Pengamanan Server

- Pengamanan Server Menggunakan Fail2Ban
- App Logwebmin
- Monitoring Koneksi Aktif ke Server
- Cek Lokasi File Exploit pada Root Web
- Mengamankan PHP

Pengamanan Server Menggunakan Fail2Ban

Fail2Ban disini ditujukan untuk memblokir IP client yang melakukan aktivitas berbahaya pada aplikasi/website pada server.

Installasi:
`sudo apt install fail2ban`

Konfigurasi Dasar untuk SSH
`sudo nano /etc/fail2ban/jail.local`

isi:
```
[sshd]
enabled = true
port = ssh
filter = sshd
logpath = /var/log/auth.log
maxretry = 5
bantime = 1h
findtime = 10m

[apache-auth]
enabled = true
port = http,https
logpath = /var/log/apache*/*error.log
maxretry = 5
```

Restart Fail2Ban
`sudo systemctl restart fail2ban`
sudo systemctl enable fail2ban

Cek Status Jail
`sudo fail2ban-client status`

Metode pengamanan paling baik dengan Fail2Ban adalah dengan menyesuaikan perilaku atau aktivitas client dalam mengakses aplikasi/website pada server. Cara mudahnya adalah melihat error yang dihasil pada tail -f /var/log/apache2/error.log.

Dari error.log tersebut dihasilkan informasi aktivitas client yang melakukan upaya berbahaya yang menghasilkan error pada server apache.

Sebagai contoh saya mau memblokir IP client dengan aktivitas error berikut:
\[Fri May 16 19:32:01.013545 2025] \[proxy_fcgi\:error] \[pid 1496923] \[client 159.65.1.31:55208] AH01071: Got error 'PHP message: PHP Warning: Constant WP_MEMORY_LIMIT already defined in /var/www/html/dpmdsos/public_html/wp-config.php on line 98' \[Fri May 16 19:32:01.330516 2025] \[proxy_fcgi\:error] \[pid 1496903] \[client 159.65.1.31:55174] AH01071: Got error 'PHP message: PHP Warning: The magic method Vc_Manager::__wakeup() must have public visibility in /var/www/html/dpupkp/public_html/wp-content/plugins/js_composer/include/classes/core/class-vc-manager.php on line 203' \[Fri May 16 19:32:01.505303 2025] \[proxy_fcgi\:error] \[pid 1496930] \[client 159.65.1.31:55479] AH01071: Got error 'PHP message: PHP Warning: Constant WP_MEMORY_LIMIT already defined in /var/www/html/dpmdsos/public_html/wp-config.php on line 98' \[Fri May 16 19:32:01.726184 2025] \[proxy_fcgi\:error] \[pid 1497083] \[client 159.65.1.31:55322] AH01071: Got error 'PHP message: PHP Warning: The magic method Vc_Manager::__wakeup() must have public visibility in /var/www/html/disperpusip/public_html/wp-content/plugins/js_composer/include/classes/core/class-vc-manager.php on line 203' \[Fri May 16 19:32:01.773397 2025] \[autoindex\:error] \[pid 1487621] \[client 209.38.120.221:57305] AH01276: Cannot serve directory /var/www/html/inspektorat/public_html/.well-known/: No matching DirectoryIndex (index.html,index.cgi,index.pl,index.php,index.xhtml,index.htm) found, and server-generated directory index forbidden by Options directive \[Fri May 16 19:32:01.836303 2025] \[autoindex\:error] \[pid 1487621] \[client 209.38.120.221:57305] AH01276: Cannot serve directory /var/www/html/inspektorat/public_html/.well-known/pki-validation/: No matching DirectoryIndex (index.html,index.cgi,index.pl,index.php,index.xhtml,index.htm) found, and server-generated directory index forbidden by Options directive \[Fri May 16 19:32:01.907570 2025] \[autoindex\:error] \[pid 1487621] \[client 209.38.120.221:57305] AH01276: Cannot serve directory /var/www/html/inspektorat/public_html/.well-known/acme-challenge/: No matching DirectoryIndex (index.html,index.cgi,index.pl,index.php,index.xhtml,index.htm) found, and server-generated directory index forbidden by Options directive

