Hochschule Wismar
Fakultät für Ingenieurwissenschaften
Bereich Elektrotechnik und Informatik

IT-Forensik-Projekt II

ERKENNEN VON ANGRIFFSMECHANISMEN AUF WEBSERVER UNTER VERWENDUNG VON DVWA UND SPLUNK UND ABLEITUNG VON ABWEHRSTRATEGIEN

Sommersemester 2023

Aufgabenstellung

Im Rahmen der geplanten Arbeit werden beispielhaft bekannte Angriffsmechanismen der OWASP Top 10-Liste auf Webserver unter Verwendung von DVWA und Splunk betrachtet. Konkret werden in einem Docker-Container SQL-Injections, Command-Injections und Cross-Site Scripting (XSS) gegen einen Webserver über DVWA durchgeführt und über Splunk ausgewertet. Die Dokumentation der Splunk-Docker und DVWA-Docker Installation ist Teil dieser Ausarbeitung, ebenso die Forensische Ausarbeitung durch Auswertung der Logs. Im Ergebnis steht zum einem die Nachvollziehbarkeit der Angriffe, die Ableitung von Erkennungsmechanismen. Weiter werden mögliche Abwehrmechanismen beschrieben und die Arbeit mit einem Fazit beendet.

Inhalt

1 Einleitung

Betreiber einer Website oder eines Online-Shop müssen mit Angriffen von Hackern rechnen und sich vor diesen möglichst schützen, denn Angreifer versuchen die Websites z. B. für Spam und Phishing zu missbrauchen. Unter Angriffen können nicht autorisierte Zugriffe und Zugriffsversuche verstanden werden. Einerseits durch das unbefugte Eindringen in Rechenzentren und der Beeinflussung von Personen, z. B. Passwörter und sensible Informationen zu verraten, meist jedoch erfolgen Angriffe über das Netzwerk.[1]

Das Web als Medium, ist nicht nur im Consumer-Bereich unverzichtbar, die Abwicklung der Geschäftsprozesse zwischen Geschäftspartnern (B2B) sowie zwischen Bürger und Behörde (E-Government) nehmen stetig über das Web zu.[2]

Daher sollten sich Anbieter von digitalen Produkten im Web über die Risiken und der möglichen Angriffsszenarien bewusst sein, sowie die Angreifbarkeit untersuchen und Schutzmaßnahmen vornehmen.

Die Bedrohungen für Webanwendungen ändern sich permanent aufgrund kurzer Entwicklungszyklen unter Verwendung neuer Technologien. Neben den klassischen Angriffsmöglichkeiten wie Injections, Cross-Site Scripting (XSS), Distributed Denial of Service (DDoS) oder Brute-Force, werden häufig auch Schwachstellen und Fehler im Software-Design oder in der Softwarearchitektur ausgenutzt. Schwache oder keine ordnungsgemäße Verschlüsselung, sowie Verstöße gegen das „Least Privilege-Prinzip“ oder auch gegen „deny of default“ zählen laut der OWASP Top 10 Liste von 2021, ebenso zu den am häufigsten

---

[1] https://www.buelow-masiak.de/home/home_news_detail.htm?tx_news_pi1%5Bnews%5D=364&tx_news_pi1%5Bcontroller%5D=News&tx_news_pi1%5Baction%5D=detail&cHash=f6701cf2e8360792ace10c55a6b1ceb1 (Aufgerufen am 04.06.2023)

[2] https://www.kuketz-blog.de/risiko-webanwendung-sicherheit-von-webanwendungen-teil1/ (Aufgerufen am 04.06.2023)

ausgenutzten Schwachstellen.

2 Abgrenzung der Arbeit

Die Arbeit beschränkt sich nur auf ausgewählte Angriffe, die in der DVWA nachgestellt werden, wohlwissend, dass eine Vielzahl an weiteren Angriffs-Methodiken existieren. Die Arbeit fokussiert sich demnach auf folgende Angriffe:

SQL Injection, Command Injection, Cross Site Scpripting, Directory Traversal. Weiter ist zu berücksichtigen, dass die forensische Auswertung auf Basis der Webserverlogs erfolgt und keine z. B. Netzwerkmittschnitte herangezogen werden. Die Abwehrmechanismen werden in einem separaten Kapitel beschrieben, eine Umsetzung der Abwehrmechanismen in der DVWA ist nicht Teil dieser Arbeit. Die Entscheidung der verwendeten Tools viel auf Splunk, DVWA sowie Docker. Splunk ist ein Enterprise-SIEM, welches eine hohe Wahrscheinlichkeit mit sich bringt im späteren Verlauf der IT-Karriere eine relevante Rolle zu spielen. DVWA ist ein open-source Projekt, welches stetig weiterentwickelt wird. Neue Angriffsarten werden hier stetig eingepflegt, sodass der Nutzer sich ausschließlich auf die Angriffe fokussieren kann, ohne vorher eine dafür anfällige Web-Umgebung bauen muss. Aufgrund dessen, dass im Projekt-Team bereits Expertise sowohl im Splunk als auch im DVWA und Docker Umfeld vorhanden war, wurden diese Tools verwendet. Es gibt eine Reihe alternativer Tools die stattdessen hätten verwendet werden können, auf welche jedoch aus den genannten Gründen verzichtet worden ist.

3 Methodisches Vorgehen

Die Arbeit beginnt mit dem Betrachten der Angriffsszenarien bezugnehmend zu OWASP. Anschließend wird die Einrichtung der Testumgebung im Docker-Container dargestellt. Im Kapitel darauf, werden die ausgewählten Angriffsszenarien in der zuvor aufgesetzten Umgebung verprobt und forensisch aufgearbeitet. Die Beschreibung der Abwehrmechanismen bilden den Abschluss der Arbeit.

4 Definitionen

4.1 Webserver

Damit eine Webseite im Internet erreichbar ist, benötigt man einen speziellen Server für den Content, welcher durchgehend erreichbar und somit permanent online ist. Hierfür nutzen Webseiten-Betreiber dafür die Rechenzentren von Internet-Providern, große Unternehmen und Organisationen hingegen, nutzen häufig eigenen Webserver, auf denen Sie Ihre Intranet- und Internetinhalte hosten.

In erster Linie ist ein Webserver für die zuverlässige Auslieferung von statischen und dynamischen Inhalten an die anfragenden Clients verantwortlich.

Gibt man eine Internetadresse (z.B. www.wikipedia.de) in Ihren Browser ein, dann sendet dieser eine Anfrage an den Server, der wiederum aus dem Domainnamen die zugehörige IP-Adresse ermittelt.

Für die Übermittlung der Dateien/Inhalte werden die standardisierten Übertragungsprotokolle HTTP sowie die verschlüsselte Variante HTTPS genutzt.

Anschließend baut der HTTP-Client des Browsers mittels TCP (oder ab und an auch per UDP) eine Verbindung zum Webserver auf und stellt an diesen einen Webseiten-Request. Internetseiten bestehen aus verschiedenen HTML-Bausteinen wie Grafiken, Fotos und Videos. Für jede dieser Dateien muss eine eigene Anfrage gestellt werden, die der Webserver durch die Übermittlung der entsprechenden Inhalte beantwortet. Dazu schickt der HTTP-Server die angeforderten Dateien an den HTTP-Client, der diese mithilfe eines Interpreters auf dem Bildschirm darstellt. Hat der Client die komplette Webseite angezeigt bekommen, wird die TCP-Verbindung wieder geschlossen.[3]

[3] https://www.wintotal.de/webserver/ (Aufgerufen 04.06.2023)

4.2 OWASP

OWASP steht für Open Web Application Security Project und ist eine internationale Non-Profit-Organisation, die sich der Sicherheit von Webanwendungen widmet. Sie bietet unter anderem kostenlose Dokumentationen, Tools, Videos und Foren an und richtet sich an Entwickler aber auch an die Verantwortlichen der digitalen Produkte und informiert regelmäßig über neue Angriffsszenarien und wie man diesen vorbeugen kann. Ihr wohl bekanntestes Projekt sind die OWASP Top 10.[4] Dieses wird alle 3-4 Jahre veröffentlicht. Darin werden die am häufigsten auftretenden und am häufigsten ausgenutzten Sicherheitslücken aufgeführt, basierend auf Branchendaten und Ergebnissen umfangreicher unabhängiger Forschung.[5] In der Abbildung 1 findet sich eine Gegenüberstellung der letzten Versionen der OWASP Top 10 Liste. Neue Angriffsszenarien die es unter die Top 10 geschafft haben, sind in der Allbildung 1 markiert. Beispielsweise in 2021 - A08: Software an Data Intergrity Failures. Hier führt OWASP u.a. an, dass viele Anwendungen mittlerweile über eine Funktion zur automatischen Aktualisierung für beispielsweise notwendige Plugins oder Bibliotheken verfügen, diese Aktualisierungen werden dann ohne ausreichende Integritätsprüfung heruntergeladen und auf die zuvor vertrauenswürdige Anwendung angewendet. Angreifer könnten möglicherweise ihre eigenen Updates hochladen, welche dann verteilt und ausgeführt werden.[6]

[4] https://www.cloudflare.com/de-de/learning/security/threats/owasp-top-10/ (Aufgerufen 04.06.2023)

[5] https://de.barracuda.com/support/glossary/owasp (Aufgerufen 04.06.2023)

[6] https://owasp.org/Top10/A08_2021-Software_and_Data_Integrity_Failures/ (Aufgerufen 04.06.2023)

7 8

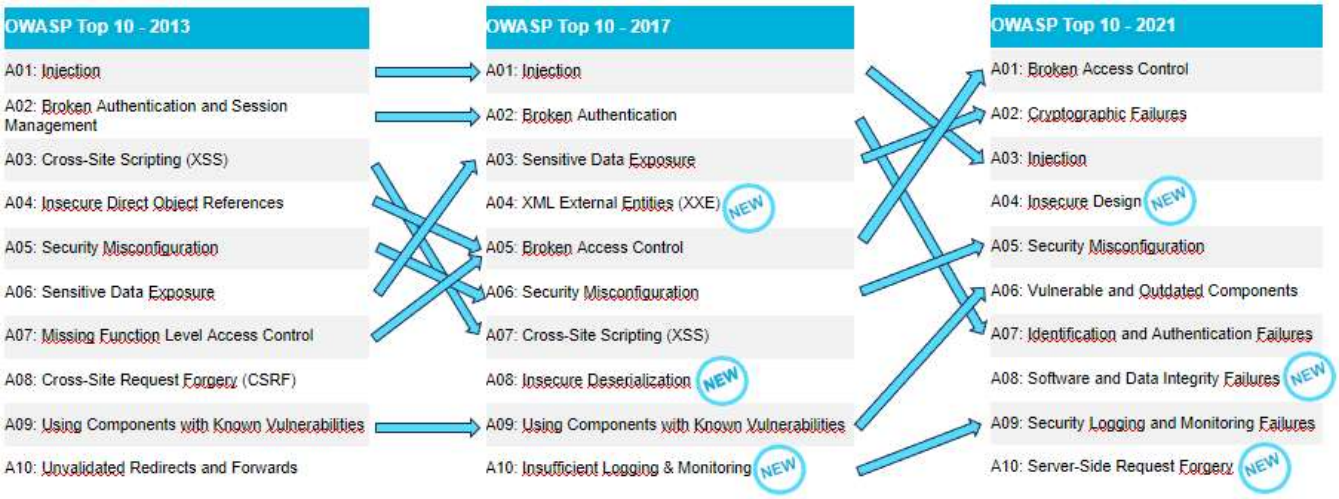

Abbildung 1: OWASP - Top 10 Listen

7 https://owasp.org/www-project-top-ten/2017/Release_Notes (Aufgerufen 08.06.2023)

