RELATÓRIO TÉCNICO

Defesa e Monitoramento de Ataques Web com WAF

AMBIENTE SIMULADO - PROJETO CIBERSEGURANÇA

Desenvolvido por:

Aline Blotta A. Jardim

SUMÁRIO

SUMÁRIO EXECUTIVO

A segurança digital é um dos maiores desafios enfrentados pelas empresas atualmente. Com o avanço da tecnologia, os ataques cibernéticos se tornaram mais frequentes, sofisticados e difíceis de detectar.

Um dos principais alvos desses ataques são os sites e sistemas acessados pela internet, que, se tiverem falhas, podem permitir o acesso de invasores, resultando em roubo de dados confidenciais, danos à reputação e até mesmo a paralisação dos serviços.

Este projeto prático demonstra como a ferramenta WAF (Web Application Firewall), pode proteger proativamente as empresas contra essas ameaças. O WAF funciona como uma barreira de proteção, que analisa todo o tráfego que chega e sai do sistema, e quando corretamente configurado, pode impedir ataques e o acesso não autorizado.

Durante os testes realizados neste projeto, foram simulados dois cenários distintos: um em que o WAF apenas monitora e registra tentativas de ataque, e outro em que ele atua de forma ativa, bloqueando qualquer atividade suspeita. Os testes incluíram exemplos reais de ameaças comuns, como tentativas de invasão, manipulação de dados e acesso indevido a arquivos internos.

Este relatório mostra, na prática, que o WAF é uma solução eficaz para detectar e bloquear ameaças digitais, atuando como uma camada essencial de proteção para sistemas expostos à internet. No entanto, sua eficácia é maximizada quando integrado a outras ferramentas de segurança e, fundamentalmente, quando há o envolvimento ativo dos times de desenvolvimento e operações.

Em um cenário cada vez mais desafiador, segurança digital não é responsabilidade de um único setor: é uma construção colaborativa que depende de processos bem definidos, ferramentas eficazes e equipes preparadas. Investir em segurança é investir na continuidade, na reputação e na capacidade da organização de crescer com confiança no ambiente digital.

OBJETIVO

Este documento tem como finalidade apresentar os resultados de uma atividade técnica conduzida em um ambiente controlado, com foco na validação da eficácia de um Web Application Firewall (WAF) configurado com o conjunto de regras OWASP Core Rule Set (CRS).

O objetivo principal foi avaliar a capacidade do WAF em proteger uma aplicação web corporativa exposta contra ataques comuns, como SQL Injection (SQLi), Cross-Site Scripting (XSS) e Local File Inclusion (LFI), por meio de simulações realizadas em ambiente interno. A análise verificou o comportamento do WAF nos modos operacionais de detecção e de bloqueio, bem como sua capacidade de registrar eventos e responder a incidentes de segurança em tempo real.

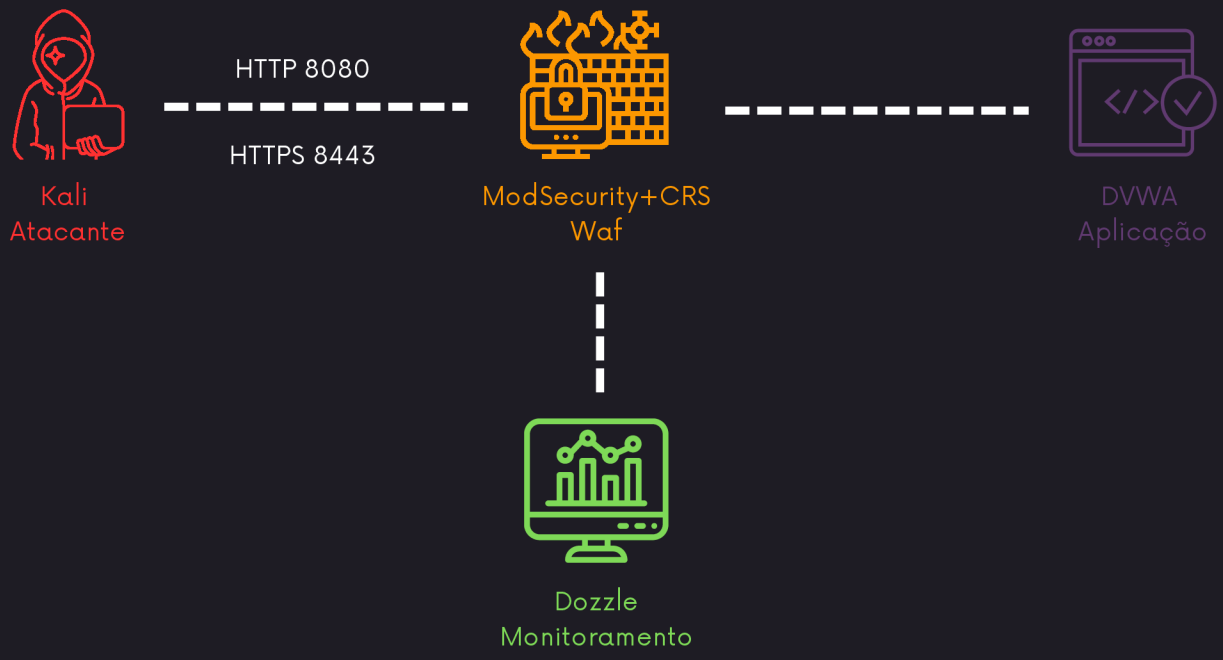
Além disso, a iniciativa teve como propósito fortalecer a postura defensiva do ambiente, promovendo maior visibilidade sobre tentativas de exploração, aprimorando os mecanismos de resposta a incidentes e contribuindo para a maturidade do processo de segurança de aplicações.

ESCOPO

A atividade foi realizada em ambiente isolado, utilizando contêineres Docker para representar os principais componentes de uma arquitetura web corporativa: aplicação vulnerável (DVWA), Firewall de Aplicação Web (ModSecurity), interface de monitoramento (Dozzle) e máquina atacante (Kali Linux).

O escopo da análise concentrou-se na camada de aplicação, com foco na exposição de vulnerabilidades específicas e na resposta do WAF frente a esses vetores.

ARQUITETURA



METODOLOGIA

A avaliação seguiu uma metodologia rigorosa e estruturada, projetada para validar a eficácia do Firewall de Aplicação Web (WAF) em todas as fases de um ciclo de resposta a incidentes.

Passos executados:

Preparação
Inicialização dos containers essenciais: DVWA (Alvo), ModSecurity (WAF), Dozzle (Monitoramento de Logs) e Kali (Atacante).

Detecção
Configuração do WAF em DetectionOnly; envio de payloads de SQLi, XSS e LFI via curl; observação em tempo real dos logs no Dozzle e coleta dos audit logs.

Bloqueio
Alteração do SecRuleEngine para modo ativo On; reenvio dos mesmos payloads; verificação de negação com HTTP 403 e registros de deny nos logs do ModSecurity.

Resposta
Correlação de eventos por txid, timestamp e client IP; extração de matched_rule id e matched_data; coleta de evidências.

Critérios de sucesso

Detecção precisa: todas as tentativas de ataque simuladas (SQLi, XSS e LFI) ativaram corretamente as regras específicas do conjunto OWASP CRS no modo de monitoramento. Além disso, o WAF classificou a atividade como uma ameaça grave, com a Rule ID 949110 (Pontuação de Anomalia) ultrapassando o limite definido.