maka saya membuat file filter baru di `nano /etc/fail2ban/filter.d/apache-php-errors.conf`

isi:

```
[Definition]
failregex = ^\[[^\]]*\] \[[^\]]*\] \[pid [0-9]*\] \[client <HOST>:[0-9]*\] AH01071: Got error 'PHP message: PHP Warning:.*$
            ^\[[^\]]*\] \[autoindex:error\] \[pid [0-9]*\] \[client <HOST>:[0-9]*\] AH01276: Cannot serve directory.*$
ignoreregex =
```

Note: gunakan Chatgpt atau Deepseek untuk membuat koding filter

Kemudian tambahkan jail baru di `/etc/fail2ban/jail.local`

Isi:

```
[apache-php-errors]
enabled  = true
port     = http,https
```

```
filter   = apache-php-errors
logpath  = /var/log/apache2/error.log
maxretry = 5
findtime = 10m
bantime  = 1h
```

Restart Fail2Ban

```
sudo systemctl restart fail2ban
```

Kemudian cek untuk melihat IP client sudah diblokir

```
sudo fail2ban-client status apache-php-errors
```

Silahkan tambahkan lagi filter lainnya sesuai dengan error.log yang diperoleh.

Apabila suatu waktu anda pun tidak dapat mengakses aplikasi/website pada server anda. Maka ada kemungkinan anda pun juga sudah diblokir oleh filter Fail2Ban yang anda buat sendiri. Maka kerjakan Tips berikut jika terjadi.

sudo fail2ban-client status <nama_jail>
# Contoh:

```
sudo fail2ban-client status apache-php-errors
```

Jika dipastikan memang IP anda yang diblokir maka lakukan seperti contoh berikut:

```
sudo fail2ban-client set apache-wp-warnings unbanip 36.75.65.41
```

dan cara paling aman adalah dengan membuat **Whitelist IP Anda** di `nano /etc/fail2ban/jail.local`
Isi:

```
[DEFAULT]
ignoreip = 127.0.0.1/8 ::1 36.75.65.41
```

atau kurangi agresivitas jail:

```
[apache-php-errors]
enabled  = true
port     = http,https
filter   = apache-php-errors
logpath  = /var/log/apache2/error.log
maxretry = 10            #ubah dari 5 jadi 10
findtime = 30m           #ubah dari 10 jadi 30
bantime  = 1h            #ubah dari 24h jadi 1h
```

Lakukan Monitoring lol Fail2Ban:

```
sudo tail -f /var/log/fail2ban.log
```

Cek daftar IP yang sudah diblokir fail2ban:

```
for jail in $(fail2ban-client status | grep 'Jail list' | cut -d: -f2 | tr ',' '\n' | tr -d ' '); do
    echo "== Jail: $jail ==";
    fail2ban-client status "$jail" | grep 'Banned IP list';
    echo;
done
```

Buka semua IP yang diblokir ada filter fail2ban:

```
for jail in $(fail2ban-client status | grep 'Jail list:' | cut -d: -f2 | tr -d ' ' | tr ',' ' '); do
  echo "Checking jail: $jail"
  banned_ips=$(fail2ban-client status "$jail" | grep 'Banned IP list' | cut -d: -f2)
  for ip in $banned_ips; do
    fail2ban-client set "$jail" unbanip "$ip" && echo "→ Unbanned $ip from $jail"
  done
done
```

Buka Blokir khusus Jail tertentu. ini misalnya modsecurity-aggressive:

```
for ip in $(sudo fail2ban-client status modsecurity-aggressive | grep 'Banned IP list:' | awk -F': ' '{print $2}'); do
    sudo fail2ban-client set modsecurity-aggressive unbanip $ip
done
```

