

Bloqueo de Bad Bots en cPanel con ModSecurity usando el Sistema Vendor

Introducción

Los **bad bots** representan una amenaza significativa para los **servidores web**, realizando miles de solicitudes no autorizadas que pueden comprometer la seguridad y el rendimiento del servidor. Este documento describe la implementación de un sistema automatizado de bloqueo de bad bots en cPanel utilizando **ModSecurity** y el sistema vendor nativo.

A diferencia de las soluciones tradicionales basadas en `.htaccess` que requieren mantenimiento manual y pueden generar inconsistencias entre diferentes servidores, esta solución aprovecha el sistema vendor de cPanel para automatizar completamente la gestión de reglas ModSecurity y actualizaciones de listas de bad bots.

Diferencias con DirectAdmin

Mientras que en DirectAdmin se requiere configuración manual en el directorio

`/usr/local/directadmin/custombuild/custom/modsecurity/conf/`, cPanel ofrece un **sistema vendor integrado** que permite:

- Instalación automática de reglas ModSecurity
- Actualizaciones automáticas de configuraciones
- Gestión centralizada sin intervención manual
- Integración nativa con el sistema de gestión de cPanel

Descripción del Sistema Castris ModSecurity

Instalador one-click

Como root de un sistema con cPanel ejecutar:

```
curl -sSL https://gitlab.castris.com/root/utilidades/-
/raw/main/cpanel/castris_mod_security/install_castris_badbots.sh | bash
Castris Bad Bots ModSecurity Installer
=====

2025-07-05 06:56:34 - Starting Castris Bad Bots installation...
2025-07-05 06:56:34 - cPanel version detected: 11.110.0.68
SUCCESS: ModSecurity is available and loaded
2025-07-05 06:56:34 - SUCCESS: ModSecurity is available and loaded
2025-07-05 06:56:34 - Downloading Castris Bad Bots files from GitLab...
2025-07-05 06:56:34 - Downloading bot_list_management/update_badbot_list.sh...
SUCCESS: Downloaded bot_list_management/update_badbot_list.sh
2025-07-05 06:56:34 - SUCCESS: Downloaded bot_list_management/update_badbot_list.sh
2025-07-05 06:56:34 - Downloading bot_list_management/install_cron.sh...
SUCCESS: Downloaded bot_list_management/install_cron.sh
2025-07-05 06:56:34 - SUCCESS: Downloaded bot_list_management/install_cron.sh
2025-07-05 06:56:34 - Downloading bot_list_management/castris_badbots_list.txt...
SUCCESS: Downloaded bot_list_management/castris_badbots_list.txt
2025-07-05 06:56:34 - SUCCESS: Downloaded bot_list_management/castris_badbots_list.txt
SUCCESS: All files downloaded successfully
2025-07-05 06:56:34 - SUCCESS: All files downloaded successfully
2025-07-05 06:56:34 - Installing Castris Bad Bots cPanel vendor...
info [modsec_vendor] You have added the vendor "Castris".

[castris] Castris
  archive_url | https://gitlab.castris.com/root/utilidades/-
/raw/main/cpanel/castris_mod_security/vendor_package/castris-badbots-v1.0.0.zip
  description | Castris Bad Bots ModSecurity Blocker
  dist_md5 | bcf790fd90f757cd2ad780b76418dba5
  dist_sha512 |
39593fa919723094ff4fe86725f956a334465c1a029bfff2621cc354c83c579c152ce645b3e88841a1d56a0b39e39a
8c3dd814754d03b8aac162e329d4691db3
  distribution | castris-badbots-01
  enabled | 1
  inst_dist | castris-badbots-01
  installed | 1
  installed_from | https://gitlab.castris.com/root/utilidades/-
/raw/main/cpanel/castris_mod_security/vendor_package/meta_castris.yaml
```

```
is_pkg |
meta_vendor_cache_file | /var/cpanel/modsec_vendors/meta_castris.cache
meta_yaml_file | /var/cpanel/modsec_vendors/meta_castris.yaml
name | Castris
path | /etc/apache2/conf.d/modsec_vendor_configs/castris
progress_bar |
report_url |
supported_versions | (3)
vendor_id | castris
vendor_url | https://castris.com
```

SUCCESS: Vendor added successfully

2025-07-05 06:56:35 - SUCCESS: Vendor added successfully

info [modsec_vendor] You have enabled the vendor "castris".

SUCCESS: Vendor enabled successfully

2025-07-05 06:56:35 - SUCCESS: Vendor enabled successfully

2025-07-05 06:56:35 - Installing bot list management system...

2025-07-05 06:56:35 - Starting Castris Bad Bots cron installation...

SUCCESS: Update script found and executable

2025-07-05 06:56:35 - SUCCESS: Update script found and executable

2025-07-05 06:56:35 - Installing weekly cron job for bad bots list update...

