Hochschule
für Technik
Stuttgart

Erarbeitung und Evaluation einer Laborumgebung zum Thema Webapplication Firewall

Bachelor Thesis
im Studiengang Informatik

Betreuender Dozent:	Prof. Dr. Jan Seedorf
Betreuer des Unternehmens:	Oliver Paukstadt
Vorgelegt am:	14. Februar 2024
Vorgelegt von:	Lukas Reinke (Mat. NR. 1001213)

1 Abstract

Hier könnte ihr abstract stehen.

Inhaltsverzeichnis

2 Abkürzungsverzeichnis

WAF Web Application Firewall

WAF & IAM Web Application Firewall & Identity und Access Management

IT-SiG IT-Sicherheitsgesetzes

DSGVO Datenschutz-Grundverordnung

HTTP Hypertext Transfer Protokoll

3 Abbildungsverzeichnis

4 Einleitung

4.1 Einleitung

Das Feld der Web Application Firewall (WAF) gewinnt in den aktuell immer mehr an Relevanz. Dem Markt wird in den nächsten fünf Jahren ein jährliches Wachstum um 19,9 % auf 14,6 Mrd. vorhergesagt **WebApplicationFirewall**. Auch Ausarbeitungen zum *Stand der Technik* wie sie zum Beispiel im Deutschen IT-Sicherheitsgesetzes (IT-SiG) und der Datenschutz-Grundverordnung (DSGVO) gefordert werden, beschreiben eine WAF als notwendig zur absicherung einer Webanwendung[1, 3.1.19 Schutz von Webanwendungen].

Diese Bachelor Thesis beschäftigt sich mit der Erarbeitung von Lerneinheiten und einer Laborumgebung anhand derer das Thema WAF vermittelt werden kann. Es werden sowohl Design und Implementierung einer WAF vermittelt. Jedoch sollen auch Themen vermittelt werden, die ein Consultant dessen Aufgabe die Betreuung einer WAF ist in seinem Berufsalltag benötigt. Dazu zählen zum Beispiel deployment und Anpassung einer WAF an die zu schützende Web-Anwendung.

4.2 Umfang und Abgrenzung

5 Theoretische Grundlagen

5.1 NAT und Reverse Proxy

5.2 HTTP

Das Hypertext Transfer Protokoll (HTTP) ist ein Protokoll in der Internet-Kommunikation, dass zur übertagung diverser Daten unterschiedlicher Datentypen genutzt werden kann. Sein Haupt-Einsatzgebiet ist die Datenübertragung zwischen Webseiten und Clients. Seit der Einführung in 1991 wurde es in mehreren RFCs erweitert und ist inzwischen in version drei.

In seiner aktuellen Form kann es Gebrauch von TCP-Sitzungen machen um Verbindungen über längere Zeit aufrecht zu halten und fortgeschrittenere Kommunikation, wie push Nachrichten, zu erlauben. Außerdem kann TCP-Pipelining genutzt werden um die parallele Abarbeitung von Anfragen zu ermöglichen und nicht auf die *Acknowledge*-Nachrichten von TCP warten zu müssen. Um seine Grundfunktion zu erläutern wird in diesem Kapitel die statuslose Kommunikation, definiert in Version 1.0 die mit einem unmittelbaren Anfrage-Antwort Muster arbeitet, beschrieben. Hierin initiiert ein Client mit einer Anfrage Nachricht [1] eine Verbindung, die von einem Server verarbeitet und mit einer Antwort-Nachricht beantwortet wird. Darauf wird die Verbindung geschlossen. Dem Sever ist es nun nicht mehr möglich dem Client weitere Daten zu senden, ohne das der Client eine weitere Anfrage-Nachricht schickt.

HTTP-Nachrichten Die Grundlegende Einheit einer Hypertext Transfer Protokoll (HTTP)-Kommunikation wird als *Nachricht* bezeichnet. Da HTTP ein Klartext-Protokoll ist, werden diese in menschenlesbarer Form als Text übertragen. Eine Nachricht besteht aus einer *Start-Zeile*, die die Nachricht entweder als Anfrage oder Antwort identifiziert. In diesen beiden Fällen hat die Zeile jeweils einen untersiedlichen Aufbau:

Request-Zeile: Ein HTTP-Request ist durch eine *Request-Zeile* identifiziert. Diese ist in drei Teile Aufgeteilt.