8 https://owasp.org/Top10/ (Aufgerufen 08.06.2023)

5 Weitere Quelle zu Schwachstellen

Die MITRE Corporation ist ebenso wie OWASP eine Non-Profit-Organisation mit Hauptsitz in den USA. Die Organisation wurde gegründet als gemeinnütziges Unternehmen gegründet, um sowohl militärischen als auch zivilen Regierungsbehörden als objektive Berater im Bereich Systemtechnik zu dienen.[9] Gemäß den CWE-Nutzungsbedingungen kann jede Organisation oder jede Einzelperson, CWE für Forschungs-, Entwicklungs- und/oder kommerzielle Zwecke frei nutzen. Die MITRE Coperation unterhält außerdem CWE. CWE steht für Common Weakness Enumeration und ist eine formale Auflistung häufiger Software- und Hardware-Schwächen, die in Architektur, Design, Code oder Implementierung auftreten und zu ausnutzbarer Sicherheit führen können. Die CWE-Initiative hat sich zum Ziel gemacht, Schwachstellen an der Quelle zu beseitigen, indem Software- und Hardwarekäufer, Architekten, Designer und Programmierer darin geschult werden, wie sie die häufigsten Fehler beseitigen können, bevor ein Produkt ausgeliefert wird.[10] Neben dem MITRE Att&ck Framework (Adversarial Tactics, Techniques & Common Knowledge), eine öffentlich verfügbare Wissensdatenbank über Cyberangriffe samt Taktiken und Verfahren[11], wurde auch eine CWE Top 25 Liste veröffentlicht, diese wird jährlich aktualisiert. Der Unterschied zwischen der OWASP Top 10 und der CWE Top 25 Liste ist der Detaillierungsgrad.

Das sieht man gut in der nachfolgenden Tabelle 1. Während OWASP Injection ganz allgemein als einen Punkt in der Liste führt, schlüsselt CWE das Thema Injection weiter auf:

---

[9] https://www.mitre.org/who-we-are/our-story (Aufgerufen 07.06.2023)

[10] https://cwe.mitre.org/about/faq.html#what_is_cwe_weakness_meaning (Aufgerufen 07.06.2023)

[11] https://www.security-insider.de/was-ist-das-mitre-attck-framework-a-99bb9d4fc17ce387f41bf1d7dd2ebed0/ (Aufgerufen 07.06.2023)

Tabelle 1: OWASP und CWE

OWASP	CWE[12]
A03: Injection	CWE-78: Unsachgemäße Neutralisierung spezieller Elemente, die in einem OS-Befehl verwendet werden („OS Command Injection“) CWE-89: Unsachgemäße Neutralisierung spezieller Elemente, die in einem SQL-Befehl verwendet werden („SQL Injection“) CWE-94: Unsachgemäße Steuerung der Generierung von Code („Code-Injection“)

CWE veröffentlicht sowohl eine TOP 25 Liste für Softwarefehler als auch eine TOP 25 Liste für Hardwarefehler.[13] Die OWASP Top 10 fokussiert sich lediglich auf webanwendungsspezifische Softwareprobleme.

Die MITRE Corporation veröffentlicht zusätzlich auch die CVE-Liste (Common Vulnerabilities and Exposures). Diese kann als Schwachstellenkatalog für spezifische Schwachstellen für konkrete Softwareprodukte angesehen werden, während CWE ein Standard zur Klassifizierung und Beschreibung der Arten von Schwachstellen ist, die zu spezifischen Schwachstellen führen können.[14] [15]

Zur Veranschaulichung wurde folgende Tabelle 2 angelegt um am Beispiel des Angriffes Directory Traversal (auch Path Traversal genannt, fällt unter OWASP –

[12] https://cwe.mitre.org/data/definitions/1344.html (Aufgerufen 07.06.2023)

[13] https://cwe.mitre.org/scoring/lists/2021_CWE_MIHW.html (Aufgerufen 07.06.2023)

[14] https://cwe.mitre.org/about/faq.html#what_is_cwe_weakness_meaning (Aufgerufen 07.06.2023)

[15] https://www.codiga.io/blog/cve-vs-cwe/ (Aufgerufen 07.06.2023)

Broken Access Control) die Zuordnung der CWEs und CVEs aufzuzeigen.

Tabelle 2: OWASP und CWE und CVE

OWASP	CWE	CVE[16]
A01: Broken Access Control	CWE-22: Unsachgemäße Beschränkung eines Pfadnamens auf ein eingeschränktes Verzeichnis („Path Traversal“)	CVE-2023-34409: In Percona Monitoring and Management (PMM) Server 2.x vor 2.37.1 formalisiert und bereinigt die Authentifizierungsfunktion in auth_server.go URL-Pfade nicht ordnungsgemäß. CVE-2023-34407: OfflinePlayerService.exe im Harbinger Offline Player 4.0.6.0.2 ermöglicht die Directory Traversal als LocalSystem über ..\ in einer URL.
	CWE-23: Relative Path Traversal	CVE-2023-1112: In Drag and Drop Multiple File Upload Contact Form 7 5.0.6.1 wurde eine Schwachstelle gefunden. Die Manipulation des Arguments upload_name führt zu einem relativen Path Traversal. Es ist möglich, den Angriff aus der Ferne zu starten. CVE-2022-23854: AVEVA InTouch Access Anywhere-Versionen 2020 R2 und älter sind anfällig für einen Path-Traversal-Exploit, der es einem nicht authentifizierten Benutzer mit Netzwerkzugriff ermöglichen könnte, Dateien auf dem System außerhalb des sicheren Gateway-Webservers zu lesen.

16 https://www.cvedetails.com/vulnerability-list/cweid-22/vulnerabilities.html (Aufgerufen 08.06.2023)

6 Gängige Angriffe auf Webserver

Injection-Angriffe sind stark verbreitet. Es werden durch den Angreifer System- oder SQL-Befehle in Eingabefelder oder Anfragen eingeschleust, um beispielsweise eine Datenbank-Anfrage/Manipulation auszulösen und ggf. vertrauliche Daten ausgeben zu lassen. Bekannte Beispiele sind SQL-Injection oder Command-Injection-Angriffe.

Auch stark verbreitet ist das Cross-Site Scripting (XSS). Es ermöglicht dem Angreifer, eigenen Code in eine Website einzufügen, der dann von dem Besucher der Website ausgeführt wird. Dies kann dazu führen, dass sensible Informationen gestohlen werden oder bösartiger Code ausgeführt wird.[17]

Eine andere Art von Angriffen bietet der Distributed Denial of Service (DDoS). Hier versucht der Angreifer, den Webserver durch eine Flut von Anfragen zu überlasten. Dadurch wird der Server überwältigt und fällt aus, weshalb er nicht mehr auf legitime Anfragen antworten kann.[18]

Bei Brute-Force-Angriffen versucht der Angreifer, sich Zugriff auf den Webserver zu verschaffen, indem er systematisch verschiedene Benutzernamen und Passwörter ausprobiert, bis er erfolgreich ist.

Bei Directory-Traversal zielt der Angreifer, auf Dateien oder Verzeichnisse ab, auf die er normalerweise keinen Zugriff hat, indem er spezielle Zeichenfolgen oder Pfade in URLs verwendet.

Ganz allgemein kann man sagen, dass Schwachstellen in einer Software durch Angreifer identifiziert und ausgenutzt werden. Sie suchen nach bekannten Schwachstellen in der verwendeten Software des Webservers, wie z.B. veralteten Versionen von CMS-Systemen oder nicht gepatchten

---

17 https://crashtest-security.com/de/code-injection-angriff/ (Aufgerufen 08.06.2023)

18 https://www.akamai.com/de/glossary/what-is-ddos (aufgerufen 08.06.2023)

Sicherheitslücken. Diese Schwachstellen werden ausgenutzt, um unautorisierten Zugriff zu erlangen oder bösartigen Code einzuschleusen.

Es ist daher wichtig, dass Webserver-Betreiber verschiedene Sicherheitsmaßnahmen ergreifen, um sich vor solchen Angriffen zu schützen. Dazu gehören regelmäßige Updates der Software, die Verwendung sicherer Passwörter, das Validieren und Filtern von Benutzereingaben, das Implementieren von Firewalls und Intrusion Detection/Prevention-Systemen sowie das Überwachen des Serververkehrs auf verdächtige Aktivitäten. Auf die Abwehrmechanismen wird konkret in Kapitel 10 eingegangen.

7 Schutz vor Angriffen auf Webserver

Ein Schutz vor Angriffen auf Webserver ist auch im Sinne des Datenschutzes. Denn ein erfolgreicher Angriff kann dazu führen, dass sensible Informationen gestohlen werden. Dies kann personenbezogene Daten wie Benutzerdaten, Kundendaten andere vertrauliche Informationen betreffen. Durch den Schutz des Webservers schützt man auch die Privatsphäre und die Datenintegrität der Nutzer.

Ein Angriff kann außerdem zu einer Unterbrechung des Dienstes führen. Der Server könnte überlastet, beschädigt oder manipuliert werden, was zur Nichtverfügbarkeit der Website führt. Umsatzeinbußen, Vertrauensverlust bei den Nutzern und ein Image-Schaden für das Unternehmen können die Folge sein.

Ein Vertrauensverlust und ein Image-Schaden können auch entstehen, wenn die Webseite manipuliert wird, indem schädlicher Code eingeschleust und Daten abgegriffen werden.

Unternehmen unterliegen verschiedener gesetzlichen Anforderungen und Compliance-Richtlinien. Verstöße gegen diese Anforderungen aufgrund eines Sicherheitsvorfalls, können zu rechtlichen Konsequenzen, Bußgeldern und Reputationsschäden führen.[19]

Neben technischen Sicherheitsmaßnahmen, wie regelmäßigen Updates und Patches, starken Authentifizierungsverfahren, sowie einem Monitoring-Betrieb, sollten auch organisatorische Maßnahmen getroffen werden wie die Schulung der Mitarbeiter in ihre, Sicherheitsbewusstsein.

.

[19] https://www.badencloud.de/it-security-gefahr-durch-cyberangriffe (Aufgerufen 08.06.2023)

8 Einrichten des Testsystems

Die vorliegende Ausarbeitung beschäftigt sich mit der Konfiguration und Integration eines Testsystems auf Windows, das für die Sicherheitsprüfung von Webanwendungen entwickelt wurde. Das Testsystem besteht aus drei Hauptkomponenten, nämlich dem Damn Vulnerable Web Application (DVWA)-Framework, der Splunk-Plattform und Docker.