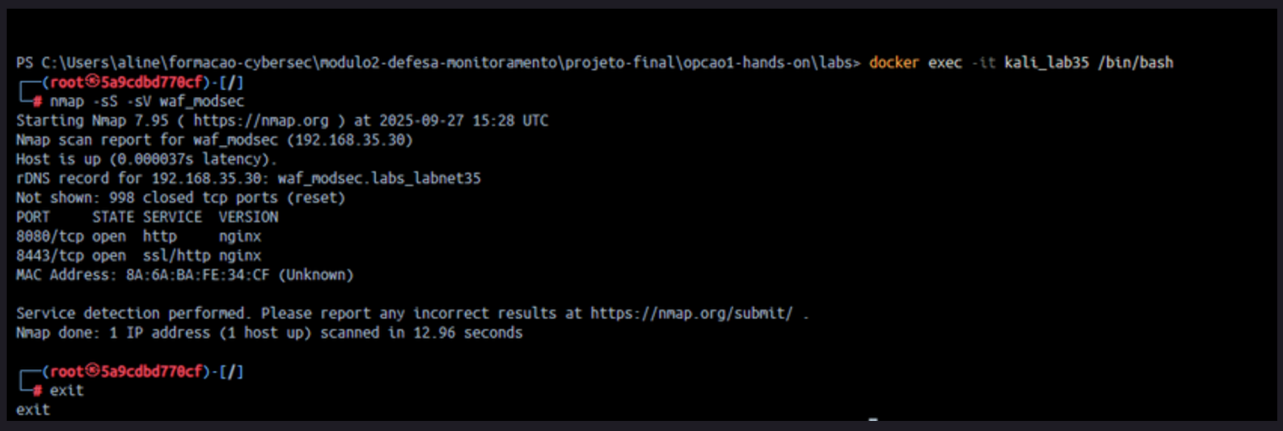
Bloqueio realizado: Após a transição para o modo ativo (On), o WAF negou todas as requisições maliciosas. O atacante foi imediatamente impedido de acessar o alvo, recebendo o código HTTP 403 Forbidden, provando que a ameaça foi neutralizada antes de atingir a aplicação.

Rastreabilidade: Todos os eventos de ataque e bloqueio foram totalmente rastreáveis e correlacionáveis. A utilização de metadados como txid e timestamp garantiu uma linha do tempo precisa, permitindo a extração do payload exato (matched_data) como prova de conceito. Este resultado demonstra uma defesa eficaz e a capacidade técnica para auditar e responder a incidentes com base em evidências irrefutáveis.

EXECUÇÃO E EVIDÊNCIAS

A atividade foi conduzida em ambiente controlado, com o objetivo de validar a eficácia do WAF frente a tentativas reais de exploração da aplicação web. Foram simulados ataques direcionados às vulnerabilidades de SQL Injection (SQLi), Cross-Site Scripting (XSS) e Local File Inclusion (LFI), com monitoramento contínuo e coleta de evidências para análise forense.

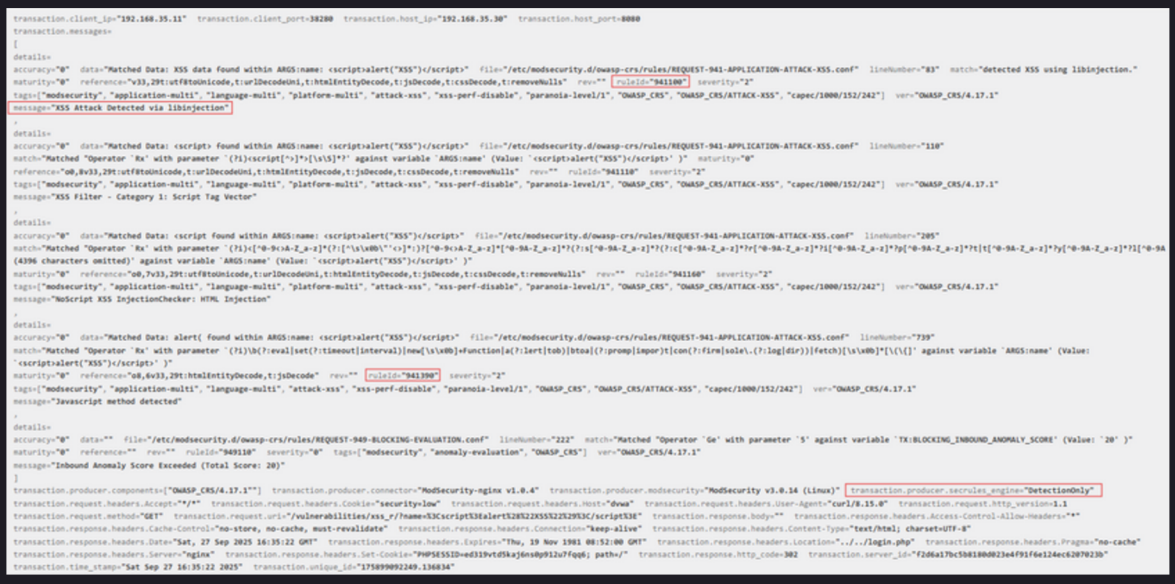
Reconhecimento: Execução de varredura com Nmap a partir do container Kali Linux para identificar portas e serviços expostos. Foram identificadas as portas 8080 (HTTP) e 8443 (HTTPS) sob gerenciamento do WAF, confirmando a exposição da aplicação por meio do ModSecurity.



Testes em modo detecção: Foram executados comandos de SQL Injection (SQLi), Cross-Site Scripting (XSS) e Local File Inclusion (LFI) via requisições HTTP utilizando curl, com o WAF operando em modo “DetectionOnly”. Todas as requisições foram aceitas com status HTTP 302, indicando redirecionamento.

```
┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/sqli/?id=1'+OR+'1'='1'--+-&Submit=Submit" \
  -H "Host: dvwa" \
  -H "Cookie: PHPSESSID=test; security=low" \
  -w "Status: %{http_code}\n"
Status: 302

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
Status: 302

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/fi/?page=../../../../etc/passwd" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
Status: 302
```

O WAF detectou os ataques e registrou os eventos nos logs, com ativação das regras OWASP CRS correspondentes: Rule ID 942100 para SQLi, Rule ID 941100 para XSS e Rule ID 930120 para LFI. Além da identificação individual, foi registrada a ativação da Rule ID 949110, que indica que a pontuação de anomalia (Anomaly Scoring) ultrapassou o limite predefinido. Este achado comprova que o WAF não apenas identificou a ameaça, mas também a classificou como grave, validando sua prontidão para a fase de bloqueio.



Testes em modo bloqueio: Foram reenviados os mesmos payloads de SQL Injection, Cross-Site Scripting e Local File Inclusion via requisições HTTP com curl, com o WAF reconfigurado para o modo de bloqueio; as requisições foram negadas retornando HTTP 403.

Terminal

```
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> docker exec -it kali_lab35 /bin/bash
┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/exec/?ip=127.0.0.1;ls&Submit=Submit" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
</html>
Status: 403

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx</center>
</body>
</html>
Status: 403

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/fi/?page=../../../../etc/passwd" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx</center>
</body>
</html>
Status: 403
```

O bloqueio foi efetivo e os logs exibidos no Dozzle confirmaram a interdição das requisições com ativação das regras OWASP CRS 942100 (SQLi), 941100 (XSS) e 930120 (LFI), registrando em cada entrada a descrição do ataque, modo Blocking, Rule ID, txid, timestamp, client IP, request line e matched_data; a regra 949110 também foi ativada quando a soma das pontuações das detecções ultrapassou o limite predefinido.

Conforme verificado, foi realizado acompanhamento dos eventos via Dozzle durante toda a execução dos testes. Os logs foram capturados com timestamps precisos, permitindo rastreabilidade e análise forense dos eventos. Todas as reconfigurações do ambiente foram documentadas e organizadas.

RESPOSTA A INCIDENTE (NIST IR)

Esta etapa foi construída com base na simulação de um cenário real de ataque à aplicação web. Embora o exercício prático tenha aplicado apenas a transição do WAF para o modo bloqueio, a autora optou por incluir ações complementares, com o objetivo de demonstrar uma resposta estruturada e alinhada às boas práticas de Segurança da Informação.