SUCCESS: Cron job installed: /etc/cron.d/castris-badbot-update

2025-07-05 06:56:35 - SUCCESS: Cron job installed: /etc/cron.d/castris-badbot-update

2025-07-05 06:56:35 - Restarting cron service...

SUCCESS: Cron service restarted

2025-07-05 06:56:35 - SUCCESS: Cron service restarted

2025-07-05 06:56:35 - Testing cron installation...

WARNING: Cron syntax test failed (this might be normal on some systems)

2025-07-05 06:56:35 - WARNING: Cron syntax test failed (this might be normal on some systems)

SUCCESS: Cron installation test passed

2025-07-05 06:56:35 - SUCCESS: Cron installation test passed

2025-07-05 06:56:35 - Running initial bad bots list update...

2025-07-05 06:56:35 - Starting Castris Bad Bots List update...

2025-07-05 06:56:35 - Downloading new bad bots list from:

https://raw.githubusercontent.com/mitchellkrogza/apache-ultimate-bad-bot-blocker/master/_generator_lists/bad-user-agents-htaccess.list

SUCCESS: Downloaded from primary URL

2025-07-05 06:56:36 - SUCCESS: Downloaded from primary URL

SUCCESS: New bad bots list installed

```
2025-07-05 06:56:36 - SUCCESS: New bad bots list installed
2025-07-05 06:56:36 - New list contains 515 entries
2025-07-05 06:56:36 - Testing Apache configuration...
SUCCESS: Apache configuration test passed
2025-07-05 06:56:36 - SUCCESS: Apache configuration test passed
2025-07-05 06:56:36 - Reloading Apache configuration...
SUCCESS: Apache configuration reloaded
2025-07-05 06:56:38 - SUCCESS: Apache configuration reloaded
2025-07-05 06:56:38 - Statistics:
2025-07-05 06:56:38 - - Previous list: 0 entries
2025-07-05 06:56:38 - - New list: 515 entries
2025-07-05 06:56:38 - - Change: 515 entries
SUCCESS: Bad bots list update completed successfully
2025-07-05 06:56:38 - SUCCESS: Bad bots list update completed successfully
SUCCESS: Initial update completed successfully
2025-07-05 06:56:38 - SUCCESS: Initial update completed successfully
```

```
=====
Castris Bad Bots Cron Installation
=====
```

Installation completed successfully!

Configuration:

- Update script: /usr/local/bin/castris/update_badbot_list.sh
- Cron file: /etc/cron.d/castris-badbot-update
- Log file: /var/log/castris_cron_install.log
- Schedule: Every Sunday at 2:00 AM

Manual commands:

- Run update now: /usr/local/bin/castris/update_badbot_list.sh
- Check cron logs: tail -f /var/log/castris_badbot_update.log
- Remove cron: rm -f /etc/cron.d/castris-badbot-update && systemctl restart cron

The bad bots list will be automatically updated weekly.

Check the logs for update status and statistics.

SUCCESS: Castris Bad Bots cron installation completed

```
2025-07-05 06:56:38 - SUCCESS: Castris Bad Bots cron installation completed
```

SUCCESS: Bot list management system installed

```
2025-07-05 06:56:38 - SUCCESS: Bot list management system installed
```

```
2025-07-05 06:56:38 - Testing installation...
SUCCESS: Apache configuration test passed
2025-07-05 06:56:38 - SUCCESS: Apache configuration test passed
SUCCESS: Vendor installation verified
2025-07-05 06:56:38 - SUCCESS: Vendor installation verified
```

```
=====
Castris Bad Bots Installation Complete!
=====
```

- ☐ cPanel ModSecurity vendor installed
- ☐ Bot list management system installed
- ☐ Apache configuration validated

TESTING:

Test the installation with:

```
curl -H 'User-Agent: BadBot' http://yourserver.com/
(Should return 406 Not Acceptable)
```

MANAGEMENT:

- Update bot list: /usr/local/bin/castris/update_badbot_list.sh
- Check vendor: /scripts/modsec_vendor list
- Disable vendor: /scripts/modsec_vendor disable castris
- Remove vendor: /scripts/modsec_vendor remove castris

LOGS:

- Installation: /var/log/castris_badbots_install.log
- Bot updates: /var/log/castris_badbot_update.log
- ModSecurity: /usr/local/apache/logs/modsec_audit.log

The system will automatically update bad bots lists weekly.

Check cron with: cat /etc/cron.d/castris-badbot-update

SUCCESS: Installation completed successfully!

```
2025-07-05 06:56:38 - SUCCESS: Installation completed successfully!
```

“ Abajo esta la descripción técnica de todo el trabajo, y en mi [Gitlab el resto](#) Es público.