Angesichts der wachsenden Bedeutung von Webanwendungen in unserer heutigen digitalen Ära ist es von größter Bedeutung, ihre Sicherheit zu gewährleisten. Angreifer nutzen zunehmend ausgeklügelte Techniken, um Schwachstellen in Webanwendungen auszunutzen und Zugriff auf sensible Daten zu erlangen. Daher ist es unerlässlich, umfassende Sicherheitstests durchzuführen, um potenzielle Schwachstellen zu identifizieren und entsprechende Schutzmaßnahmen zu implementieren.

Um diesen Herausforderungen gerecht zu werden, wurde das hier beschriebene Testsystem genutzt. Das Damn Vulnerable Web Application (DVWA)-Framework dient als speziell entwickelte Testanwendung, die eine Vielzahl bekannter Schwachstellen und Angriffsvektoren enthält. Es ermöglicht Entwicklern und Sicherheitsexperten, verschiedene Angriffsszenarien zu simulieren und Schwachstellen aufzudecken.

Parallel dazu spielt Docker eine entscheidende Rolle bei der Einrichtung des Testsystems, da es eine effiziente Möglichkeit bietet, Anwendungen in isolierten Containern auszuführen. Docker ermöglicht die einfache Bereitstellung und Skalierung von Anwendungen, ohne dabei Konflikte mit dem zugrunde liegenden Betriebssystem zu verursachen. Sobald Docker erfolgreich installiert ist, kann ein Container für das DVWA-Framework erstellt werden.

Die Integration von Splunk in das Testsystem ist von großer Bedeutung, da Splunk eine leistungsstarke Plattform für das Sammeln, Analysieren und Visualisieren von Protokolldaten darstellt. Splunk bietet fortschrittliche Such- und Berichtsfunktionen, um Sicherheitsbedrohungen zu erkennen und darauf zu

reagieren. Nach der Installation von Splunk erfolgt die Konfiguration, um die Protokolldaten des DVWA-Frameworks zu erfassen und in Splunk zu übertragen. Dies umfasst die Festlegung spezifischer Protokolle und Schnittstellen, um relevante Daten in Echtzeit zu erfassen und in Splunk zu indizieren.

Nach erfolgreicher Einrichtung des Testsystems stehen umfassende Funktionen zur Verfügung, um Sicherheitstests von Webanwendungen durchzuführen. Das DVWA-Framework ermöglicht die Identifizierung von Schwachstellen und die Durchführung verschiedener Angriffsszenarien, während Splunk die Analyse und Überwachung von Protokolldaten erleichtert, um potenzielle Sicherheitsbedrohungen zu erkennen.

Im Rahmen dieser Ausarbeitung erfolgt eine selektive Analyse und Evaluierung dieses Testsystems, um dessen Effektivität bei der Erkennung und Abwehr von definierten Sicherheitsrisiken zu untersuchen. Es werden verschiedene Testfälle durchgeführt, um die Leistungsfähigkeit und Zuverlässigkeit des Systems zu überprüfen. Dabei werden potenzielle Schwachstellen identifiziert, mögliche Angriffsszenarien simuliert und die Reaktion des Testsystems darauf ausgewertet.

8.1 Docker

Docker Desktop wird von der offiziellen Seite heruntergeladen und auf einem Windows-Computer installiert.[20] Eine eigene Konfigurationsdatei für den Docker wurde geschrieben.

[20] https://www.docker.com/products/docker-desktop/ (Aufgerufen 07.06.2023)

```
version: '3'
services:
  dvwa:
    image: vulnerables/web-dvwa
    ports:
      - 80:80
    volumes:
      - dvwa-logs:/var/log/apache2

  splunk:
    image: splunk/splunk:latest
    user: "0:0"
    environment:
      - SPLUNK_START_ARGS=--accept-license
      - SPLUNK_PASSWORD=Ch@ngeMe!
      - SPLUNK_USER=root
    ports:
      - 8000:8000
      - 8089:8089
    volumes:
      - dvwa-logs:/var/log/dvwa:ro

volumes:
  dvwa-logs:
```

Abbildung 2: Screenshot - Docker-Compose-Datei

Diese sog. Docker-Compose-Datei beinhält alle relevanten Parameter um einen Container mit dem DVWA-Image sowie einen Container mit Splunk zu starten. Außerdem haben beide Container Zugriff auf ein gemeinsames Verzeichnis „dvwa-logs“. In dieses Verzeichnis lagert der DVWA-Webserver seine Log-Files aus. Splunk hat somit später Zugriff auf diese Logfiles und kann diese einlesen.

Anschließend wird im cmd der Befehl “docker-compose-up" ausgeführt, um den DVWA und Splunk-Container mit den angegebenen Konfigurationen zu erstellen.

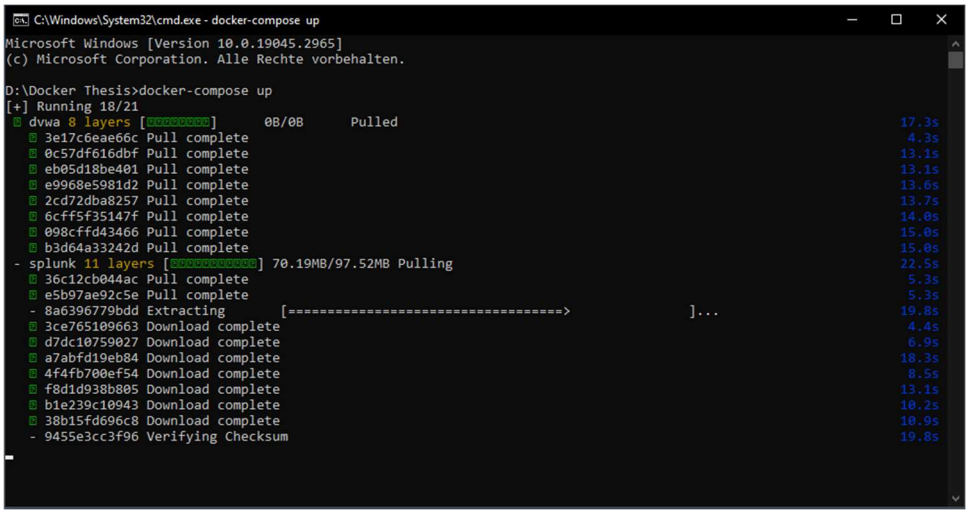

Abbildung 3: Screenshot - Ausführung Docker-Compose-Datei

Nachdem die Einrichtung erfolgt ist, sind zwei neue Container im Docker verfügbar.

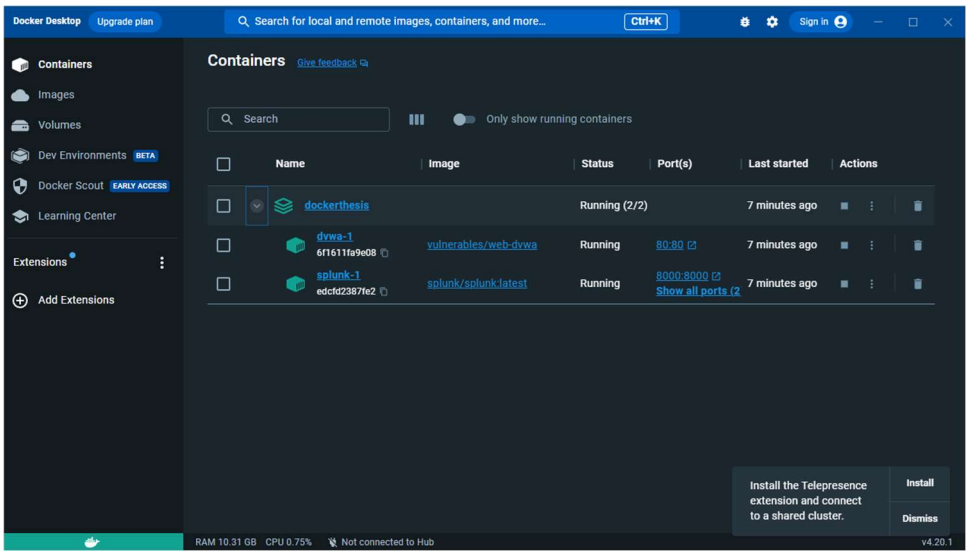

Abbildung 4: Screenshot - Container in Docker

Dort den hinterlegten Link in der Spalte “Port” ist ein direkter Zugriff auf DVWA und Splunk durch den Browser möglich.

8.2 DVWA

Die im DVWA hinterlegten Einstellungen werden nur zwecks Übersicht dargestellt. Auf einzelne Einstellungen und die Gründe für die Wahl wird nicht

eingegangen.

Über den Link im Docker wird auf DVWA zugegriffen. Auf der linken Seite befindet sich nun eine Auswahl an Angriffsszenarien, die abgebildet bzw. versucht werden können.

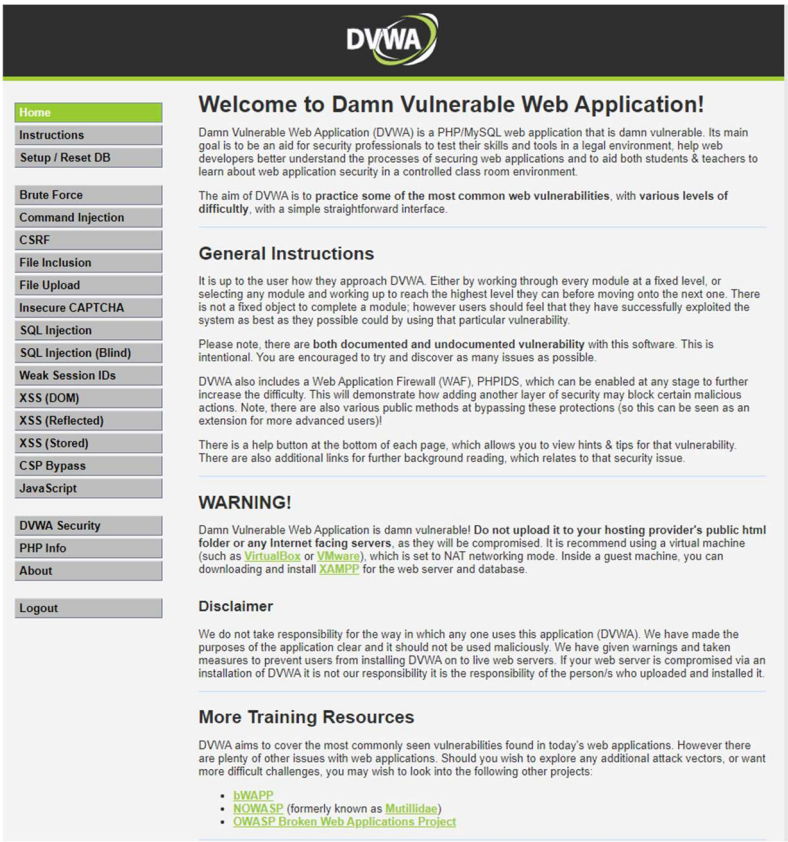

Abbildung 5: Screenshot - Docker Startseite

Die DVWA bietet mehrere Sicherheitsstufen. Alle Tests laufen auf der niedrigsten Sicherheitsstufe der Anwendung.