Detecção: O incidente foi identificado em 27 de setembro de 2025 às 13h40min pela equipe de Segurança da Informação, após alerta da ferramenta de monitoramento Dozzle que exibiu entradas suspeitas originadas do IP 192.168.35.11 correspondentes às regras OWASP CRS para SQL Injection (Rule ID 942100), Cross-Site Scripting (Rule ID 941100) e Local File Inclusion (Rule ID 930120), com ativação adicional da regra agregadora 949110 indicando correlação de eventos pela soma de scores acima do threshold configurado; todas as requisições maliciosas retornaram HTTP 302, confirmando que não houve bloqueio e que o WAF estava em modo DetectionOnly, registrando alertas sem aplicar ações disruptivas.

Contenção: A equipe de resposta ativou imediatamente o modo de bloqueio do WAF (SecRuleEngine On) e aplicou bloqueios temporários por IP, incluindo 192.168.35.11, para interromper o tráfego malicioso; O sistema de monitoramento foi configurado para enviar alertas em tempo real ao time de segurança sempre que as Rule IDs envolvidas fossem acionadas, permitindo acompanhamento ativo durante a contenção.

Com base nos payloads capturados no campo matched_data, a equipe criou regras CRS temporárias que bloqueavam padrões observados e variantes imediatas do ataque.

Essas regras foram projetadas para mitigar rapidamente tráfego semelhante sem depender de mudanças no código da aplicação. A equipe documentou cada regra temporária com justificativa, autor e prazo de expiração para remoção após correção definitiva.

Para validar a contenção, reenviaram-se os payloads originais de forma controlada e confirmou-se a mudança de comportamento das respostas de HTTP 302 para HTTP 403; endpoints não essenciais foram desativados e acessos administrativos expostos foram restringidos por ACLs e listas de IP confiáveis, reduzindo a superfície de ataque durante a remediação. Por fim, abriu-se ticket de incidente com resumo das ações, responsáveis e próximas etapas, comunicou-se o time de Segurança e Operações, e planejou-se a correção definitiva no código da aplicação seguida de revisão e remoção das regras temporárias após validação.

Erradicação: A equipe aplicou correções na aplicação DVWA, implementando sanitização de entradas e consultas parametrizadas para eliminar vetores de injeção; containers comprometidos foram substituídos por imagens limpas e qualquer artefato suspeito detectado nos volumes foi removido; regras permanentes do CRS foram ajustadas com base no matched_data observado e as regras temporárias foram mantidas apenas até validação, quando foram removidas; credenciais administrativas potencialmente expostas foram rotacionadas e privilégios revisados. Em ambiente isolado, foram realizados testes de verificação reenviando os mesmos payloads e executando varreduras com sqlmap e Burp para confirmar que as tentativas que antes tinham sucesso passaram a ser bloqueadas, sem execução indesejada na aplicação; registros e logs foram auditados para garantir ausência de persistência do atacante antes de iniciar a recuperação e restaurar operações normais.

Recuperação: Os serviços foram reimplantados a partir de imagens validadas e configurados com controles reforçados, restaurando as rotinas normais de operação somente após verificação de integridade dos sistemas e confirmação de que os pontos de entrada corrigidos não eram mais exploráveis.

Durante um período de monitoramento intensificado de 72 horas manteve-se nível elevado de telemetria e alertas ativos para as Rule IDs envolvidas e, após confirmação de estabilidade, o nível de logging e a retenção foram ajustados de volta ao padrão; comunicações formais foram enviadas aos stakeholders com resumo das ações, janela de recuperação e recomendações, e um plano de rollback documentado permaneceu disponível para execução imediata caso fosse detectada qualquer regressão durante a janela pós-implantação.

Lições Aprendidas: Ficou claro que o WAF atuou como mitigador imediato, mas não substitui correções no código, portanto é preciso adotar práticas que antecipem e corrijam vulnerabilidades cedo; isso inclui revisão contínua do código para encontrar falhas, executar verificações automáticas e revisões manuais antes de liberar mudanças, e bloquear lançamentos quando forem encontradas falhas críticas.

Planejar verificações regulares de segurança nas aplicações e testes manuais especializados para avaliar a lógica do negócio, transformar cada vulnerabilidade em um chamado rastreável com prioridade e retestar após a correção; padronizar modelos e listas de verificação para novos endpoints e APIs. Envolver os times de desenvolvimento e operações em treinamentos práticos sobre como escrever código mais seguro, revisar dependências e responder a incidentes, além de realizar exercícios e simulações periódicas para reduzir o tempo de detecção e resposta. Atualizar os playbooks para incluir preservação de evidências, retestes controlados e critérios claros para avançar entre fases, padronizar a criação e expiração de regras temporárias no WAF e implantar bloqueios automáticos por pontuação para diminuir trabalho manual e acelerar a mitigação.

PLANO DE AÇÃO 80/20

Ação Estratégica	Impacto	Prioridade	Facilidade
Realizar backup das configurações críticas do WAF e do Servidor Web/Proxy.	Alto	Alta	Alta
Fechar porta 8080 e forçar uso de HTTPS (8443/443)	Alto	Alta	Alta
Configurar o WAF/Proxy para limitar o número de requisições por IP em um curto período.	Alto	Alta	Alto
Adicionar cabeçalhos de segurança no servidor (como X-Frame-Options e Content-Security-Policy)	Alto	Alta	Alto
Treinar equipe de desenvolvimento em práticas seguras	Alto	Alta	Alta
Validação e sanitização de dados de entrada	Alto	Alta	Média
Correção de vulnerabilidades de código (SQLi, XSS, LFI)	Alto	Alta	Média
Implementar senhas fortes e autenticação multifator para proteger contas	Alto	Alta	Média
Controle de acesso baseado em menor privilégio	Alto	Alta	Média
Integrar logs do WAF com sistema SIEM/SOAR	Alto	Média	Média
Realizar tuning das regras OWASP CRS (falsos positivos/negativos)	Médio	Média	Alta

CONCLUSÃO

A análise realizada confirma que o WAF desempenha um papel fundamental na proteção da camada de aplicação, atuando como uma barreira eficaz contra diversos ataques. Sua operação inicial em modo de detecção permitiu visibilidade sobre padrões de ataque e, aliada a uma interface de monitoramento capaz de correlacionar eventos, forneceu subsídios para a tomada de decisão. A transição para o modo de bloqueio demonstrou a capacidade de interromper tentativas de exploração em tempo real, reforçando sua função como mecanismo de contenção imediata.

Entretanto, o WAF representa apenas uma camada de proteção, sendo um componente indispensável de uma estratégia de Defesa em Profundidade. Por atuar na camada de aplicação, sua eficácia depende de otimizações contínuas, como o refinamento de regras (tuning), a gestão de regras temporárias, e a integração com SIEM/SOAR para automação avançada de mitigação baseada em pontuação. Além disso, a segurança requer medidas estruturais complementares, como o fechamento de portas HTTP e a aplicação de cabeçalhos de segurança HTTP no Proxy/Servidor.

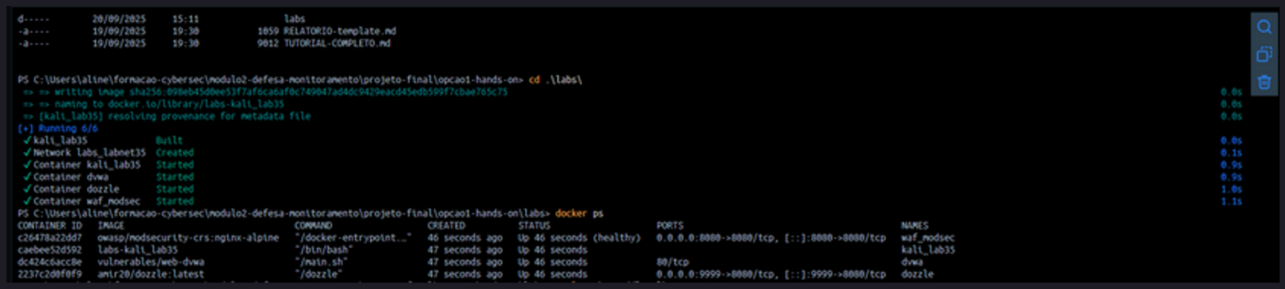
Nesse contexto, destaca-se a importância de incorporar práticas DevSecOps ao ciclo de desenvolvimento, garantindo que a segurança seja considerada desde as fases iniciais do projeto. Isso inclui revisão contínua de código, testes automatizados e manuais, bloqueio de lançamentos em caso de falhas críticas e gestão eficiente de vulnerabilidades, com foco na validação de entradas e correção da causa raiz no código-fonte.