Die HTTP-Methode: beschreibt die

Status-Zeile:

[1] Im folgenden werden HTTP-Anfrage und das Englische HTTP-request synonym verwendet

5.3 OWASP Top-Ten

5.4 Web Application Firewall

Mit dem Namen WAF wird eine Sicherheits-Anwendung beschrieben, die in der Lage ist den Datenverkehr zu und von einer Webanwendung zu analysieren und auf Sicherheitskritische Inhalte zu überprüfen. Eine WAF ist hierbei in der Lage gefährliche Inhalte nicht nur zu blockieren, sondern anfragen auch so zu verändern, dass sie unschädlich sind und trotzdem verarbeitet werden können. Im Gegensatz zu einer *klassischen* Firewall, die auf den Schichten 3 und 4 des OSI/ISO Schichten-Modells arbeitet, ist eine WAF in der Lage passierenden Datenverkehr auf der Anwendungsschicht (Schicht 7) zu analysieren. Hierbei liegt der Fokus hauptsächlich auf dem HTTP. Es ist jedoch auch möglich andere, im Kontext von Webanwendungen genutzte Protokolle wie FTP zu analysieren. Um des gesamten Inhalt der zu analysierenden Nachrichten sehen zu können (Deep Packet Inspection) kann eine WAF Protokolle zur Sicherstellung von Vertraulichkeit und Integrität (SSL/TLS) terminieren.
Es gibt mehrere kommerzielle Hersteller aber auch Opern-Source Entwickler die WAFs zur Verfügung stellen. Diese Angebote können auf eine vielzahl von Wegen logisch vor einer Webanwendung positioniert werden. Während große Hosting- und Serveranbieter wie Cloudflare or Microsoft azure es ermöglichen mittels wenigen kicks eine WAF vor die bei ihnen untergebrachten webserver zu installieren, können die Angebote anderer, eigenständiger WAFs auf unterschiedliche Methoden betrieben werden:

1. Als physischer Server in einem Rechenzentrum
2. Als Eigenständige virtuelle Maschine innerhalb der eingenen Virtualisierungsumgebung
3. Als Einheit in einer Containervirtualisierungsumgebung
4. Als Modul oder Addon einer Webserver-Anwendung (NGINX/Apache webserver)

Daraus ergeben sich zwei gängige Deployment Szenarien.

- *On premise Deployment* bei dem sich die WAF neben den zu schützenden Anwendungen im gleichen Netzwerk befindet
- *Cloud WAF* wo die WAF von einem Drittanbieter zur Verfügung Gestellt wird und mittels IP-Whitelisting sichergestellt werden muss, dass ein zugriff auf die Webanwendung nur durch die WAF erfolgen kann.

Der Betrieb einer WAF erfordert die Anpassung an die zu schützende Anwendung. Es muss eine Konfiguration sichergestellt werden, dass die Funktion einer Webanwendung nicht beeinträchtigt ist ohne dabei die schützenden Funktion der WAF einzuschränken.

Um abgrenzen zu können welche Inhalte für Lerninhalte in Frage kommen und abzuschätzen was in der vorgegeben Zeit vermittelt werden kann wird im folgenden Kapitel eine WAF detailliert beschrieben.

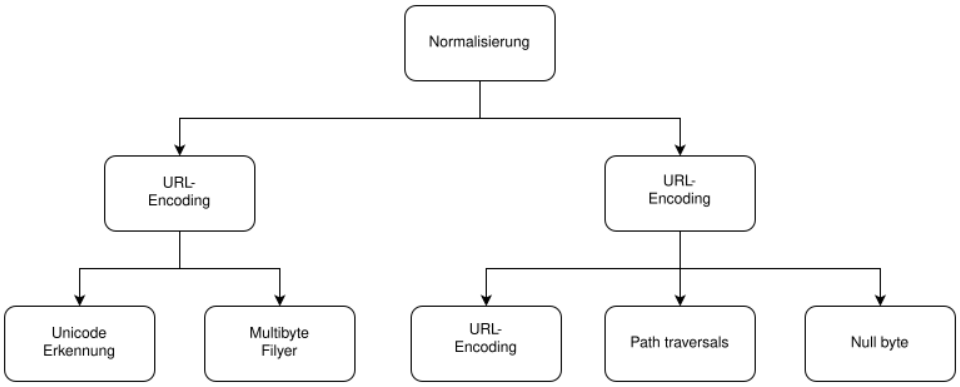

Abbildung 1: Normalisierungs-Schritte

5.4.1 Verarbeitung einer Anfrage

Request Parsing

Muster-Abgleich gegen Regeln

Logging

Weiteres Vorgehen

5.4.2 Erweiterte Funktionen

Lernen von Regeln aus vorhergegangenem Datenverkehr

KI-Features

5.4.3 Deployment einer WAF

Postitionierung einer WAF

Betrieb einer WAF

5.4.4 Schwächen und Nachteile einer WAF

6 Design der Lernumgebung

6.1 Zu übermittelnde Inhalte

6.2 Technische Umsetzung

6.2.1 Evaluation verfügbarer Produkte

Auswahl der WAF! (WAF!)-Anwendung

Verwundbare Anwendungen

6.2.2 Labor-Umgebung

Die in Kapitel 6.1 beschriebenen Inhalte sollen in einem Praxisnahen Umfeld vermittelt werden. Dazu kommt nach den Abwägungen aus Kapitel 6.2.1, die Waf-Applikation ModSecurity zum Einsatz. Die zu diesem Zweck vorgesehene Laborumgebung muss einige Kriterien erfüllen:

Einheitliches Deployment: Der Ausgangspunkt der Lerneinheiten muss reproduzierbar und wiederholbar sein. Bei wiederholten Durchführungen der Übungen soll es einfach sein den Lernenden ohne zusätzlichen Manuellen Konfigurationsaufwand eine Laborumgebung zu übergeben.

Modifizierbarketit der Anwendungen: Um in den Lerneinheiten grundlegende Techniken zu übermitteln, ist es notwendig Basis-Funktionen entfernen zu können. Und die

Bekannte Basis-Technologien:

Komplexe Netzwerkumgebungen:

6.3 Lerneinheiten

6.3.1 Teil 1: Erster Kontakt zu einer WAF

6.3.2 Teil 2: Grundlegende Angriffe

SQL Injections

Cross Site Scripting (XSS)

6.3.3 Teil3: Details und Gefahren in der Nutzung einer WAF

Log Analyse

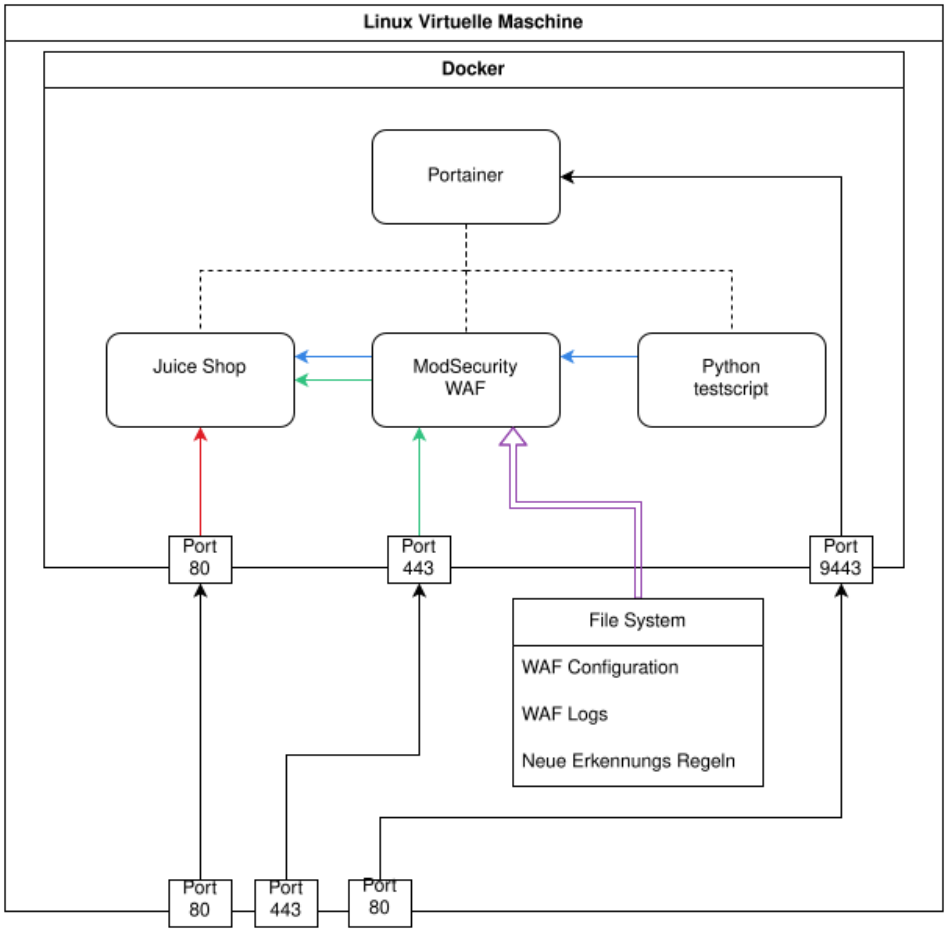

Abbildung 2: Aufbau der Laborumgebung

Filter evasion Taktiken

7 Evaluation

7.1 Evaluation mit Probanden

7.2 Überlegungen zur Bewertung

8 Fazit

9 Bibliografie

[1] *Stand Der Technik*, https://www.teletrust.de/publikationen/broschueren/stand-der-technik/. (besucht am 07. 01. 2024).