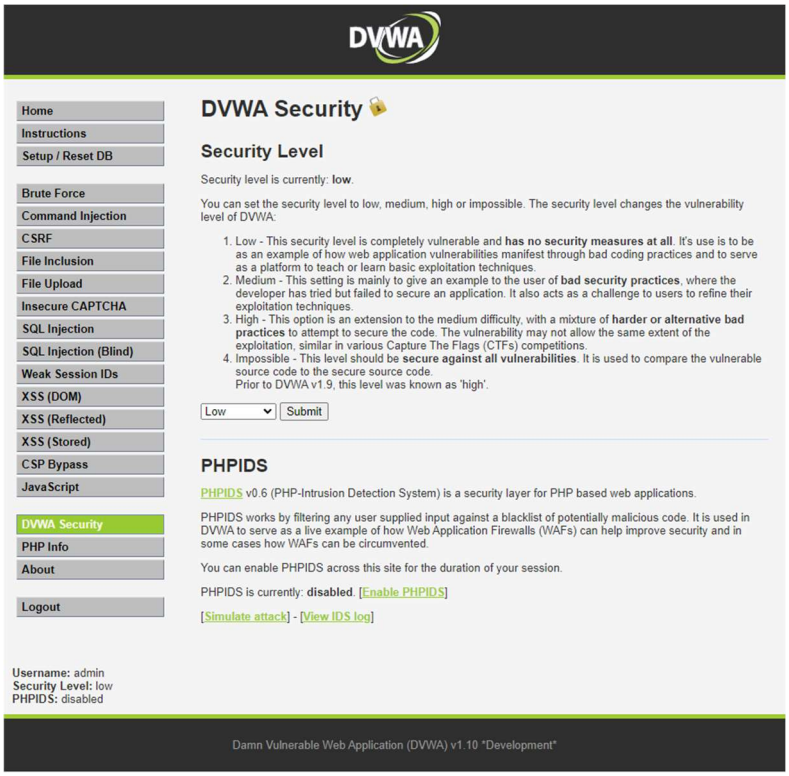

Abbildung 6: Screenshot - DVWA Security Level

Die DVWA ist nun eingerichtet und kann genutzt werden. Um eine Dokumentation der SQL-Injections zu gewährleisten, wird anschließend Splunk eingerichtet.

8.3 Splunk

Über den Link im Docker wird nun auf Splunk zugegriffen und eingerichtet. Da das Ziel der Ausarbeitung nicht die Einrichtung des Splunk ist, wird das Thema nur beiläufig behandelt. Im Abbildungsverzeichnis werden die Screenshots der einzelnen Schritte jedoch hinzugefügt.

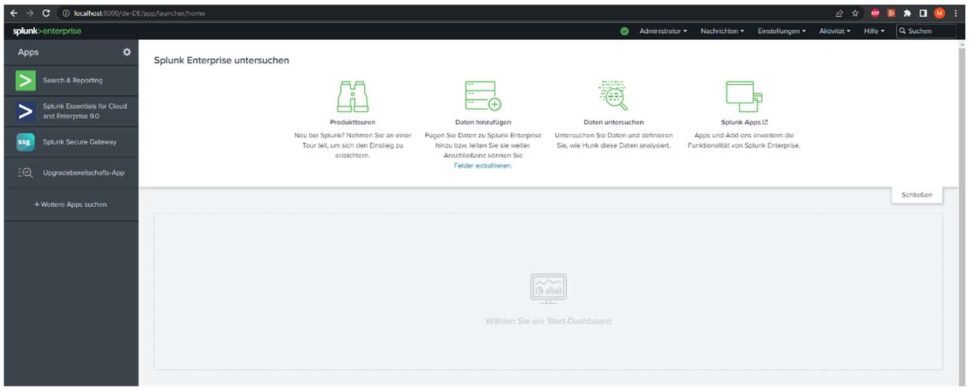

Abbildung 7: Screenshot - Splunk

Die Parameter in Splunk werden so eingerichtet, dass genaustens nachvollzogen werden kann, welche Angriffe auf den Webserver erfolgt sind. Hierfür wird eine eigene App in Splunk mit dem Namen “Wings Web Attack Forensics” erstellt.

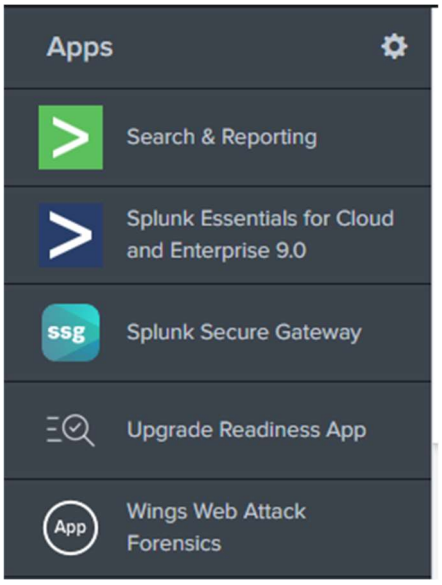

Abbildung 8: Screenshot - Splunk Apps

Zur genauen Analyse werden die drei nachfolgenden Add-ons mit Splunk verknüpft:

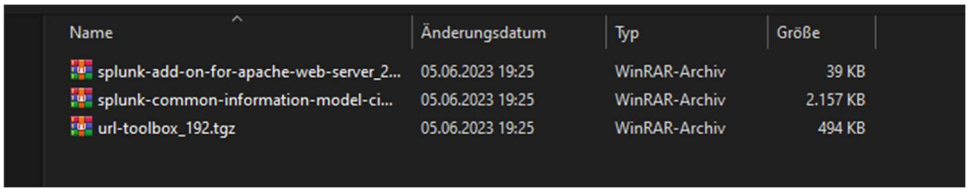

Abbildung 9: Screenshot - Splunk Addons

Bei dem “Splunk Common Information Model (CIM)” handelt es sich um eine Splunk Erweiterung für eine einheitliche Darstellung verschiedener Datenquellen, um eine effizientere Analyse möglich zu machen.[21]

Das Add-on für den Apache-Web-Server dient der Erfassung, Analyse und Überwachung der Logdaten, sodass jegliche Zugriffe auf das DVWA nachvollzogen werden kann. Es liefert so zu sagen die Parsing-Informationen für Splunk, wie Felder und dazugehörige Werte später in den einzelnen Protokolleinträgen zu parsen sind.[22]

Als letztes Add-On wurde die Splunk URL-Toolbox installiert. Eine der Hauptfunktionen von URL-Toolbox ist das korrekte Parsen von URLs und komplizierten TLDs (Top Level Domain) unter Verwendung der Mozilla Suffix List. Andere Funktionen wie Shannon-Entropie, Zählen, Suites, Bedeutungsverhältnis, bayesianische Analyse usw. sind ebenfalls verfügbar.[23]

Innerhalb der Spunk-Benutzoberfläche werden die Dateien jeweils einzelnen hochgeladen.

---

21 https://splunkbase.splunk.com/app/1621 (Aufgerufen 08.06.2023)

22 https://splunkbase.splunk.com/app/3186 (Aufgerufen 08.06.2023)

23 https://splunkbase.splunk.com/app/2734 (Aufgerufen 08.06.2023)

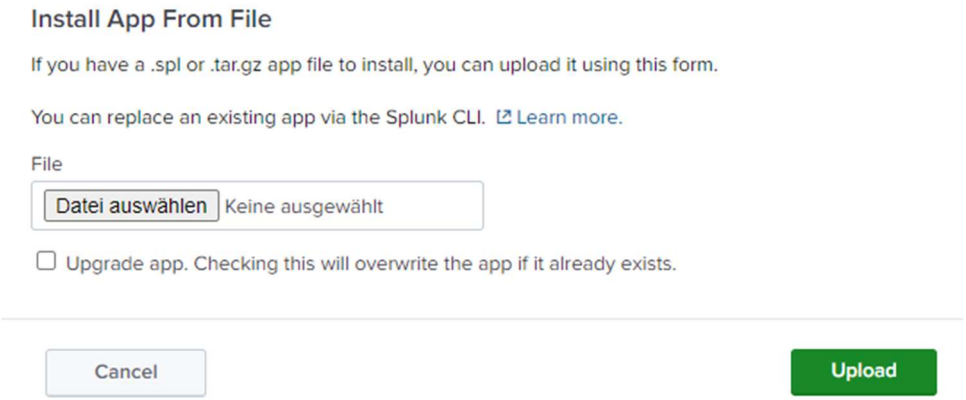

Abbildung 10: Screenshot - Hochladen Dateien in Splunk

8.4 Datamodels

Ein Splunk-Datamodel ist eine organisierte Struktur, die es ermöglicht, Daten in Splunk effizient zu analysieren und zu visualisieren. Es handelt sich um eine logische Darstellung von Daten, die auf spezifische Anwendungsfälle oder Datenquellen zugeschnitten ist.

Ein Splunk-Datamodel besteht aus einer Sammlung von Objekten wie Ereignistypen, Feldern, Makros und Beschreibungen, die miteinander verknüpft sind, um einen bestimmten Aspekt der Daten darzustellen. Es kann zum Beispiel ein Sicherheitsdatenmodell sein, das speziell für die Analyse von Sicherheitsereignissen entwickelt wurde, oder ein Netzwerkdatenmodell, das Informationen über Netzwerkverkehr und Verbindungen enthält.

Insgesamt ermöglichen Splunk-Datamodelle eine bessere Strukturierung und Analyse von Daten in Splunk, um wertvolle Erkenntnisse zu gewinnen und komplexe Fragestellungen zu beantworten. Bei der Entwicklung unserer automatischen Alarme ist immer auf die CIM-Datamodels zurückgegriffen worden. Diese stammen aus der zuvor installierten App „Splunk Common Information Model (CIM)“.

8.5 Automatische Alarme

Um im späteren Verlauf jeden einzelnen Angriff genauer analysieren zu können und zu prüfen ob etwaige Angriffe als diese automatisiert identifiziert werden können, wurden Alarme innerhalb von Splunk errichtet. Diese Alarme greifen teilweise auf sog. „Macros“ zurück. Macros in Splunk sind dabei in erster Linie nichts anderes als simple „Listen“. Also Aneinanderreihungen von Strings, Wörtern, Buchstaben o. ä.

In dem untenstehenden Beispiel sehen wir das Macro „SQL_INJECTION_INDICATORS“. Dieses beinhält Schlagworte, welche oft im Zusammenhang mit SQL Injections gesehen werden.[24] Der Inhalt unseres Macros stammt hierbei aus bspw. Online Repositories wie Gitlab aber auch den Ergebnissen unserer eigenen SQL Injection Versuche aus vorherigen Semestern und Modulen.