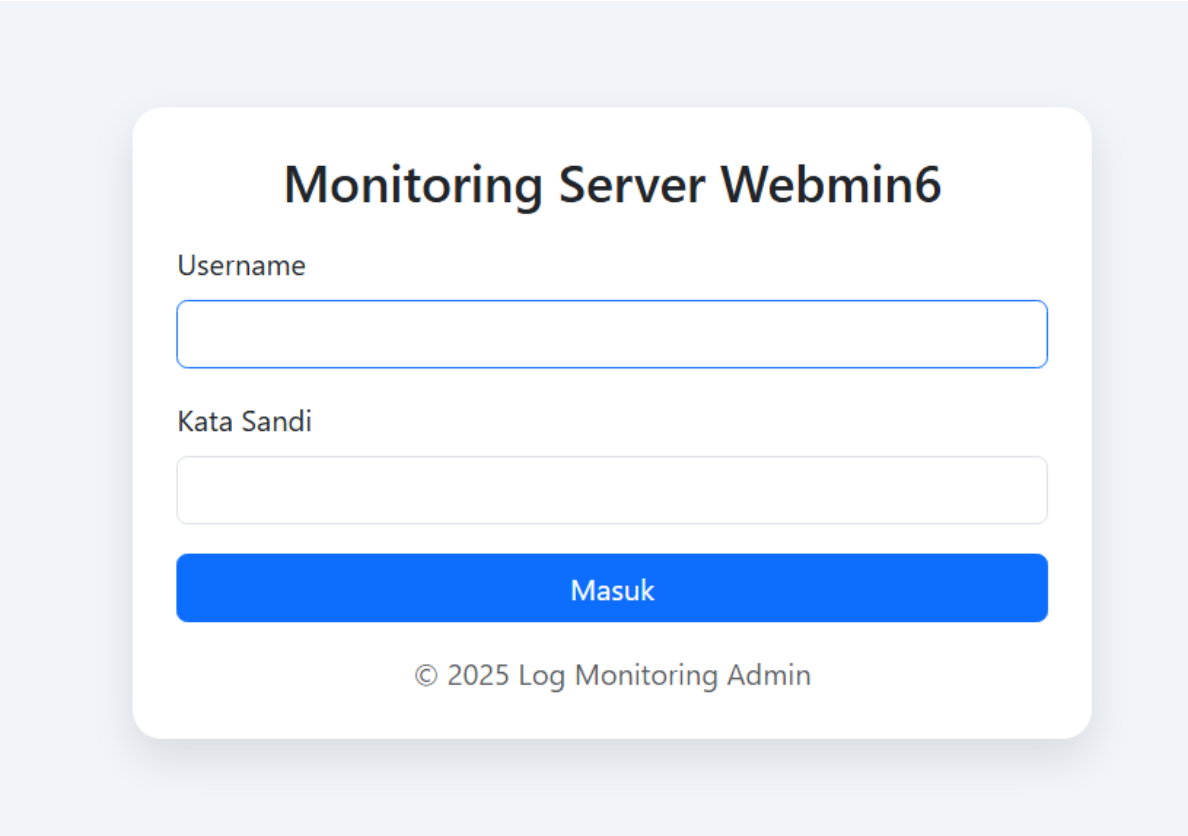
Cara Membuka blokir IP tertentu pada semua jail:

```
IP="182.8.131.150"
```

```
for jail in $(fail2ban-client status | grep 'Jail list:' | cut -d: -f2 | tr -d ' ' | tr ',' ' '); do
  echo "Checking jail: $jail"
  if fail2ban-client status "$jail" | grep -q "$IP"; then
    fail2ban-client set "$jail" unbanip "$IP" && echo "→ Unbanned $IP from $jail"
  else
    echo "No ban for $IP in $jail"
  fi
done
```