Arquitectura del Paquete

El sistema está compuesto por tres componentes principales:

1. Vendor Package

- `meta_castris.yaml`: Configuración del vendor que define las URLs de descarga y metadatos
- `00_castris_badbots.conf`: Reglas ModSecurity optimizadas con IDs únicos (1090901-1090905)
- `castris-badbots-v1.0.0.zip`: Paquete ZIP que cPanel descarga automáticamente

2. Bot List Management

- `update_badbot_list.sh`: Script de actualización semanal de listas
- `install_cron.sh`: Instalador automatizado del cron
- `castris_badbots_list.txt`: Lista inicial de bad bots conocidos

3. Instalador Automático

- `install_castris_badbots.sh`: Script principal que orquesta toda la instalación
- `pre_install_setup.sh`: Script de verificación de prerequisites (opcional)

Funcionamiento del Sistema Vendor

El sistema vendor de cPanel permite que las reglas ModSecurity se gestionen automáticamente:

1. **Descarga Automática:** cPanel descarga el ZIP desde la URL especificada en `meta_castris.yaml`
2. **Inyección de Reglas:** Las reglas se instalan automáticamente en `/etc/apache2/conf.d/modsec_vendor_configs/castris/`
3. **Actualización Transparente:** Los cambios se aplican sin intervención manual
4. **Persistencia:** Las configuraciones sobreviven a actualizaciones de cPanel

Reglas ModSecurity Implementadas

ID 1090901: Bloqueo principal basado en User-Agent

- Utiliza `@pmFromFile` para comparar contra la lista de bad bots
- Respuesta HTTP 406 (Not Acceptable)
- Logging completo para monitoreo

ID 1090902: Bloqueo de User-Agent vacío

- Detecta solicitudes sin User-Agent

- Protección contra herramientas automatizadas básicas

ID 1090903: Detección de patrones sospechosos

- Identifica comportamientos anómalos en headers
- Análisis de patrones de solicitudes sospechosas

ID 1090904: Rate limiting avanzado

- Límite de 100 solicitudes por hora por IP
- Prevención de ataques de fuerza bruta

ID 1090905: Inicialización de contadores

- Gestión de estado para rate limiting
- Optimización de memoria y rendimiento

Proceso de Instalación

Instalación Automática

El script `install_castris_badbots.sh` realiza las siguientes operaciones:

1. Verificación de Prerrequisitos

- Comprueba que cPanel esté instalado y funcionando
- Verifica que ModSecurity esté habilitado
- Valida permisos de administrador

2. Descarga de Componentes

- Descarga únicamente los archivos necesarios desde GitLab
- Verifica integridad mediante checksums
- Maneja fallos de conectividad con URLs de respaldo

3. Instalación del Vendor

```
/scripts/modsec_vendor add https://gitlab.castris.com/root/utilidades/-  
/raw/main/cpanel/castris_mod_security/vendor_package/meta_castris.yaml  
/scripts/modsec_vendor enable castris
```

4. Configuración de Actualizaciones

- Instala cron para actualizaciones semanales (domingos 2:00 AM)
- Configura logs de actualización en `/var/log/castris_badbot_update.log`
- Establece URLs primarias y de respaldo para listas

5. Validación del Sistema

- Prueba reglas con User-Agents conocidos
- Verifica logs de ModSecurity

- Confirma respuestas HTTP correctas

6. Limpieza

- Elimina archivos temporales de instalación
- Optimiza configuraciones de Apache
- Reinicia servicios solo si es necesario

Gestión de Listas de Bad Bots

El sistema mantiene actualizadas las listas de bad bots mediante:

Fuentes de Datos

- **Primaria:** `https://download.castris.com/badbots/castris_badbots_list.txt`
- **Respaldo:** GitHub Apache Ultimate Bad Bot Blocker

Actualización Automática

- Ejecución semanal vía cron
- Respaldo automático de listas anteriores
- Logging detallado de cambios y estadísticas
- Reinicio automático de Apache solo cuando es necesario

Gestión de Fallos

- Fallback automático a URLs de respaldo
- Conservación de listas anteriores en caso de fallo
- Alertas en logs para problemas de conectividad

Archivos de Configuración

Estructura en el Servidor

```
/etc/apache2/conf.d/modsec_vendor_configs/castris/  
├─ 00_castris_badbots.conf # Reglas ModSecurity  
  
/usr/local/apache/conf/modsec2/  
├─ castris_badbots_list.txt # Lista activa de bad bots  
├─ castris_badbots_list.txt.backup # Respaldo de la lista anterior  
  
/etc/cron.d/  
├─ castris-badbot-update # Cron de actualización semanal
```

```
/var/log/
```

```
|— castris_badbot_update.log
```

```
# Logs de actualizaciones
```

Configuración del Cron

```
# Actualización semanal domingos 2:00 AM  
0 2 * * 0 root /usr/local/bin/castris/update_badbot_list.sh >>  
/var/log/castris_badbot_update.log 2>&1
```