24 https://github.com/SigmaHQ/sigma/blob/master/rules/web/webserver_generic/web_sql_injection_in_access_logs.yml

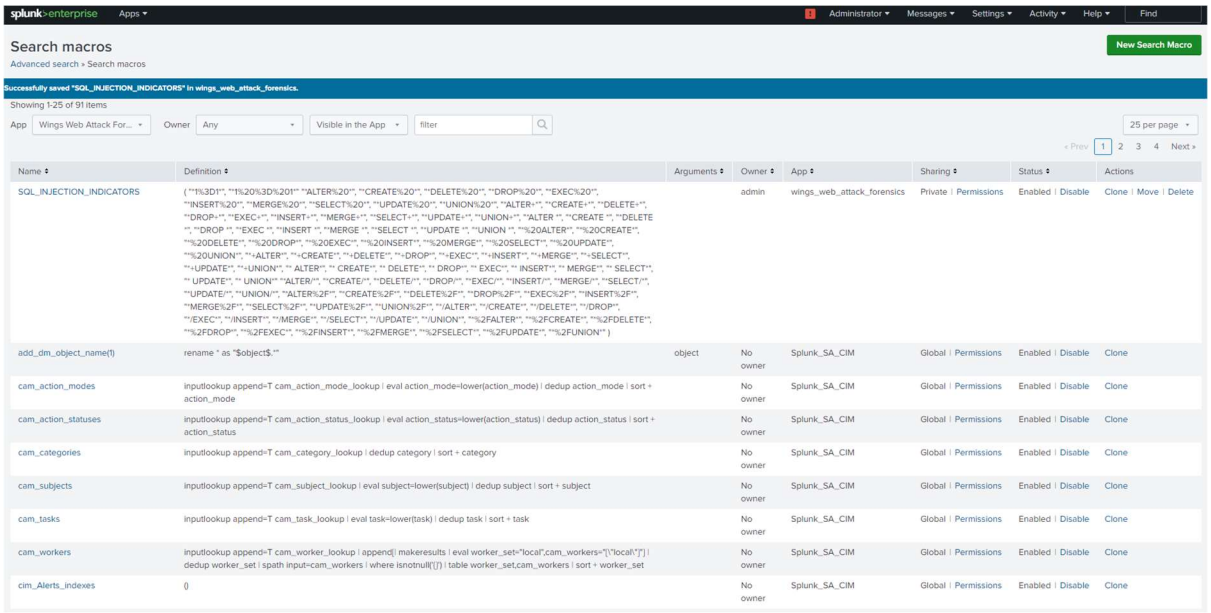

Abbildung 11: Screenshot - Splunk Macro

Nachdem die Macros entsprechend befüllt waren, wurden die eigentlichen Alarme eingerichtet.

Abbildung 12: Screenshot - Splunk Alarme

Anschließend werden testweise SQL-Injections in DVWA getätigt, um sicherzustellen, dass Splunk richtig eingerichtet worden ist.

Im Nachfolgenden wird exemplarisch der Alarm für einen SQL-Injection Alarm erläutert.

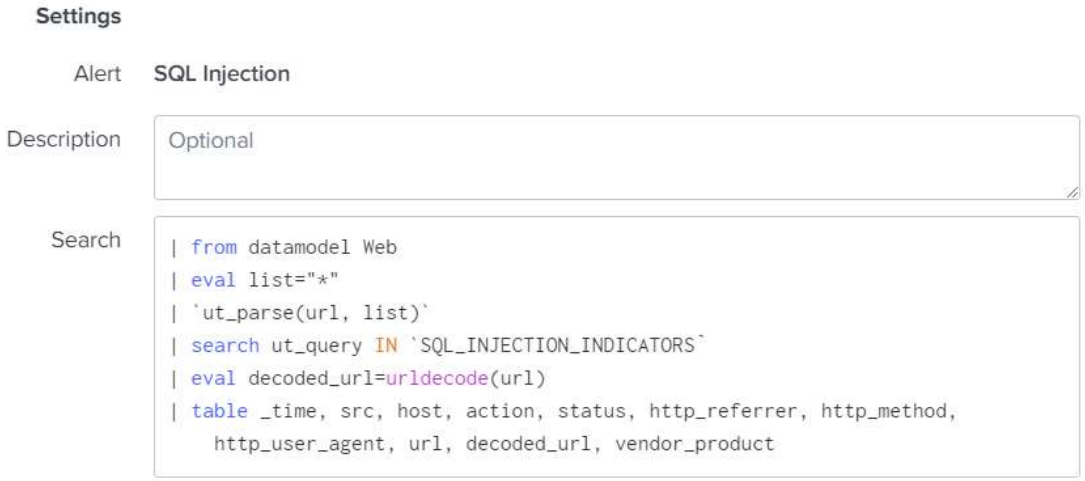

Abbildung 13: Screenshot - Splunk Alarm SQL-Injection

Mittels der sog. Splunk Language (SPL) wurden Suchabfragen definiert, welche nach bestimmten Mustern in den Protokolleinträgen des Webservers suchen. „| from datamodel Web“ zeigt in diesem Beispiel alle Protokolleinträge des Webservers an. Die darauffolgenden zwei Zeilen sind Funktionen aus der eingangs beschriebenen URL-Toolbox. Diese Funktionen ermöglichen es uns das Feld „URL“ in detailliertere Sub-Felder zu teilen. Anschließend werden die Felder nach den vorher definierten SQL-Injection-Indikatoren durchsucht. Bei einem Treffer liefert Splunk alle in dem „Table“-Befehlt definierten Felder tabellarisch und zeigt diese an. Das Feld URL wird dabei außerdem dekodiert dargestellt, sodass typische URL-Codes in Klartext übersetzt angezeigt werden.

Diese Alarme werden in Splunk so definiert, dass sie automatisch permanent die Log-Files überwachen. Sobald ein Alarm Ergebnisse bringt, wird ein Alarm ausgelöst. Dieser ist dann in der Alarm-Übersicht in Splunk zu sehen.

	Time	Fired alerts	App	Type	Severity	Mode	Actions
☐	2023-06-17 09:50:00 UTC	Directory Traversal	wings_web_attack_forensics	Scheduled	High	Per Result	View results
☐	2023-06-17 09:50:00 UTC	SQL Injection	wings_web_attack_forensics	Scheduled	High	Per Result	View results
☐	2023-06-17 09:48:00 UTC	SQL Injection	wings_web_attack_forensics	Scheduled	High	Per Result	View results

Abbildung 14: Screenshot - Splunk Alarme Übersicht 1

Abbildung 15: Screenshot - Splunk Alarme Übersicht 2

9 Durchführung der Angriffe

9.1 SQL-Injection

9.1.1 Kurzbeschreibung

SQL - Structured Query Language ist eine Abfrage-Sprache für die Verwendung und Kommunikation mit einer Datenbank. Mittels SQL können Aktionen zum Abrufen, Löschen und Speichern von Daten in der Datenbank durchgeführt werden.

Ein Angreifer versucht, die in der Webanwendung verwendete SQL-Abfrage zu manipulieren und durch einen SQL-Injection Angriff direkten Zugriff auf Ihre Daten zu erlangen. Dies geschieht in der Regel über ein Eingabefeld oder Kommentarfeld eines Webformulars oder andere für den Benutzer frei zugängliche Möglichkeiten.

Werden diese Eingaben nicht überprüft und der SQL-Befehl als Datenbankanfrage von der Datenbank ausgeführt, werden je nach SQL-Befehl, Daten, beispielsweise von anderen Nutzern, ausgegeben gelöscht oder verändert.