A padronização de processos também contribui para fortalecer a resposta a incidentes. A criação de checklists para novos endpoints e APIs, a atualização de playbooks com critérios claros de transição entre fases e a preservação de evidências são medidas que aumentam a consistência e a resiliência operacional.

Por fim, o engajamento dos times de desenvolvimento e operações em treinamentos práticos e simulações periódicas é essencial para reduzir o tempo de resposta, promover a cultura de segurança e garantir atuação coordenada diante de novos incidentes. A integração entre tecnologia, processos e pessoas é o que consolida uma estratégia de segurança eficaz e sustentável.

EVIDÊNCIAS

```
PS C:\Users\aline> docker --version
Docker version 28.2.2, build e6534b4
PS C:\Users\aline> cd .\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on> dir


    Diretório: C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        20/09/2025     15:11                labs
-a----        19/09/2025     19:30           1059 RELATORIO-template.md
-a----        19/09/2025     19:30           9012 TUTORIAL-COMPLETO.md


PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on> cd .\labs\
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> dir


    Diretório: C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        19/09/2025     19:39                scripts
-a----        20/09/2025     15:33           1487 docker-compose.yml
-a----        19/09/2025     19:30            218 Dockerfile.kali
-a----        20/09/2025     15:34          81208 logs_waf_evidencias.txt
```



```
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> curl http://localhost:8080


StatusCode        : 200
StatusDescription : OK
Content           :
                    <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

                    <html xmlns="http://www.w3.org/1999/xhtml">

                        <head>

                                <meta http-equiv="Content...
RawContent        : HTTP/1.1 200 OK
                    Connection: keep-alive
                    Pragma: no-cache
                    Vary: Accept-Encoding
                    Access-Control-Allow-Headers: *
                    Content-Length: 1523
                    Cache-Control: no-cache, must-revalidate
                    Content-Type: text/ht...
Forms             : {}
Headers           : {[Connection, keep-alive], [Pragma, no-cache], [Vary, Accept-Encoding], [Access-Control-Allow-Headers, *]...}
Images            : {@{innerHTML=; innerText=; outerHTML=<IMG src="dvwa/images/login_logo.png">; outerText=; tagName=IMG; src=dvwa/images/login_logo.png}}
InputFields       : {@{innerHTML=; innerText=; outerHTML=<INPUT class=loginInput name=username>; outerText=; tagName=INPUT; class=loginInput; name=username}, @{innerHTML=; innerText=; outerHTML=<INPUT class=loginInput
                    type=password value="" name=password AUTOCOMPLETE="off">; outerText=; tagName=INPUT; class=loginInput; type=password; value=; name=password; AUTOCOMPLETE=off}, @{innerHTML=; innerText=; outerHTML=<INPUT
                    type=submit value=Login name=Login>; outerText=; tagName=INPUT; type=submit; value=Login; name=Login}, @{innerHTML=; innerText=; outerHTML=<INPUT type=hidden value=97e12f85763c104e412ac68a51d7a864
                    name=user_token>; outerText=; tagName=INPUT; type=hidden; value=97e12f85763c104e412ac68a51d7a864; name=user_token}}
Links             : {@{innerHTML=Damn Vulnerable Web Application (DVWA); innerText=Damn Vulnerable Web Application (DVWA); outerHTML=<A href="http://www.dvwa.co.uk/" target=_blank>Damn Vulnerable Web Application (DVWA)</A>;
                    outerText=Damn Vulnerable Web Application (DVWA); tagName=A; href=http://www.dvwa.co.uk/; target=_blank}}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 1523
```

DVWA

- Setup DVWA
- Instructions
- About

Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in: **/var/www/html/config/config.inc.php**

If the database already exists, **it will be cleared and the data will be reset**.
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

Setup Check

Operating system: ***nix**
Backend database: **MySQL**
PHP version: **7.0.30-0+deb9u1**

Web Server SERVER_NAME: **localhost**

PHP function display_errors: **Disabled**
PHP function safe_mode: Disabled
PHP function allow_url_include: **Disabled**
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed

MySQL username: **app**
MySQL password: ******
MySQL database: **dvwa**
MySQL host: **127.0.0.1**

reCAPTCHA key: **Missing**

[User: www-data] Writable folder /var/www/html/hackable/uploads/: Yes
[User: www-data] Writable file /var/www/html/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: Yes

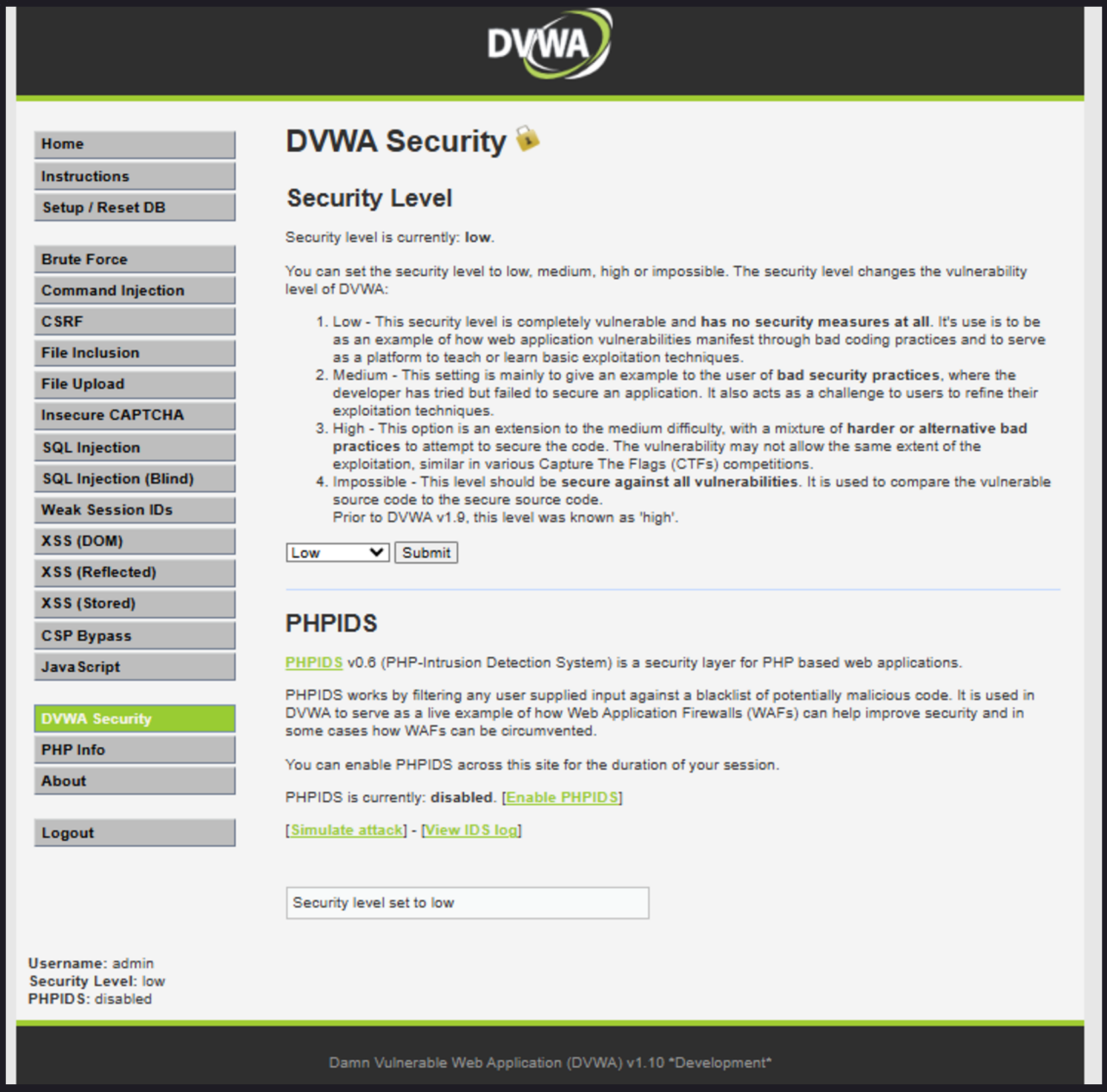
[User: www-data] Writable folder /var/www/html/config: Yes
Status in red, *indicate there will be an issue when trying to complete some modules.*

If you see disabled on either *allow_url_fopen* or *allow_url_include*, set the following in your php.ini file and restart Apache.

```
allow_url_fopen = On
allow_url_include = On
```

These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.