App Logwebmin

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





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
```

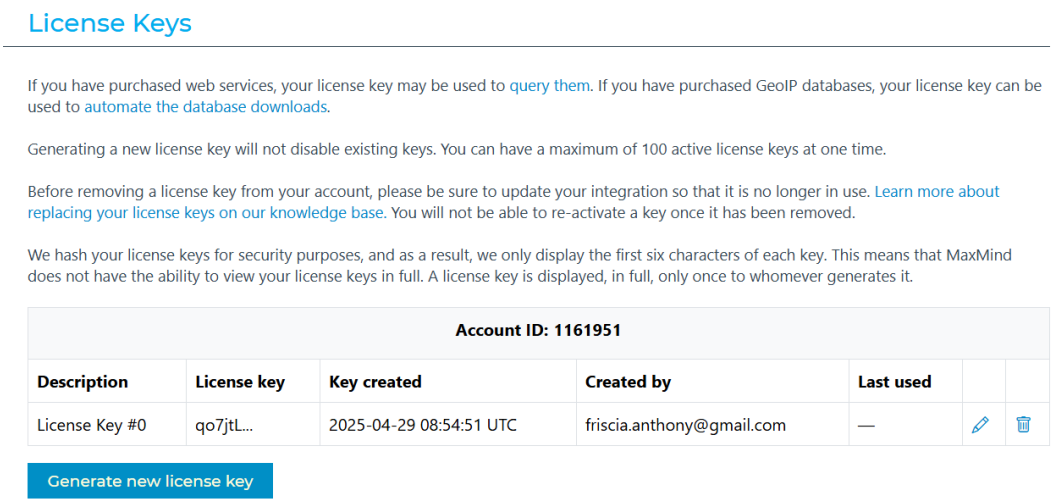
7. Jalankan service daemon

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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
```

Monitoring Koneksi Aktif ke Server

Untuk melihat koneksi aktif ke server Anda, termasuk IP yang terhubung, gunakan `netstat`: secara real time setiap 2 detik:

```
watch -n 2 netstat -ntu
```

contoh hasil:

```
Every 2.0s: netstat -ntu                    kominfos-0B: Tue Jun 24 21:28:23 2025  Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:3306          127.0.0.1:47308         ESTABLISHED
tcp        0      0 127.0.0.1:57164         127.0.0.1:556           ESTABLISHED
tcp        0      0 127.0.0.1:38366         127.0.0.1:3306          ESTABLISHED
tcp        0      0 127.0.0.1:556           127.0.0.1:57164         ESTABLISHED
tcp        0      0 192.168.80.51:22        196.251.70.166:57154    ESTABLISHED
tcp        0      0 192.168.80.51:41344     91.189.91.48:80         TIME_WAIT
tcp        0      0 192.168.80.51:2014      222.124.78.211:59143    ESTABLISHED
tcp        0      0 127.0.0.1:47308         127.0.0.1:3306          ESTABLISHED
tcp        0      0 127.0.0.1:3306          127.0.0.1:38366         ESTABLISHED
tcp        0      0 192.168.80.51:22        222.124.78.211:59160    ESTABLISHED
tcp6       0      0 192.168.80.51:443       114.125.120.16:41242    TIME_WAIT
tcp6       0      0 192.168.80.51:80        45.82.76.192:41150      SYN_RECV
tcp6       0     25 192.168.80.51:443       177.93.63.18:45129      CLOSING
udp        0      0 192.168.80.51:68        192.168.80.1:67         ESTABLISHED
udp        0      0 169.254.95.120:68       169.254.95.118:67       ESTABLISHED
```

Daftar Status Koneksi TCP & Penjelasannya

State	Penjelasan
ESTABLISHED	Koneksi telah terbentuk dan aktif digunakan untuk komunikasi data. Ini adalah koneksi yang “normal” dan siap untuk transfer data dua arah.
SYN_SENT	Klien telah mengirim paket SYN ke server dan sedang menunggu balasan (SYN-ACK). Ini adalah tahap awal dari koneksi outbound (keluar).
SYN_RECV	Server telah menerima SYN dari klien, membalas dengan SYN-ACK, dan sedang menunggu ACK dari klien untuk menyelesaikan proses handshake.
FIN_WAIT1	Aplikasi telah menutup koneksi, dan sistem telah mengirim FIN (permintaan tutup koneksi) ke pihak lain.