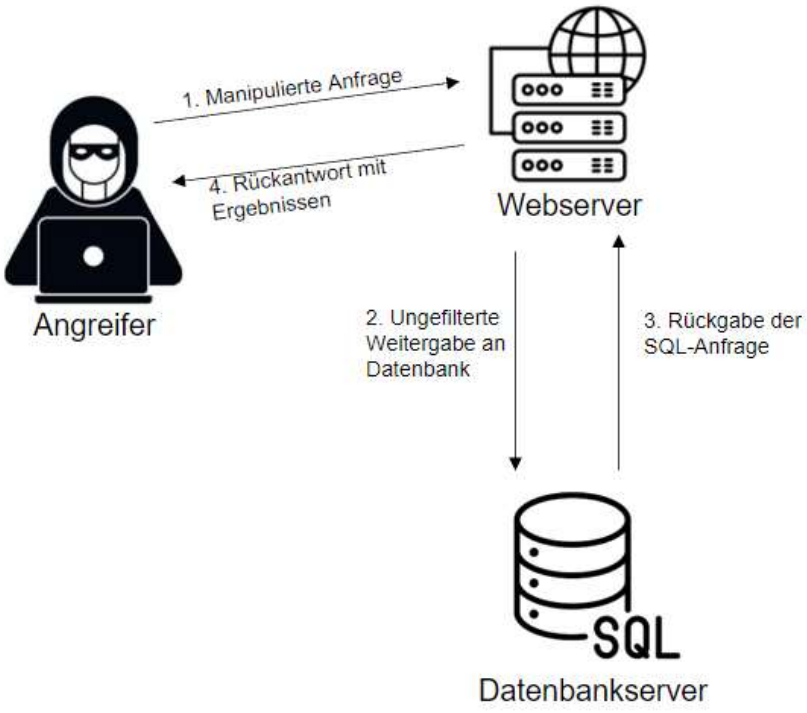

Abbildung 16: Ablauf SQL-Injection

9.1.2 Durchführung und forensische Aufarbeitung

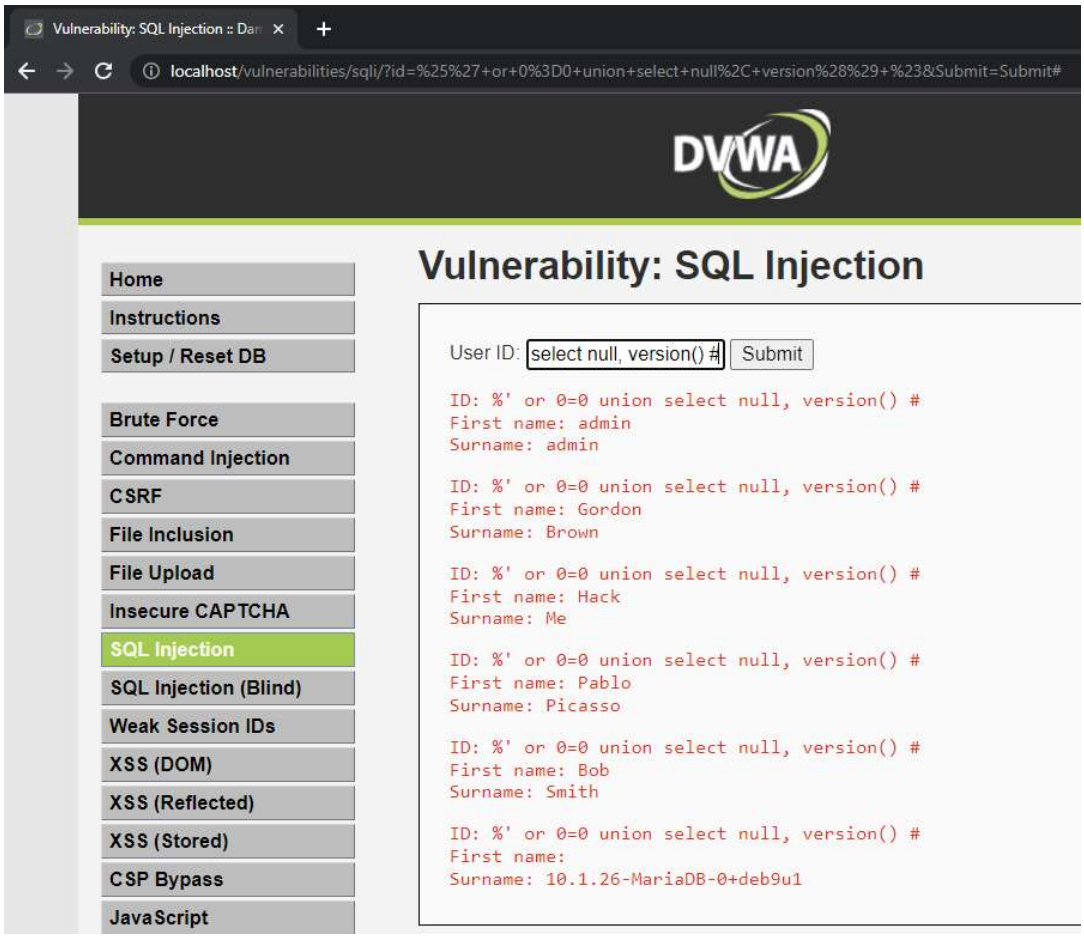

Abbildung 17: Screenshot - DVWA SQL-Injection

Das Eingabe-Feld wird anstatt mit einzelnen Zahlen befüllt zu werden, um die Nutzernamen der dahinterstehenden IDs zu erfahren, derart befüllt um den Versionsstand der verwendeten SQL-Datenbank zu erfahren.

%' or 0=0 union select null, version() #

Dieser Angriff wird unmittelbar von einem der vorher definieren Alarme detektiert.

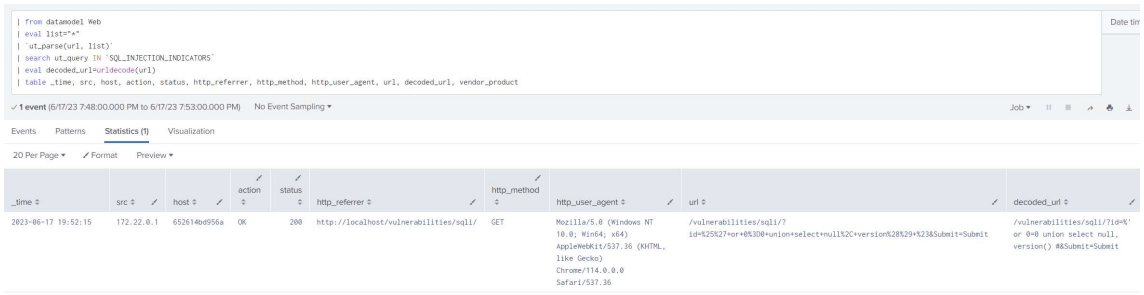

Abbildung 18: Screenshot - Splunk SQL-Injection

9.2 Cross-Site Scripting (XSS)

9.2.1 Kurzbeschreibung

Das Ziel von Cross-Site Scripting (XSS) besteht darin, schädlichen Code in eine vertrauenswürdige Website einzufügen und dann Besucher der Website dazu zu bringen, diesen Code auszuführen. Dies kann über das Eingabeformular einer Webseite von Webshops, Foren, Blogs und Wikis erfolgen. Die eingegebenen Daten werden auf der Webseite wieder als Seiteninhalt ausgegeben, wenn die Seite von Benutzern aufgerufen wird. So ist es möglich, manipulierte Daten an alle Benutzer der Webseite zu senden und somit Schadcode auf der Seite des

Clients auszuführen oder auch Daten wie z.B. Session-Cookies abzugreifen.

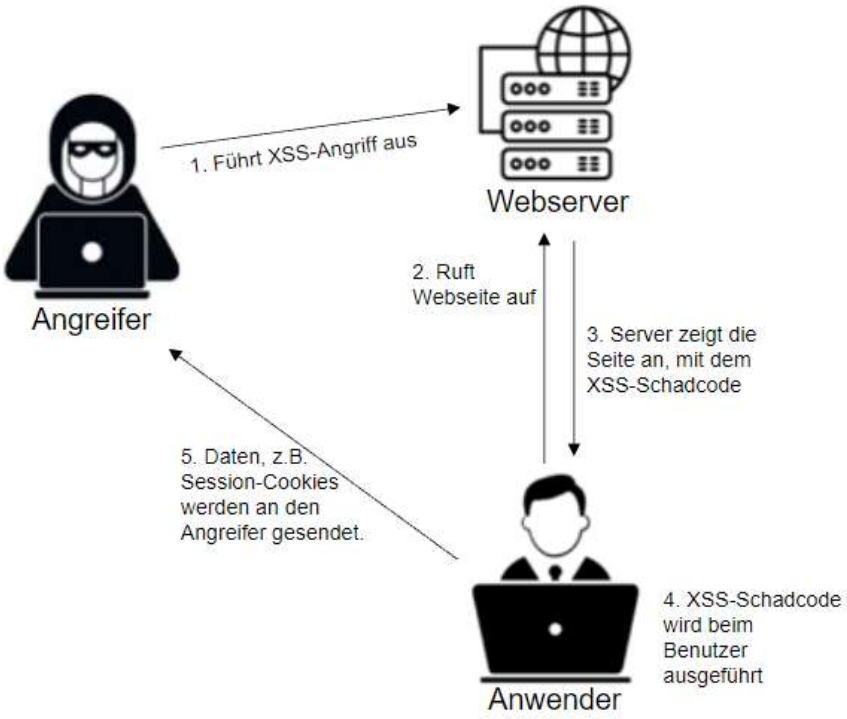

Abbildung 19: Ablauf XSS-Angriff

9.2.2 Durchführung und forensische Aufarbeitung

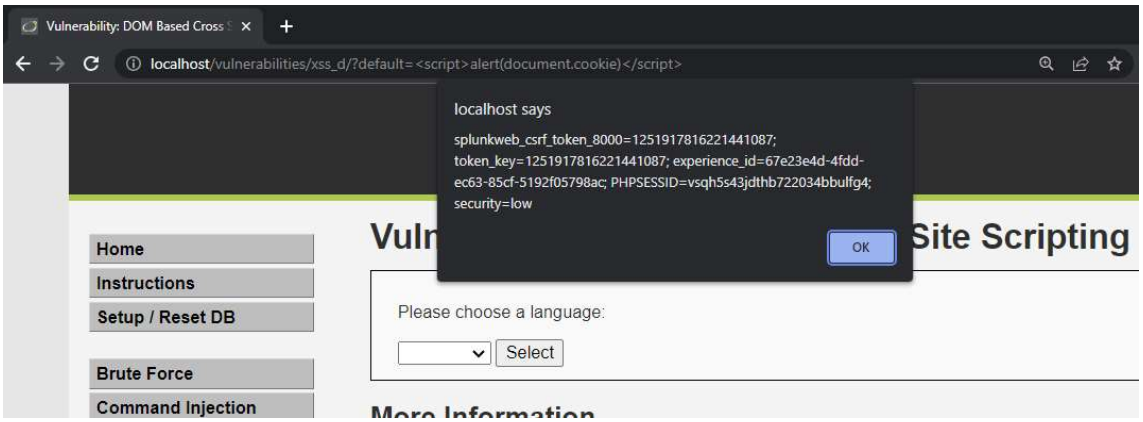

Abbildung 20: Screenshot - DVWA XSS

Mittels „<script>alert(document.cookie)</script>"-Parameter in der URL kann der Session Cookie der aktuellen Browser-Sitzung ausgelesen und angezeigt werden. Dieser Angriff spiegelt sich in der URL wider, weswegen die Versuche in Splunk unmittelbar sichtbar werden.

Abbildung 21: Screenshot Splunk XSS

9.3 Directory Traversal

9.3.1 Kurzbeschreibung

Das Ziel besteht darin, über einen manipulierten Pfad an vertrauliche Dateien zu gelangen. Ein Angreifer könnte unter Umständen Dateien mit eigenen Inhalten überschreiben, oder sich die Inhalte bestimmter Dateien ausgeben lassen.

Der Angreifer ändert einen gültigen Pfad zu einer Seite/Datei ab und setzt „../../“ ein um Dateien aus einem höherem Verzeichnis abzurufen. Bei diesen abgerufenen Dateien könnte es sich um vertrauliche interne Dokumente handeln die nicht nach außen gelangen sollen, jedoch nicht ausreichend geschützt sind vor eben einem Angriff dieser Art.

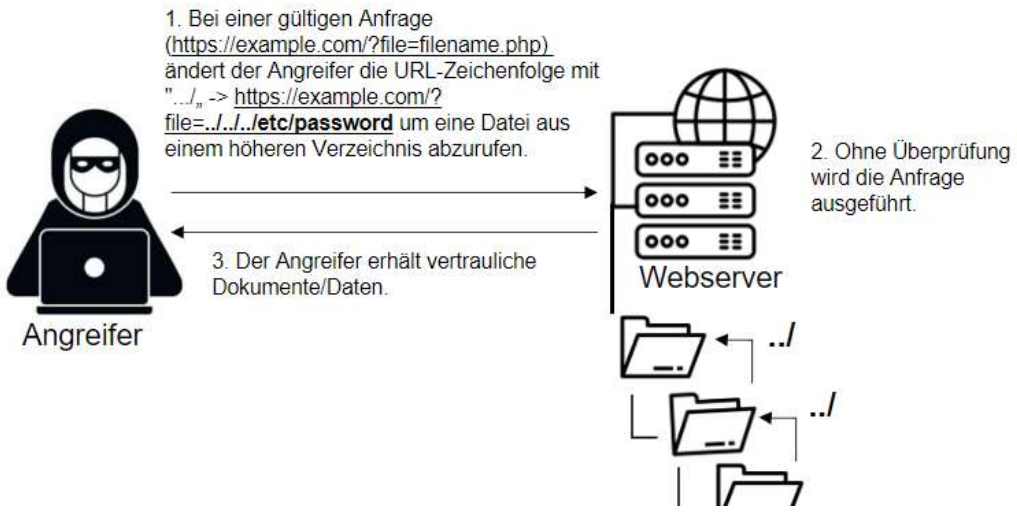

Abbildung 22: Ablauf Directory Traversal

9.3.2 Durchführung und forensische Aufarbeitung

Beim Directory Traversal Versuch, wird der page-Parameter in der URL missbraucht um nicht eine Datei aus dem aktuellen Verzeichnis zu öffnen, sondern um mehrere Ebenen höher, bspw. die Passwd-File zu öffnen.

Aufgrund dessen, dass DVWA in diesem Beispiel weder prüft was angesurft wird, noch die Berechtigungen auf höher liegende Verzeichnisse entsprechend sicher konfiguriert sind, können wir uns die Passwd-Datei anzeigen lassen.

Auch dieser Angriff ist in Splunk unmittelbar sichtbar.