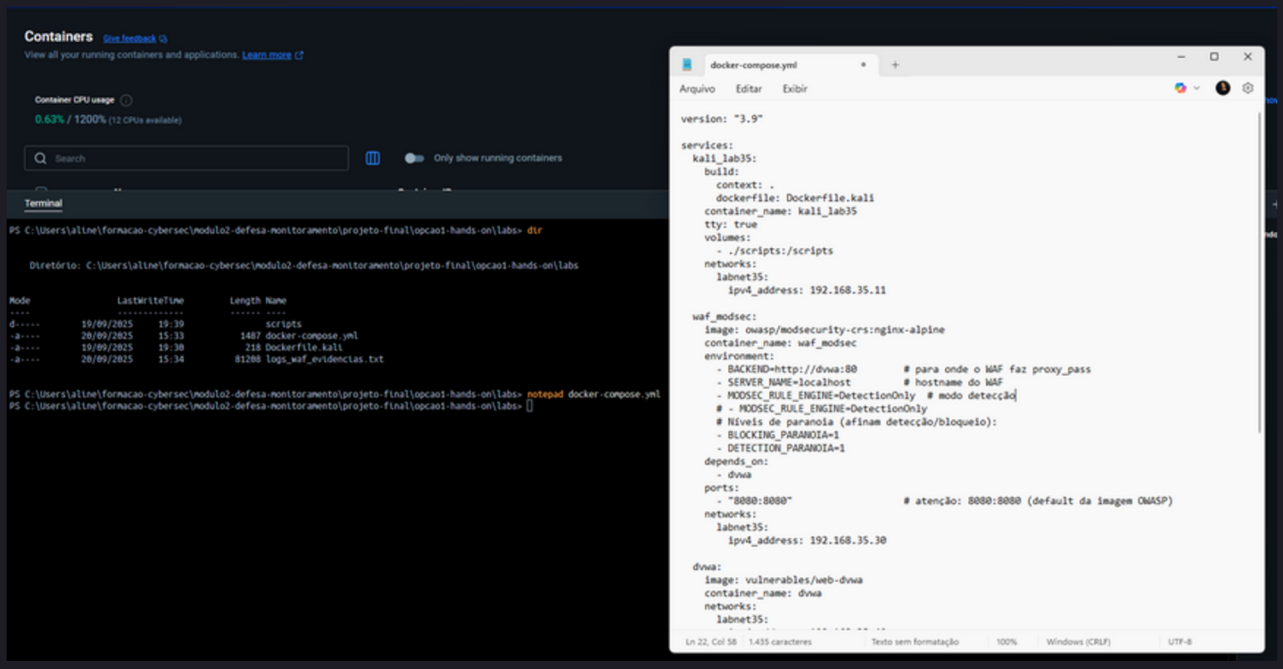
Create / Reset Database

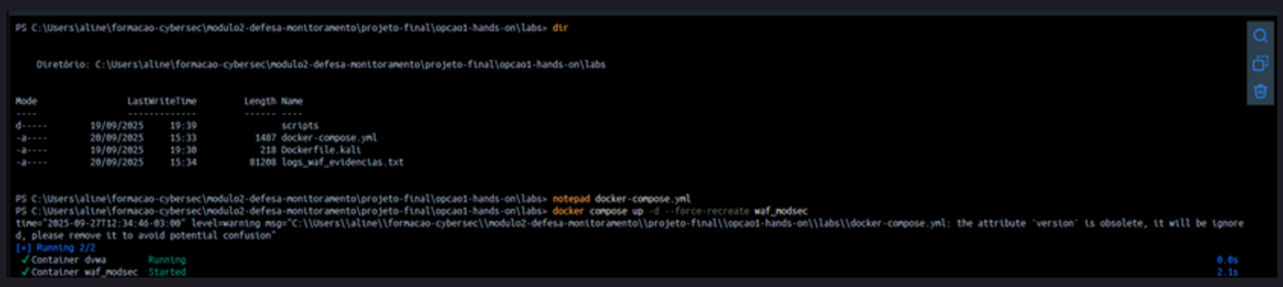



```
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> docker exec -it kali_lab35 /bin/bash
┌──(root㉿5a9cdbd770cf)-[/]
└─# nmap -sS -sV waf_modsec
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-27 15:28 UTC
Nmap scan report for waf_modsec (192.168.35.30)
Host is up (0.000037s latency).
rDNS record for 192.168.35.30: waf_modsec.labs_labnet35
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE  VERSION
8080/tcp open  http     nginx
8443/tcp open  ssl/http nginx
MAC Address: 8A:6A:BA:FE:34:CF (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.96 seconds

┌──(root㉿5a9cdbd770cf)-[/]
└─# exit
exit
```

Terminal

```
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final> cd .\opcao1-hands-on\
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on> dir


    Diretório: C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        19/09/2025     19:39                labs
-a----        19/09/2025     19:30           1059 RELATORIO-template.md
-a----        19/09/2025     19:30           9012 TUTORIAL-COMPLETO.md


PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on> cd .\labs\
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> dir


    Diretório: C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        19/09/2025     19:39                scripts
-a----        19/09/2025     19:30           1509 docker-compose.yml
-a----        19/09/2025     19:30            218 Dockerfile.kali
```