State	Penjelasan
FIN_WAIT2	Setelah menerima ACK dari FIN pertama, sistem menunggu FIN dari pihak lawan.
TIME_WAIT	Koneksi ditutup, tapi sistem menunggu selama periode waktu tertentu (biasanya 2x waktu maksimum segment life) untuk memastikan semua paket terlambat tidak akan menyebabkan kebingungan.
CLOSE_WAIT	Server menerima FIN dari klien, dan menunggu aplikasinya untuk menutup koneksi juga.
CLOSING	Kedua sisi telah mengirim FIN, tapi belum semua ACK diterima.
LAST_ACK	Sistem telah mengirim FIN (balasan), menunggu ACK dari pihak lain sebelum sepenuhnya menutup koneksi.
CLOSED	Koneksi telah ditutup sepenuhnya, tidak ada komunikasi lebih lanjut.
LISTEN	Server menunggu koneksi masuk pada port tertentu (passive open). Biasanya terlihat pada server seperti web server, SSH, dll.

Contoh Alur Normal TCP (3-way handshake + closing)

Pembukaan koneksi:

1. Klien: `SYN_SENT`
2. Server: `SYN_RECV`
3. Keduanya: `ESTABLISHED`

Penutupan koneksi:

1. Pengirim: `FIN_WAIT1`
2. Penerima: `CLOSE_WAIT`
3. Pengirim: `FIN_WAIT2`
4. Penerima: `LAST_ACK`
5. Keduanya: `TIME_WAIT`
6. Akhir: `CLOSED`

Status yang Perlu Diwaspadai

- **SYN_RECV** banyak → kemungkinan *SYN flood attack*
- **TIME_WAIT** banyak → bisa akibat skrip web atau API client yang membuka-tutup koneksi terus-menerus
- **CLOSE_WAIT** menggantung → aplikasi Anda tidak menutup koneksi dengan benar (potensi memory leak)

Cek Lokasi File Exploit pada Root Web

Cek Lokasi file Exploit yang menggunakan **shell_exec**, **curl_exec** dan simpan hasilnya di hasil_scan.txt

```
cd /var/www/html/root_aplikasi
```

```
grep -RniE 'eval\s*\(|base64_decode\s*\(|assert\s*\(|shell_exec\s*\(|system\s*\(|passthru\s*\(|file_get_contents\s*\(' . > hasil_scan.txt
```

```
grep -RniE 'eval\s*\(|base64_decode\s*\(|assert\s*\(|curl_exec\s*\(|system\s*\(|passthru\s*\(|file_get_contents\s*\(' . > hasil_scan2.txt
```

lihat hasilnya lalu hapus file tersebut (paling mencurigakan file .php yang berada di public atau storage)

Mengamankan PHP

Mengamankan PHP Apache

Untuk membatasi serangan hacker pada PHP dalam instalasi LAMPP (Linux, Apache, MySQL, PHP, PHPMyAdmin), Anda dapat mengambil beberapa langkah keamanan untuk melindungi aplikasi web Anda dari berbagai jenis serangan. Berikut adalah beberapa konfigurasi dan praktik keamanan umum yang dapat membantu Anda:

Update PHP
Pastikan Anda menjalankan versi PHP yang paling baru dan selalu diperbarui. Setiap pembaruan PHP biasanya mencakup perbaikan keamanan. Anda juga dapat mempertimbangkan untuk menggunakan PHP versi 8.x atau yang lebih baru, karena ini mendukung peningkatan keamanan dan kinerja.

