .env

.env (/var/www/html/logwebmin)

```
APP_NAME=Laravel
APP_ENV=production
APP_KEY=base64:6Nfn6cKTJBemSHH99Cn5Kb7e8u0sVuYCGHjvtg0/rLQ=
APP_DEBUG=true
APP_URL=http://localhost
MAXMIND_LICENSE_KEY=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
```

Revisi #17
Dibuat 20 Mei 2025 00:10:59 oleh Friscia Anthony
Diperbaharui 12 Oktober 2025 00:14:58 oleh Friscia Anthony