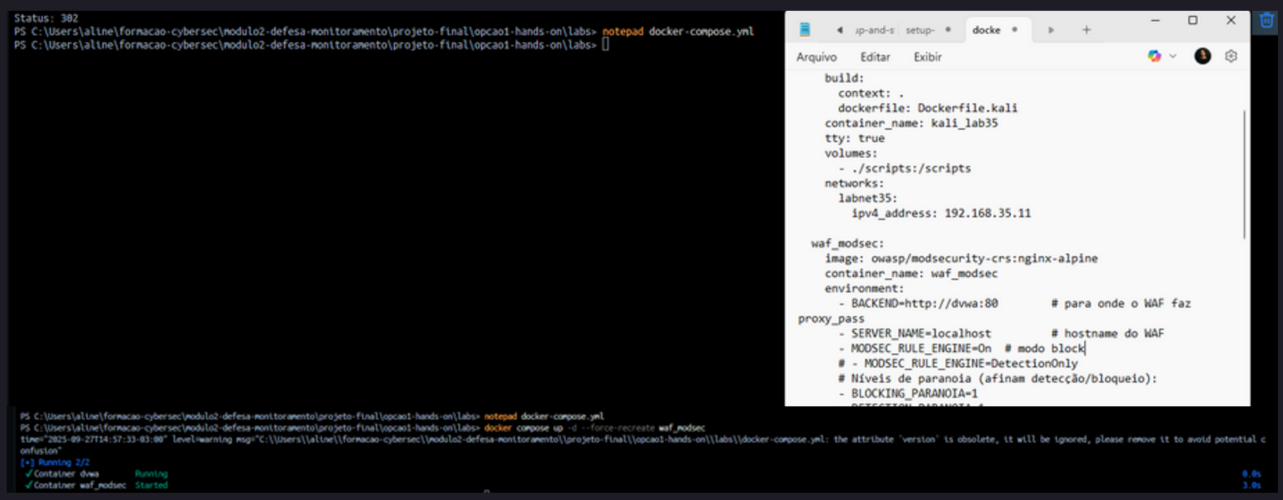




```
┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/sqli/?id=1'+OR+'1'='1'--+-&Submit=Submit" \
  -H "Host: dvwa" \
  -H "Cookie: PHPSESSID=test; security=low" \
  -w "Status: %{http_code}\n"
Status: 302

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
Status: 302

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/fi/?page=../../../../etc/passwd" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
Status: 302
```

```
transaction.client_ip="192.168.35.11"  transaction.client_port=44280  transaction.host_ip="192.168.35.30"  transaction.host_port=8080
transaction.messages=
[
details=
accuracy="0"  data="Matched Data: s&sos found within ARGS:id: 1' OR '1'='1'-- -"  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf"  lineNumber="46"  match="detected SQLi using libinjection."  maturity="0"
reference="v30,17"  rev=""  ruleId="942100"  severity="2"  tags=["application-multi", "language-multi", "platform-multi", "attack-sqli", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-SQLI", "capec/1000/152/248/66"]  ver="OWASP_CRS/4.17.1"
message="SQL Injection Attack Detected via libinjection"
,
details=
accuracy="0"  data=""  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-949-BLOCKING-EVALUATION.conf"  lineNumber="222"  match="Matched "Operator `Ge' with parameter `5' against variable `TX:BLOCKING_INBOUND_ANOMALY_SCORE' (Value: `5' )"
maturity="0"  reference=""  rev=""  ruleId="949110"  severity="0"  tags=["modsecurity", "anomaly-evaluation", "OWASP_CRS"]  ver="OWASP_CRS/4.17.1"
message="Inbound Anomaly Score Exceeded (Total Score: 5)"
]
transaction.producer.components=["OWASP_CRS/4.17.1""]  transaction.producer.connector="ModSecurity-nginx v1.0.4"  transaction.producer.modsecurity="ModSecurity v3.0.14 (Linux)"  transaction.producer.secrules_engine="DetectionOnly"
transaction.request.headers.Accept="*/*"  transaction.request.headers.Cookie="PHPSESSID=test; security=low"  transaction.request.headers.Host="dvwa"  transaction.request.headers.User-Agent="curl/8.15.0"  transaction.request.http_version=1.1
transaction.request.method="GET"  transaction.request.uri="/vulnerabilities/sqli/?id=1'+OR+'1'='1'--+-&Submit=Submit"  transaction.response.body=""  transaction.response.headers.Access-Control-Allow-Headers="*"
transaction.response.headers.Cache-Control="no-store, no-cache, must-revalidate"  transaction.response.headers.Connection="keep-alive"  transaction.response.headers.Content-Type="text/html; charset=UTF-8"
transaction.response.headers.Date="Sat, 27 Sep 2025 16:34:38 GMT"  transaction.response.headers.Expires="Thu, 19 Nov 1981 08:52:00 GMT"  transaction.response.headers.Location="../../login.php"  transaction.response.headers.Pragma="no-cache"
transaction.response.headers.Server="nginx"  transaction.response.http_code=302  transaction.server_id="f2d6a17bc5b8180d023e4f91f6e124ec6207023b"  transaction.time_stamp="Sat Sep 27 16:34:37 2025"  transaction.unique_id="1758990087744.910425"
```

```
transaction.client_ip="192.168.35.11"  transaction.client_port=38280  transaction.host_ip="192.168.35.30"  transaction.host_port=8080
transaction.messages=
[
details=
accuracy="0"  data="Matched Data: XSS data found within ARGS:name: <script>alert("XSS")</script>"  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf"  lineNumber="83"  match="detected XSS using libinjection."
maturity="0"  reference="v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls"  rev=""  ruleId="941100"  severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"]  ver="OWASP_CRS/4.17.1"
message="XSS Attack Detected via libinjection"
,
details=
accuracy="0"  data="Matched Data: <script> found within ARGS:name: <script>alert("XSS")</script>"  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf"  lineNumber="110"
match="Matched "Operator `Rx' with parameter `(?i)<script[^>]*>[\s\S]*?' against variable `ARGS:name' (Value: `<script>alert("XSS")</script>' )"  maturity="0"
reference="o0,8v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls"  rev=""  ruleId="941110"  severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"]  ver="OWASP_CRS/4.17.1"
message="XSS Filter - Category 1: Script Tag Vector"
,
details=
accuracy="0"  data="Matched Data: <script found within ARGS:name: <script>alert("XSS")</script>"  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf"  lineNumber="205"
match="Matched "Operator `Rx' with parameter `(?i)<[^0-9<>A-Z_a-z]*(?:[^\s\x0b\"'<>]*:)?[^0-9<>A-Z_a-z]*[^0-9A-Z_a-z]*(?:s[^0-9A-Z_a-z]*(?:c[^0-9A-Z_a-z]*(?:r[^0-9A-Z_a-z]*(?:i[^0-9A-Z_a-z]*(?:p[^0-9A-Z_a-z]*(?:t|t[^0-9A-Z_a-z]*(?:y[^0-9A-Z_a-z]*(?:l[^0-9A
(4396 characters omitted)' against variable `ARGS:name' (Value: `<script>alert("XSS")</script>' )"
maturity="0"  reference="o0,7v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls"  rev=""  ruleId="941160"  severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"]  ver="OWASP_CRS/4.17.1"
message="NoScript XSS InjectionChecker: HTML Injection"
,
details=
accuracy="0"  data="Matched Data: alert( found within ARGS:name: <script>alert("XSS")</script>"  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf"  lineNumber="739"
match="Matched "Operator `Rx' with parameter `(?i)\b(?:eval|set(?:timeout|interval)|new[\s\x0b]+function|a(?:lert|tob)|btoa|(?:prompt|impor)t|con(?:firm|sole\.(?:log|dir))|fetch)[\s\x0b]*[\(\{]' against variable `ARGS:name' (Value:
`<script>alert("XSS")</script>' )"
maturity="0"  reference="o0,6v33,29t:htmlEntityDecode,t:jsDecode"  rev=""  ruleId="941390"  severity="2"
tags=["modsecurity", "application-multi", "language-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"]  ver="OWASP_CRS/4.17.1"
message="Javascript method detected"
,
details=
accuracy="0"  data=""  file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-949-BLOCKING-EVALUATION.conf"  lineNumber="222"  match="Matched "Operator `Ge' with parameter `5' against variable `TX:BLOCKING_INBOUND_ANOMALY_SCORE' (Value: `20' )"
maturity="0"  reference=""  rev=""  ruleId="949110"  severity="0"  tags=["modsecurity", "anomaly-evaluation", "OWASP_CRS"]  ver="OWASP_CRS/4.17.1"
message="Inbound Anomaly Score Exceeded (Total Score: 20)"
]
transaction.producer.components=["OWASP_CRS/4.17.1""]  transaction.producer.connector="ModSecurity-nginx v1.0.4"  transaction.producer.modsecurity="ModSecurity v3.0.14 (Linux)"  transaction.producer.secrules_engine="DetectionOnly"
transaction.request.headers.Accept="*/*"  transaction.request.headers.Cookie="security=low"  transaction.request.headers.Host="dvwa"  transaction.request.headers.User-Agent="curl/8.15.0"  transaction.request.http_version=1.1
transaction.request.method="GET"  transaction.request.uri="/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E"  transaction.response.body=""  transaction.response.headers.Access-Control-Allow-Headers="*"
transaction.response.headers.Cache-Control="no-store, no-cache, must-revalidate"  transaction.response.headers.Connection="keep-alive"  transaction.response.headers.Content-Type="text/html; charset=UTF-8"
transaction.response.headers.Date="Sat, 27 Sep 2025 16:35:22 GMT"  transaction.response.headers.Expires="Thu, 19 Nov 1981 08:52:00 GMT"  transaction.response.headers.Location="../../login.php"  transaction.response.headers.Pragma="no-cache"
transaction.response.headers.Server="nginx"  transaction.response.headers.Set-Cookie="PHPSESSID=ed319vtd5kaj6ns0p912u7fqq6; path=/"  transaction.response.http_code=302  transaction.server_id="f2d6a17bc5b8180d023e4f91f6e124ec6207023b"
transaction.time_stamp="Sat Sep 27 16:35:22 2025"  transaction.unique_id="1758990922249.136834"
```




```
PS C:\Users\aline\formacao-cybersec\modulo2-defesa-monitoramento\projeto-final\opcao1-hands-on\labs> docker exec -it kali_lab35 /bin/bash
┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/exec/?ip=127.0.0.1;ls&Submit=Submit" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
</html>
Status: 403

┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx</center>
</body>
</html>
Status: 403
```

```
┌──(root㉿5a9cdbd770cf)-[/]
└─# curl -s "http://waf_modsec:8080/vulnerabilities/fi/?page=../../../../etc/passwd" \
  -H "Host: dvwa" \
  -H "Cookie: security=low" \
  -w "Status: %{http_code}\n"
<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx</center>
</body>
</html>
Status: 403
```