Abbildung 23: Screenshot - Splunk Directory Traversal

9.4 Command-Injection

9.4.1 Kurzbeschreibung

Webanwendungen müssen auch Systembefehle auf dem Webserver abrufen, auf dem sie ausgeführt werden. Wenn die Benutzereingabe nicht validiert und eingeschränkt wird, kann es zu einer Command Injection kommen. Das Ziel einer Command Injection ist es, schädliche Befehle in ein System einzufügen und diese vom System ausführen zu lassen.

Kennt der Angreifer das verwendete Betriebssystem, fügt er einen Befehl in das System ein, indem er diesen in ein Eingabefeld eingibt, dieser Befehl wird dann auf dem Host-System ausgeführt.

Eine Command Injection kann die Anwendung und ihre Daten sowie das gesamte

System, angeschlossene Server, Systeme und andere Infrastrukturen gefährden.

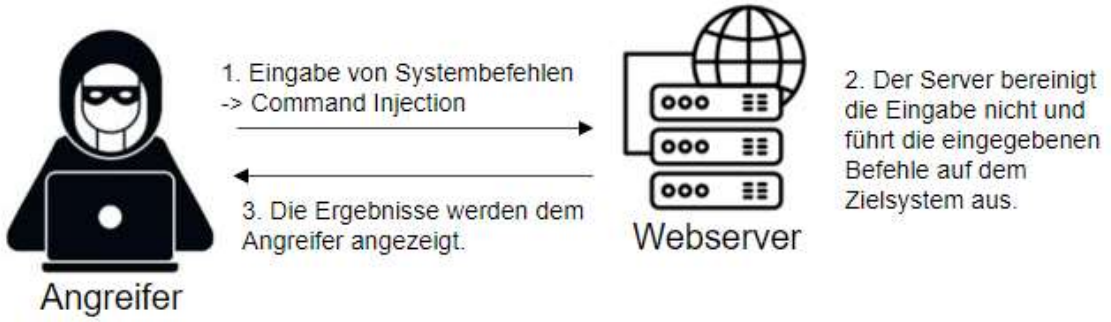

Abbildung 24: Ablauf Command Injection

9.4.2 Durchführung und forensische Aufarbeitung

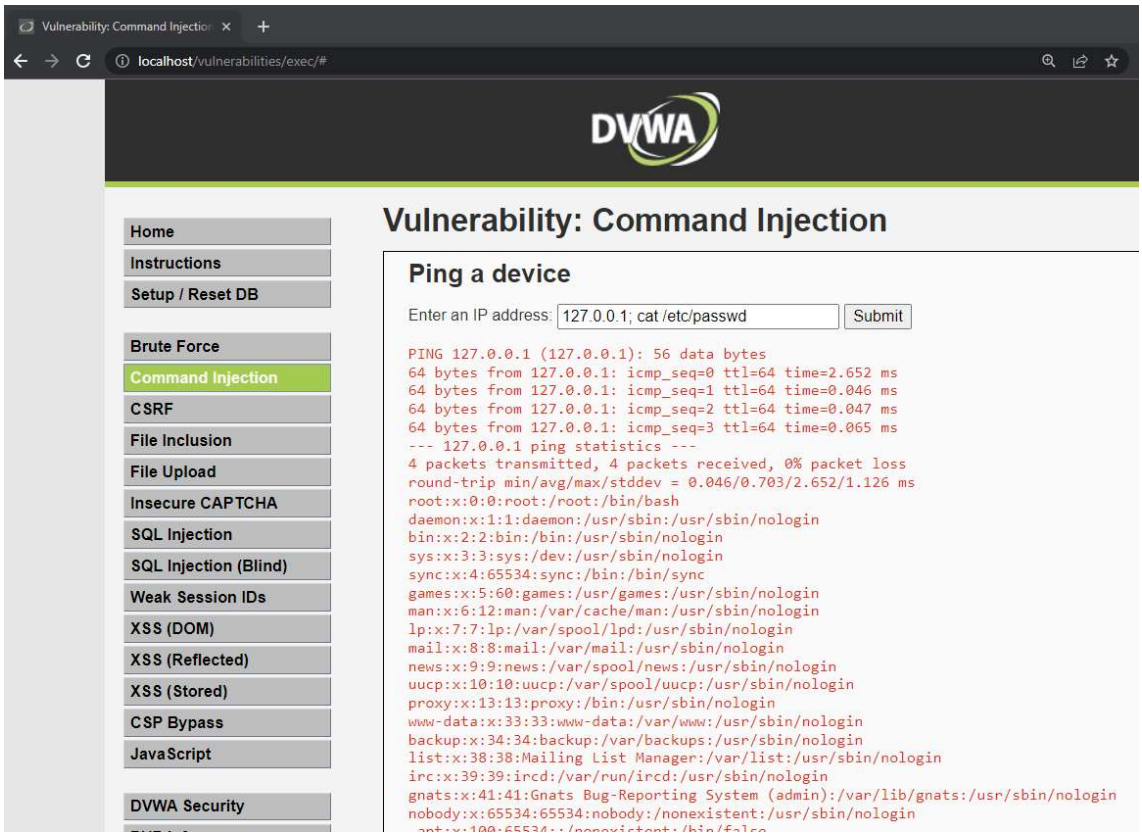

Abbildung 25: Screenshot - DVWA Command-Injection

Wie im Screenshot 25 zu erkennen ist, wird der Befehl diesmal nicht über die URL übergeben. Daher ist es nicht möglich einen Angriff mittels Command

Injection ausschließlich anhand der Webserver Logs zu Detektieren. Daher ist es wichtig, neben den eigentlichen Webserver-Protokollen auch immer weitere Protokolle wie bspw. die des darunter liegenden Betriebssystems mit einzusammeln.

10 Abwehrmechanismen

10.1 Web-Application Firewall (WAF)

Eine Web Application Firewall (WAF) umschreibt eine Sicherheitslösung, die zwischen einem Web-Server und dem Internet platziert wird, um die Web-Applikation vor Angriffen zu schützen. Sie überwacht den HTTP- und HTTPS-Datenverkehr zwischen dem Nutzer und dem Webserver und analysiert eingehende Anfragen sowie ausgehende Antworten, um potenzielle Bedrohungen zu erkennen und ggf. zu blockieren.

Die Hauptfunktionen einer WAF umfassen:

Angriffserkennung: Eine WAF verwendet verschiedene Techniken wie Signatur-Erkennung, Analyse von Verhalten sowie Erkennung von Mustern, um bekannte Angriffe zu identifizieren.

Angriffsprävention: Sobald eine Bedrohung mittels der eben beschriebenen Methoden erkannt wird, kann die WAF-Maßnahmen ergreifen, um den Angriff abzuwehren. Dies kann bspw. das Blockieren der schädlichen Anfrage, das Filtern von schädlichen Inhalten oder das Zurückweisen des Datenverkehrs umfassen.

Verhaltensüberwachung: Eine WAF analysiert den Datenverkehr kontinuierlich und erkennt ungewöhnliche oder verdächtige Muster. Dies kann auf Abweichungen von normalem Benutzerverhalten, unerwarteten Datenmustern oder Anomalien im Datenverkehr basieren. Eine WAF kann daher bspw. zusätzlich „Lernen“ was normaler Datenverkehrt ist und bei Abweichungen dessen alarmieren.

Zugriffskontrolle: Eine WAF kann außerdem auch Zugriffskontrollmechanismen bereitstellen, um den Datenverkehr nur von vertrauenswürdigen Quellen oder bestimmten Benutzern oder IP-Adressen zuzulassen. Dies kann helfen, Denial-of-Service (DoS)-Angriffe und andere böswillige Aktivitäten einzuschränken. Der

Zugriff von unautorisierten Nutzern ist demnach nicht mehr möglich. Die Autorisierung kann hierbei entweder durch ein Passwort, ein Zertifikat oder anderen Mechanismen erfolgen.

Protokollierung und Reporting: Eine WAF zeichnet Ereignisse und Vorfälle auf, um eine umfassende Übersicht über die Sicherheitslage der Webanwendung zu bieten. Protokolle und Berichte können zur Analyse von Angriffsmustern, zur forensischen Untersuchung und zur Erfüllung von Compliance-Anforderungen verwendet werden. Diese Protokolle können anschließend in SIEM-Systeme wie bspw. Splunk überführt werden, um zu einem späteren Zeitpunkt genauere Analysen zu vollführen.

Eine WAF kann als eigenständiges Gerät oder als Software in einer Netzwerkumgebung implementiert werden. Es gibt auch cloudbasierte Lösungen, bei der der Datenverkehr über externe Server umgeleitet wird, die die Analyse und Filterung dann durchführen.[25] [26]

10.2 Netzwerksegmentierung

Netzwerksegmentierung ist ein Sicherheitskonzept, bei dem ein Netzwerk in kleinere Teilnetzwerke oder Segmente unterteilt wird. Jedes Segment enthält eine Gruppe von Ressourcen mit ähnlichen Sicherheitsanforderungen oder Funktionen. Die Segmente sind voneinander isoliert und der Datenverkehr zwischen ihnen wird über Firewalls, Router oder andere Sicherheitsvorrichtungen kontrolliert.

Die Netzwerksegmentierung bietet verschiedene Vorteile für den Schutz der IT-

[25] https://www.rapid7.com/de/cybersecurity-grundlagen/web-application-firewalls/

[26] https://www.f5.com/de_de/glossary/web-application-firewall-waf

Infrastruktur:

Reduzierung der Angriffsfläche: Durch die Aufteilung des Netzwerks in Segmente wird die Angriffsfläche reduziert. Selbst wenn ein Angreifer Zugriff auf ein Segment erhält, ist es schwieriger, auf andere Segmente zuzugreifen, die unterschiedliche Sicherheitskontrollen haben. Dies erschwert es potentiellen Angreifern, sich in der gesamten Infrastruktur auszubreiten.

Kontrolle des Datenverkehrs: Die Segmentierung ermöglicht eine feinere Kontrolle über den Datenverkehr innerhalb des Netzwerks. Durch den Einsatz von Firewalls oder anderen Sicherheitsvorrichtungen können Regeln festgelegt werden, um den Datenverkehr zwischen den Segmenten zu überwachen, zu filtern und zu begrenzen. Dadurch können bösartige Aktivitäten erkannt und blockiert werden.

Durchsetzung von Sicherheitsrichtlinien: Jedes Segment kann spezifische Sicherheitsrichtlinien haben, die auf die darin enthaltenen Ressourcen zugeschnitten sind. Durch die Segmentierung kann die Durchsetzung von Sicherheitsrichtlinien für den Zugriff, die Authentifizierung und die Datenfreigabe vereinfacht werden. Dies erhöht die Sicherheit, da jede Ressource nur die erforderlichen Berechtigungen und den Zugriff auf andere Segmente hat.

Reduzierung von Auswirkungen von Angriffen: Wenn ein Segment kompromittiert wird, kann die Segmentierung dazu beitragen, dass der Schaden begrenzt bleibt. Durch die Isolation der Segmente kann sich ein Angriff oder eine Kompromittierung nicht unmittelbar auf die gesamte IT-Infrastruktur ausbreiten. Dies ermöglicht eine schnellere Erkennung, Eindämmung und Wiederherstellung der gesamten Umgebung nach einem Angriff.