Monitoreo y Validación

Testing del Sistema

```
# Estos comandos DEBEN devolver 406 Not Acceptable  
curl -H 'User-Agent: BadBot' http://yourserver.com/  
curl -H 'User-Agent: nikto' http://yourserver.com/  
curl -H 'User-Agent: wget' http://yourserver.com/  
  
# Este comando DEBE funcionar normalmente (200 OK)  
curl -H 'User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36'  
http://yourserver.com/
```

Análisis de Logs

```
# Ver bloqueos en tiempo real  
tail -f /usr/local/apache/logs/modsec_audit.log | grep 1090901  
  
# Estadísticas de bloqueos  
grep "1090901" /usr/local/apache/logs/modsec_audit.log | wc -l  
  
# Top bots bloqueados  
grep "1090901" /usr/local/apache/logs/modsec_audit.log | \  
grep -o 'User-Agent: [^']*' | sort | uniq -c | sort -nr | head -10
```

Monitoreo de Actualizaciones

```
# Logs de actualización de listas
tail -f /var/log/castris_badbot_update.log

# Verificar última actualización
grep "Statistics:" /var/log/castris_badbot_update.log | tail -5
```

Consideraciones de Seguridad

Ventajas del Sistema

- **Automatización Completa:** Sin intervención manual requerida
- **Actualizaciones Regulares:** Listas actualizadas semanalmente
- **Persistencia:** Configuraciones que sobreviven a actualizaciones del sistema
- **Monitoreo:** Logging detallado para análisis forense
- **Fallback:** Múltiples fuentes de datos para alta disponibilidad

Precauciones

- **Falsos Positivos:** Monitorear que no se bloquee tráfico legítimo
- **Impacto en Rendimiento:** Las reglas son optimizadas pero requieren monitoreo
- **Conectividad:** Dependencia de URLs externas para actualizaciones
- **Logs:** Gestión del crecimiento de archivos de log

Mantenimiento y Solución de Problemas

Verificación de Estado

```
# Verificar vendor habilitado
/scripts/modsec_vendor list

# Verificar ModSecurity cargado
httpd -M | grep security2

# Verificar archivos de configuración
ls -la /etc/apache2/conf.d/modsec_vendor_configs/castris/
```

Solución de Problemas Comunes

Apache no inicia

```
# Verificar sintaxis de configuración
httpd -t

# Revisar logs de error
tail -f /usr/local/apache/logs/error_log
```

Reglas no funcionan

```
# Verificar lista existe
ls -la /usr/local/apache/conf/modsec2/castris_badbots_list.txt

# Verificar permisos
chmod 644 /usr/local/apache/conf/modsec2/castris_badbots_list.txt
```

Actualizaciones fallan

```
# Ejecutar actualización manual con debug
bash -x /usr/local/bin/castris/update_badbot_list.sh

# Verificar conectividad
curl -I https://download.castris.com/badbots/castris_badbots_list.txt
```

Desinstalación Completa

```
# Deshabilitar y remover vendor
/scripts/modsec_vendor disable castris
/scripts/modsec_vendor remove castris

# Eliminar cron
rm -f /etc/cron.d/castris-badbot-update
systemctl restart cron

# Limpiar archivos
rm -rf /usr/local/bin/castris
rm -f /usr/local/apache/conf/modsec2/castris_badbots_list.txt*
```

```
rm -f /var/log/castris_*.log
```

```
# Reiniciar Apache
```

```
systemctl restart httpd
```

Conclusiones

Este sistema representa una evolución significativa en la gestión de bad bots para cPanel, ofreciendo automatización completa y mantenimiento mínimo. La integración con el sistema vendor nativo de cPanel garantiza persistencia y compatibilidad a largo plazo, mientras que las actualizaciones automáticas de listas mantienen la protección actualizada contra nuevas amenazas.

La implementación de reglas ModSecurity optimizadas con IDs únicos evita conflictos con otras configuraciones, y el sistema de monitoreo integral permite análisis detallado del tráfico bloqueado y la efectividad del sistema.

Recursos Adicionales

- **Código fuente:** https://gitlab.castris.com/root/utilidades/-/tree/main/cpanel/castris_mod_security
- **Documentación técnica:** Incluida en el README.md del repositorio
- **Soporte:** <https://castris.com>
- **Lista de bad bots:** Basada en Apache Ultimate Bad Bot Blocker

Disclaimer

Esta herramienta se proporciona tal como está para propósitos de seguridad. Los administradores son responsables de probar y validar la configuración en su entorno antes del despliegue en producción. Se recomienda revisar el código fuente antes de ejecutar scripts de instalación automática.

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Revision #1

Created 2025-07-05 05:16:11 UTC by Abkrim

Updated 2025-07-05 05:38:01 UTC by Abkrim