transaction.client_ip="192.168.35.11" transaction.client_port=46612 transaction.host_ip="192.168.35.30" transaction.host_port=8080
transaction.messages=
[
details=
accuracy="0" data="Matched Data: s&sos found within ARGS:id: 1' OR '1'='1'-- -" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf" lineNumber="46" match="detected SQLi using libinjection." maturity="0"
reference="v30,17" rev="" ruleId="942100" severity="2" tags=["application-multi", "language-multi", "platform-multi", "attack-sqli", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-SQLI", "capec/1000/152/248/66"] ver="OWASP_CRS/4.17.1"
message="SQL Injection Attack Detected via libinjection"
,
details=
accuracy="0" data="" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-949-BLOCKING-EVALUATION.conf" lineNumber="222" match="Matched "Operator `Ge' with parameter `5' against variable `TX:BLOCKING_INBOUND_ANOMALY_SCORE' (Value: `5')"
maturity="0" reference="" rev="" ruleId="949110" severity="0" tags=["modsecurity", "anomaly-evaluation", "OWASP_CRS"] ver="OWASP_CRS/4.17.1"
message="Inbound Anomaly Score Exceeded (Total Score: 5)"
]
transaction.producer.components=["OWASP_CRS/4.17.1""] transaction.producer.connector="ModSecurity-nginx v1.0.4" transaction.producer.modsecurity="ModSecurity v3.0.14 (Linux)" transaction.producer.secrules_engine="Enabled"
transaction.request.headers.Accept="*/*" transaction.request.headers.Cookie="PHPSESSID=test; security=low" transaction.request.headers.Host="dvwa" transaction.request.headers.User-Agent="curl/8.15.0" transaction.request.http_version=1.1
transaction.request.method="GET" transaction.request.uri="/vulnerabilities/sqli/?id=1'+OR+'1'+'1'---+-&Submit=Submit"
transaction.response.body="<html> <head><title>403 Forbidden</title></head> <body> <h1>403 Forbidden</h1> <hr> nginx </body> </html> " transaction.response.headers.Access-Control-Allow-Headers="*"
transaction.response.headers.Access-Control-Allow-Methods="GET, POST, PUT, DELETE, OPTIONS" transaction.response.headers.Access-Control-Allow-Origin="*" transaction.response.headers.Access-Control-Max-Age="3600"
transaction.response.headers.Connection="keep-alive" transaction.response.headers.Content-Length="146" transaction.response.headers.Content-Type="text/plain" transaction.response.headers.Date="Sat, 27 Sep 2025 18:25:11 GMT"
transaction.response.headers.Server="nginx" transaction.response.http_code=403 transaction.server_id="cd463f31af8011abbd098576c95fedf33dbee3c1" transaction.time_stamp="Sat Sep 27 18:25:11 2025" transaction.unique_id="175899751156.152407"

transaction.client_ip="192.168.35.11" transaction.client_port=43996 transaction.host_ip="192.168.35.30" transaction.host_port=8080
transaction.messages=
[
details=
accuracy="0" data="Matched Data: XSS data found within ARGS:name: <script>alert("XSS")</script>" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf" lineNumber="83" match="detected XSS using libinjection."
maturity="0" reference="v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls" rev="" ruleId="941100" severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"] ver="OWASP_CRS/4.17.1"
message="XSS Attack Detected via libinjection"
,
details=
accuracy="0" data="Matched Data: <script> found within ARGS:name: <script>alert("XSS")</script>" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf" lineNumber="110"
match="Matched "Operator `Rx' with parameter `(?i)<script[^>]*>[\s\S]*?' against variable `ARGS:name' (Value: `<script>alert("XSS")</script>')" maturity="0"
reference="o0,8v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls" rev="" ruleId="941110" severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"] ver="OWASP_CRS/4.17.1"
message="XSS Filter - Category 1: Script Tag Vector"
,
details=
accuracy="0" data="Matched Data: <script found within ARGS:name: <script>alert("XSS")</script>" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf" lineNumber="205"
match="Matched "Operator `Rx' with parameter `(?i)<[^0-9<>A-Z_a-z]*(?:[^\s\x0b\"'<>]*:)?[^0-9<>A-Z_a-z]*[^0-9A-Z_a-z]*?(?:s[^0-9A-Z_a-z]*?(?:c[^0-9A-Z_a-z]*?r[^0-9A-Z_a-z]*?i[^0-9A-Z_a-z]*?p[^0-9A-Z_a-z]*?t|t[^0-9A-Z_a-z]*?y[^0-9A-Z_a-z]*?l[^0-9A
(4396 characters omitted)' against variable `ARGS:name' (Value: `<script>alert("XSS")</script>')"
maturity="0" reference="o0,7v33,29t:utf8toUnicode,t:urlDecodeUni,t:htmlEntityDecode,t:jsDecode,t:cssDecode,t:removeNulls" rev="" ruleId="941160" severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"] ver="OWASP_CRS/4.17.1"
message="NoScript XSS InjectionChecker: HTML Injection"
,
details=
accuracy="0" data="Matched Data: alert(found within ARGS:name: <script>alert("XSS")</script>" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-941-APPLICATION-ATTACK-XSS.conf" lineNumber="739"
match="Matched "Operator `Rx' with parameter `(?i)\b(?:eval|set(?:timeout|interval)|new[\s\x0b]+Function|a(?:lert|tob)|btoa|(?:promp|impor)t|con(?:firm|sole\.(?:log|dir))|fetch)[\s\x0b]*[\(\{]' against variable `ARGS:name' (Value:
`<script>alert("XSS")</script>')"
maturity="0" reference="o8,6v33,29t:htmlEntityDecode,t:jsDecode" rev="" ruleId="941390" severity="2"
tags=["modsecurity", "application-multi", "language-multi", "attack-xss", "xss-perf-disable", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-XSS", "capec/1000/152/242"] ver="OWASP_CRS/4.17.1"
message="Javascript method detected"
,
details=
accuracy="0" data="" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-949-BLOCKING-EVALUATION.conf" lineNumber="222" match="Matched "Operator `Ge' with parameter `5' against variable `TX:BLOCKING_INBOUND_ANOMALY_SCORE' (Value: `20')"
maturity="0" reference="" rev="" ruleId="949110" severity="0" tags=["modsecurity", "anomaly-evaluation", "OWASP_CRS"] ver="OWASP_CRS/4.17.1"
message="Inbound Anomaly Score Exceeded (Total Score: 20)"
]
transaction.producer.components=["OWASP_CRS/4.17.1""] transaction.producer.connector="ModSecurity-nginx v1.0.4" transaction.producer.modsecurity="ModSecurity v3.0.14 (Linux)" transaction.producer.secrules_engine="Enabled"
transaction.request.headers.Accept="*/*" transaction.request.headers.Cookie="security=low" transaction.request.headers.Host="dvwa" transaction.request.headers.User-Agent="curl/8.15.0" transaction.request.http_version=1.1
transaction.request.method="GET" transaction.request.uri="/vulnerabilities/xss_r/?name=%3Cscript%3Ealert%28%22XSS%22%29%3C/script%3E"
transaction.response.body="<html> <head><title>403 Forbidden</title></head> <body> <h1>403 Forbidden</h1> <hr> nginx </body> </html> " transaction.response.headers.Access-Control-Allow-Headers="*"
transaction.response.headers.Access-Control-Allow-Methods="GET, POST, PUT, DELETE, OPTIONS" transaction.response.headers.Access-Control-Allow-Origin="*" transaction.response.headers.Access-Control-Max-Age="3600"
transaction.response.headers.Connection="keep-alive" transaction.response.headers.Content-Length="146" transaction.response.headers.Content-Type="text/plain" transaction.response.headers.Date="Sat, 27 Sep 2025 18:25:25 GMT"
transaction.response.headers.Server="nginx" transaction.response.http_code=403 transaction.server_id="cd463f31af8011abbd098576c95fedf33dbee3c1" transaction.time_stamp="Sat Sep 27 18:25:25 2025" transaction.unique_id="175899752585.782573"