Erfüllung von Compliance-Anforderungen: Die Segmentierung kann helfen, bestimmte Compliance-Anforderungen zu erfüllen, insbesondere in Bereichen wie Datenschutz, Datentrennung oder dem Schutz sensibler Informationen. Durch die klare Trennung von Daten und Ressourcen in verschiedene Segmente

können Compliance-Vorgaben leichter eingehalten werden.[27] [28]

10.3 Demilitarisierte Zone

Die demilitarisierte Zone (DMZ) ist ein Teilnetzwerk in einer IT-Infrastruktur, das als eine Art Pufferzone zwischen dem internen Netzwerk und dem externen, unsicheren Netzwerk (normalerweise das Internet) fungiert. Die DMZ ist physisch oder logisch von den internen Netzwerken getrennt und enthält Systeme und Ressourcen, die für den öffentlichen Zugriff bestimmt sind.

Die Hauptfunktion einer DMZ besteht darin, die Sicherheit des internen Netzwerks zu erhöhen, indem sie eine Schutzschicht für öffentlich zugängliche Dienste und Ressourcen bereitstellt.

Öffentlich zugängliche Ressourcen wie beispielsweise Webserver werden also in einer Zone platziert, um zu vermeiden, dass potenziell unsichere Verbindungen direkt auf das interne Netzwerk zugreifen können.[29]

10.4 Sicherheits-Updates / Patches

Sicherheitsupdates spielen eine entscheidende Rolle bei der allgemeinen IT-Sicherheit. Sie dienen dazu, bekannte Schwachstellen und Sicherheitslücken in Software, Betriebssystemen, Anwendungen und anderen IT-Komponenten zu beheben. Diese Schwachstellen können von Angreifern ausgenutzt werden, um unbefugten Zugriff zu erlangen, Malware einzuschleusen oder andere schädliche Aktivitäten durchzuführen. Durch das regelmäßige Anwenden von Sicherheitsupdates werden diese Lücken geschlossen und das Risiko von Angriffen erheblich reduziert. Neben der Behebung von Sicherheitslücken

---

27 https://www.zscaler.de/resources/security-terms-glossary/what-is-network-segmentation

28 https://www.paloaltonetworks.de/cyberpedia/what-is-network-segmentation

29 https://www.fortinet.com/de/resources/cyberglossary/what-is-dmz

können Updates auch Fehlerbehebungen, Leistungsverbesserungen und neue Sicherheitsfunktionen enthalten, die die Integrität und Zuverlässigkeit Ihrer Systeme erhöhen. Viele Sicherheitsstandards und Compliance-Richtlinien (wie beispielsweise die DSGVO oder PCI DSS) verlangen das regelmäßige Anwenden von Sicherheitsupdates.

10.5 Sichere Authentifizierung und Zugriffskontrollen

Eine starke Authentifizierung und Zugriffskontrolle ist entscheidend, um unbefugten Zugriff auf Webanwendungen zu verhindern. Dies beinhaltet die Verwendung von sicheren Passwörtern, der Implementierung von Zwei-Faktor-Authentifizierung (2FA) und der Begrenzung der Zugriffsrechte auf erforderliche Funktionen.

Durch eine sichere Authentifizierung und Zugriffskontrollen wird sichergestellt, dass nur autorisierte Benutzer auf die IT-Infrastruktur zugreifen können. Dies verhindert den unbefugten Zugriff auf sensible Daten, Systeme und Ressourcen. Ohne angemessene Authentifizierungsmethoden könnten sich potenzielle Angreifer leicht als legitime Benutzer ausgeben und auf vertrauliche Informationen oder Systeme zugreifen. Zugriffskontrollen ermöglichen außerdem eine granulare Steuerung und Verwaltung von Benutzerrechten. Das bedeutet, dass Benutzer nur auf die Ressourcen zugreifen können, die für ihre Rolle und Verantwortlichkeiten relevant sind. Dadurch wird das Prinzip des geringsten Privilegs (Principle of least privilege) umgesetzt, bei dem Benutzer nur die minimalen Berechtigungen erhalten, die für ihre Arbeit erforderlich sind. Dies reduziert das Risiko von Fehlkonfigurationen, versehentlichen Datenverlusten oder böswilligen Handlungen durch interne Benutzer.

So erhält beispielsweise ein Nutzer eines Webshops lediglich Zugriff auf dessen eigenen Bereichs um ihm Rahmen seiner Rolle (Kunde) zu agieren. Nicht jedoch auf den Bereich anderer Nutzer sowie die generelle administrative Oberfläche

des Webshops.[30] [31] [32] [33]

10.6 Input Validation (Eingabe Überprüfung)

Input-Validierung bezieht sich auf den Prozess der Überprüfung und Filterung von Benutzereingaben, um sicherzustellen, dass sie den erwarteten Formaten, Mustern und Grenzen entsprechen, bevor sie in eine Anwendung oder einen Webserver eingefügt werden. Der Zweck der Input-Validierung besteht darin, schädliche oder unerwünschte Eingaben abzufangen und die Sicherheit und Zuverlässigkeit der Anwendung zu gewährleisten. Durch eine umfassende Input-Validierung können Webserver gegen verschiedene Arten von Angriffen geschützt werden, indem potenziell schädliche Eingaben erkannt und abgelehnt werden.[34] [35]

10.7 Intrusion Detection / Intrusion Prevent Systeme

Ein Intrusion Detection System (IDS) und ein Intrusion Prevention System (IPS) sind Sicherheitssysteme, die entwickelt wurden, um potenzielle Bedrohungen und Angriffe in einem Netzwerk oder einer IT-Umgebung zu erkennen und darauf zu reagieren. Obwohl IDS und IPS ähnliche Funktionen haben, gibt es einige

---

30 https://www.cloudflare.com/de-de/learning/access-management/what-is-access-control/

31 https://www.security-insider.de/was-ist-zugriffskontrolle-a-1084944/

32 https://www.computerweekly.com/de/antwort/Was-sind-die-gaengigen-Methoden-zur-Authentifizierung

33 https://www.bsi.bund.de/DE/Themen/Verbraucherinnen-und-Verbraucher/Informationen-und-Empfehlungen/Cyber-Sicherheitsempfehlungen/Accountschutz/Zwei-Faktor-Authentisierung/zwei-faktor-authentisierung_node.html

34 https://cheatsheetseries.owasp.org/cheatsheets/Input_Validation_Cheat_Sheet.html

35 https://www.w3.org/WAI/tutorials/forms/validation/

Unterschiede zwischen den beiden.[36]

10.7.1 Intrusion Detection System (IDS)

Ein IDS überwacht den Datenverkehr und die Aktivitäten in einem Netzwerk, um verdächtige oder potenziell schädliche Aktivitäten zu identifizieren. Es analysiert den Netzwerkverkehr, die Systemprotokolle und andere Ereignisdaten, um nach Anzeichen von Angriffen oder Verstößen gegen Sicherheitsrichtlinien zu suchen. Das IDS erkennt Angriffssignaturen oder anomales Verhalten, vergleicht sie mit einer Datenbank bekannter Angriffsmuster und löst bei Bedarf einen Alarm aus, um den Sicherheitsadministratoren eine Benachrichtigung zu geben. Ein IDS kann dabei helfen, potenzielle Sicherheitsvorfälle zu erkennen und zu dokumentieren, ermöglicht jedoch keine direkte automatische Reaktion auf Angriffe.

10.7.2 Intrusion Prevention System (IPS)

Ein IPS geht über die Funktionen eines IDS hinaus, da es nicht nur auf Bedrohungen hinweist, sondern auch aktiv Maßnahmen ergreift, um Angriffe abzuwehren oder zu blockieren. Ein IPS kann den Netzwerkverkehr analysieren, die erkannten Angriffsmuster identifizieren und darauf basierend automatisch Schutzmaßnahmen ergreifen. Dies kann beispielsweise das Blockieren verdächtiger IP-Adressen, das Ändern der Firewall-Regeln oder das Zurückweisen schädlicher Pakete umfassen. Ein IPS ist in der Lage, proaktiv auf Angriffe zu reagieren und die Sicherheitslücken in Echtzeit zu schließen.[37]

---

[36] https://www.juniper.net/de/de/research-topics/what-is-ids-ips.html

[37] https://www.varonis.com/de/blog/ids-vs-ips-was-ist-der-unterschied

Grundsätzlich lässt sich somit sagen, dass nicht jeder einzelne Abwehrmechanismus gleich stark gewichtet in Bezug auf Wichtigkeit und Kritikalität ist. Jedes einzelne Mittel hat hierbei seine Daseinsberechtigung und spielt eine wichtige Rolle beim Abwehren von Angriffen auf Webservern. Es ist viel mehr das Zusammenspiel der verschiedenen Mechanismen, das einen guten Schutz ausmacht. Dabei sollte stet das Prinzip der Verteidigung in der Tiefe beachtet werden.

Das Prinzip der Verteidigung in der Tiefe (Defense in Depth)[38] besagt, dass mehrere Sicherheitsschichten implementiert werden sollten, um verschiedene Angriffsszenarien abzudecken. Durch das Zusammenspiel verschiedener Abwehrmechanismen wie Firewalls, Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), Web Application Firewalls (WAFs), Sicherheitsupdates und sicherer Authentifizierung wird ein umfassender Schutzmechanismus geschaffen. Wenn eine Sicherheitsmaßnahme umgangen wird, stehen noch andere Schichten bereit, um Angriffe abzuwehren oder zu erkennen.

[38] https://www.forcepoint.com/de/cyber-edu/defense-depth

11 Fazit

Mittels des beschriebenen Setups aus Docker, Splunk und DVWA ist eine genaue Analyse der Schwachstellen des Webservers möglich. DVWA bietet kostenlos Möglichkeiten an, um gängige Angriffe wie z.B. Injections auszuprobieren und zu begreifen. Splunk ist außerdem eine sehr verbreitete Lösung hinsichtlich des Monitorings von Systemen und Reporting von Auffälligkeiten. Sodass die Verwendung der erstellten Testumgebung eine gute Basis bildet um verschiedene Angriffs-Methodiken zu verstehen, zu verfolgen und auswerten zu können und um praktische Erfahrung mit dem bei Unternehmen oftmals eingesetzten Tool Splunk, sammeln zu können. Entscheidend ist ein Verständnis für die diversen Angriffe, von denen man als Webseiten-Betreiber betroffen sein könnte. Nur so kann eine Sicherheitsstrategie entwickelt und technische wie organisatorische Maßnahmen etabliert werden. Technische Sicherheitsvorkehrungen müssen nicht unbedingt teuer sein, denn es gibt viele Open-Source Alternativen die verwendet werden können beispielsweise für das Monitoring. Nur Webserverlogs alleine reichen nicht aus um eine vollumfängliche forensische Auswertung gewährleisten zu können, es sollten noch Informationen über die Systeme wie Version des Betriebssystems oder Netzwerkmittschnitte herangezogen werden.

Abbildungsverzeichnis

Tabellenverzeichnis