Konfigurasi PHP
Edit file **php.ini** di **/etc/php/8.1/cli/php.ini** dan/atau **/etc/php/8.1/apache2/php.ini** dan terapkan konfigurasi keamanan berikut:

- Batasi maksimum ukuran upload file dengan parameter **`upload_max_filesize`** dan **`post_max_size`**.
- Matikan **`register_globals`**.
- Matikan **`allow_url_fopen`** dan **`allow_url_include`** untuk mencegah inklusi dan akses ke URL eksternal.
- Setel **`display_errors`** menjadi **Off** dalam produksi untuk menghindari menampilkan kesalahan PHP ke publik.
- Aktifkan disable_functions untuk memblokir fungsi-fungsi berbahaya jika Anda tahu apa yang Anda lakukan.
- Aktifkan open_basedir untuk membatasi direktori yang diizinkan PHP untuk mengakses.

Menggunakan opsi **`disable_functions`** dalam konfigurasi PHP dapat membantu mengurangi risiko keamanan dengan memblokir fungsi-fungsi tertentu yang dapat digunakan oleh penyerang untuk melakukan aktivitas berbahaya. Namun, penggunaan **`disable_functions`** harus dilakukan dengan hati-hati, karena mematikan fungsi-fungsi tertentu dapat mempengaruhi fungsi normal dari aplikasi PHP Anda. Beberapa fungsi yang sering kali dinonaktifkan karena potensi risiko keamanan termasuk:

1. **exec**: Fungsi `exec()` digunakan untuk menjalankan perintah shell dari dalam skrip PHP. Ini harus dinonaktifkan untuk mencegah penyerang menjalankan perintah shell di server Anda.

2. **system**: Sama seperti `exec()`, `system()` juga digunakan untuk menjalankan perintah shell. Ini harus dinonaktifkan.
3. **shell_exec**: Fungsi `shell_exec()` juga digunakan untuk menjalankan perintah shell dan harus dinonaktifkan.
4. **passthru**: `passthru()` adalah fungsi lain yang dapat digunakan untuk menjalankan perintah shell. Ini juga harus dinonaktifkan.
5. **popen**: `popen()` memungkinkan skrip PHP untuk membuka pipe ke proses eksternal. Ini harus dinonaktifkan.
6. **proc_open**: Fungsi ini serupa dengan `popen()` dan juga digunakan untuk menjalankan proses eksternal. Ini harus dinonaktifkan.
7. **eval**: Fungsi `eval()` digunakan untuk mengeksekusi kode PHP dari string. Penyerang dapat memanfaatkannya dengan memasukkan kode berbahaya ke dalam string.
8. **assert:** Fungsi `assert()` juga dapat digunakan untuk mengeksekusi kode PHP dari string. Ini harus dinonaktifkan.
9. **system_popen**: Ini adalah alternatif lain untuk `popen()` dan harus dinonaktifkan.
10. **dl**: Fungsi `dl()` digunakan untuk memuat ekstensi PHP dinamis. Ini dapat membuka celah keamanan jika dinonaktifkan.
11. **proc_close:** Ini adalah fungsi terkait dengan `proc_open()` yang digunakan untuk menutup proses yang telah dibuka dengan `proc_open()`. Dinonaktifkan juga.
12. **pcntl_exec**: Fungsi ini memungkinkan penggunaan fungsi `exec()` dalam lingkungan PHP-FPM. Jika Anda tidak menggunakannya, disarankan untuk dinonaktifkan.

Cari baris yang mengandung **`disable_functions`**. Jika Anda tidak menemukannya, Anda dapat menambahkannya di bagian paling bawah file **php.ini**.

Tambahkan nama fungsi-fungsi yang ingin Anda nonaktifkan, dipisahkan oleh tanda koma. Misalnya, jika Anda ingin menonaktifkan fungsi exec, system, dan eval, tambahkan baris berikut:

```
disable_functions = exec, system, eval
```

Setelah menyimpan perubahan, Anda perlu merestart Apache agar perubahan ini berlaku:

systemctl restart apache2

Catatan:
Pastikan bahwa Anda memeriksa bahwa aplikasi PHP Anda tetap berfungsi dengan baik setelah mengaktifkan disable_functions, karena beberapa fungsi PHP mungkin diperlukan untuk fungsi normal aplikasi Anda. Selalu uji aplikasi Anda dengan cermat setelah mengubah pengaturan ini.