transaction.client_ip="192.168.35.11" transaction.client_port=50066 transaction.host_ip="192.168.35.30" transaction.host_port=8080
transaction.messages=
[
details=
accuracy="0" data="Matched Data: /../ found within ARGS:page: ../../../../etc/passwd" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-930-APPLICATION-ATTACK-LFI.conf" lineNumber="35"
match="Matched "Operator `Rx' with parameter `(?i)(?:[/\x5c]|%(?:2(?:f|5(?:2f|5c|c(?:1%259c|0%25af))|%46)|5c|c(?:0%(?:[2aq]f|5c|9v)|1%(?:[19p]c|8s|af)))(?:bg%q|(?:e|f(?:8%8)?0%8)0%80%af)f|u(?:221[56]|EFC8|F025|002f)|%3(?:2(?:%(?:%6|4)6|F)|5%%63))1 (394 characters omitted)' against variable `ARGS:page' (Value: `../../../../etc/passwd')"
maturity="0" reference="o28,4v4,48o2,4v30,22" rev="" ruleId="930100" severity="2"
tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-lfi", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-LFI", "capec/1000/255/153/126"] ver="OWASP_CRS/4.17.1"
message="Path Traversal Attack (/../) or (/.../)"
,
details=
accuracy="0" data="Matched Data: /../ found within REQUEST_URI_RAW: /vulnerabilities/fi/?page=../../../../etc/passwd" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-930-APPLICATION-ATTACK-LFI.conf" lineNumber="68"
match="Matched "Operator `Rx' with parameter `(?:(?:^|[\x5c/;])\.{2,3}[\x5c/;]|[\x5c/;]\.{2,3}[\x5c/;])' against variable `REQUEST_URI_RAW' (Value: `/vulnerabilities/fi/?page=../../../../etc/passwd')" maturity="0" reference="o28,4v4,48" rev=""
ruleId="930110" severity="2" tags=["application-multi", "language-multi", "platform-multi", "attack-lfi", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-LFI", "capec/1000/255/153/126"] ver="OWASP_CRS/4.17.1"
message="Path Traversal Attack (/../) or (/.../)"
,
details=
accuracy="0" data="Matched Data: ../ found within ARGS:page: ../../../../etc/passwd" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-930-APPLICATION-ATTACK-LFI.conf" lineNumber="68"
match="Matched "Operator `Rx' with parameter `(?:(?:^|[\x5c/;])\.{2,3}[\x5c/;]|[\x5c/;]\.{2,3}[\x5c/;])' against variable `ARGS:page' (Value: `../../../../etc/passwd')" maturity="0" reference="o0,3v30,22" rev="" ruleId="930110" severity="2"
tags=["application-multi", "language-multi", "platform-multi", "attack-lfi", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-LFI", "capec/1000/255/153/126"] ver="OWASP_CRS/4.17.1"
message="Path Traversal Attack (/../) or (/.../)"
,
details=
accuracy="0" data="Matched Data: etc/passwd found within ARGS:page: ../../../../etc/passwd" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-930-APPLICATION-ATTACK-LFI.conf" lineNumber="99"
match="Matched "Operator `PmFromFile' with parameter `lfi-os-files.data' against variable `ARGS:page' (Value: `../../../../etc/passwd')" maturity="0" reference="o12,10v30,22t:utf8toUnicode,t:urlDecodeUni,t:normalizePathWin" rev=""
ruleId="930120" severity="2" tags=["modsecurity", "application-multi", "language-multi", "platform-multi", "attack-lfi", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-LFI", "capec/1000/255/153/126"] ver="OWASP_CRS/4.17.1"
message="OS File Access Attempt"
,
details=
accuracy="0" data="Matched Data: etc/passwd found within ARGS:page: ../../../../etc/passwd" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-932-APPLICATION-ATTACK-RCE.conf" lineNumber="631"
match="Matched "Operator `PmFromFile' with parameter `unix-shell.data' against variable `ARGS:page' (Value: `../../../../etc/passwd')" maturity="0" reference="o12,10v30,22t:cmdLine,t:normalizePath" rev="" ruleId="932160" severity="2"
tags=["modsecurity", "application-multi", "language-shell", "platform-unix", "attack-rce", "paranoia-level/1", "OWASP_CRS", "OWASP_CRS/ATTACK-RCE", "capec/1000/152/248/88"] ver="OWASP_CRS/4.17.1"
message="Remote Command Execution: Unix Shell Code Found"
,
details=
accuracy="0" data="" file="/etc/modsecurity.d/owasp-crs/rules/REQUEST-949-BLOCKING-EVALUATION.conf" lineNumber="222" match="Matched "Operator `Ge' with parameter `5' against variable `TX:BLOCKING_INBOUND_ANOMALY_SCORE' (Value: `30')"
maturity="0" reference="" rev="" ruleId="949110" severity="0" tags=["modsecurity", "anomaly-evaluation", "OWASP_CRS"] ver="OWASP_CRS/4.17.1"
message="Inbound Anomaly Score Exceeded (Total Score: 30)"
]
transaction.producer.components=["OWASP_CRS/4.17.1""] transaction.producer.connector="ModSecurity-nginx v1.0.4" transaction.producer.modsecurity="ModSecurity v3.0.14 (Linux)" transaction.producer.secrules_engine="Enabled"
transaction.request.headers.Accept="*/*" transaction.request.headers.Cookie="security=low" transaction.request.headers.Host="dvwa" transaction.request.headers.User-Agent="curl/8.15.0" transaction.request.http_version=1.1
transaction.request.method="GET" transaction.request.uri="/vulnerabilities/fi/?page=../../../../etc/passwd"
transaction.response.body="<html> <head><title>403 Forbidden</title></head> <body> <h1>403 Forbidden</h1> <hr> nginx </body> </html> " transaction.response.headers.Access-Control-Allow-Headers="*"
transaction.response.headers.Access-Control-Allow-Methods="GET, POST, PUT, DELETE, OPTIONS" transaction.response.headers.Access-Control-Allow-Origin="*" transaction.response.headers.Access-Control-Max-Age="3600"
transaction.response.headers.Connection="keep-alive" transaction.response.headers.Content-Length="146" transaction.response.headers.Content-Type="text/plain" transaction.response.headers.Date="Sat, 27 Sep 2025 18:25:36 GMT"
transaction.response.headers.Server="nginx" transaction.response.http_code=403 transaction.server_id="cd463f31af8011abbd09857dc95fe0f33dbee3c1" transaction.time_stamp="Sat Sep 27 18:25:36 2025" transaction.unique_id="1758999753634.380878"