Hochschule Worms

Fachbereich Informatik

Studiengang Angewandte Informatik B.Sc.

Penetration Testing kommerzieller Webanwendungen

Dokumentation des Praxissemesters bei der Firma

WALLSEC

Bruno Macedo da Silva
676839
inf3645@hs-worms.de
Bebelstraße 22 Z18
67549 Worms

Betreuer	Prof. Dr. Zdravko Bozakov Peter Todorov
Bearbeitungszeitraum:	Summer Semester 2022
Abgabedatum:	xxx. XXXXXX XXXX
Sperrvermerk:	Ja/Nein

Inhaltsverzeichnis

Glossar

Confidentiality, Integrity and Availability (CIA) Beschreibt die drei wichtigsten Schutzziele der IT-Sicherheit, und zwar Vertraulichkeit, Integrität und Verfügbarkeit (Wendzel, 2018) .

Common Vulnerability Scoring System (CVSS) Internationale Standards für die Bewertung von Verwundbarkeiten von IT-Systemen. Es wurde im Jahr 2005 von dem National Infrastructure Advisory Councils entstanden und ist heute von dem Forum of Incident Response and Security Teams verwaltet (Security Insider, 2019) .

National Institute of Standards and Technology (NIST) USA-Behörden dafür zuständig, Reglungen im Bereich Informationstechnologie zu vereinheitlichen und voranzutreiben (Hochschule Worms, 2018) .

Open-source intelligence (OSINT) Datenerhebung und Sammlung aus offenen Quellen, wie von Online-Repositories, Nachrichten, sozialen Netzwerken, wissenschaftliche Texten unter anderen öffentlichen Quellen. In diesem Fall gibt es keine direkte Kontakte mit dem Ziel. Es kann passive Reconnaissance genannt werden (Yeboah-Ofori, 2018).

Open Web Application Security Project®(OWASP®) Non-Profit Organisation, die sich darauf fokussiert, die Sicherheit in dem Umgang mit Webanwendungen zu gewährleisten. Die Organisation verteilt Open-Source Informationen über sichere Entwicklung, Dokumentation, Best-Practices zu dem sicheren Umgang in dem Internet und Bildung (Triaxiom Security, 2018).

Rules of Engagement (ROE) Bezieht sich auf ein vertragliches Dokument, der zwischen Kunden und Tester geschlossen wird, um den Umfang und die Rahmenbedingungen des Testes festzulegen. In diesem Dokument steht unter anderen folgenden Informationen: Umgang mit sensitiven Daten, Notfallkontakten, Identifikation der

zu testenden Objekten und Einschränkungen des Testobjekte (Triaxiom Security, 2018).

Damn Vulnerable Web Application (DVWA) ist eine mit Schwachstellen absichtlich entwickelte Webanwendung für Test- und Lernumgebung. Sie wird meist von Entwickler verwendet, um sich mit Schwachstellen und Best-Practices zu kennen. Dieses Plattform hat als Umfang die meisten bekannten Web-Angriffe (DVWA TEAM, 2016) .

Proof of Concept (PoF) ist eine Demonstration, dass eine Methode funktioniert. In dem Sicherheitsbereich zeigt, dass eine Schwachstelle ausnutzbar ist. (Malwarebytes, 2022).

Cross-Site Scripting (DVWA) ist ein Angriff, wo bösartige Code in eine Webanwendung absichtlich hinzugefügt wird, um Anmeldedaten oder Sitzungen-Informationen zu fangen. In diesem Fall Ziel des Angriffes ist es, eine legitime Nutzung der Anwendung vorzutäuschen, um Informationen zu stehlen (wie Passwörter, persönliche oder finanzielle Daten) oder die Anwendung zu beschädigen (Mahmoud et al., 2017) .

Burp Suite auch Burp genannt ist eine von der Firma PortSwigger in Java-Programmiersprache entwickelte Anwendungen für die Durchführung von Sicherheitstests in Webanwendungen. Mit verschiedenen Funktionalitäten unterstützt Anwendungen unterstützt während allen Phasen eines Penetration Testings von Reconnaissance bis zum Angriff (Junmei and YanChengkang, 2021).

Cortex Wie TheHive Project, Cortex ist auch Open Source Plattform für die Verwaltung und Weiterarbeitung von Sicherheitsvorfälle. Es funktioniert wie eine Analysis Engine, die Informationen sammelt und Antworten/Aktionen je nach Fälle durchführt. Es kann allein oder integriert mit TheHive funktionieren (Project, 2021).

Cyberangriff Angriffe, die über den Cyberspace stattfinden. Solche Angriffe zielen Unternehmen und deren Infrastrukturen, um sie zu zerstören, sie zu lähmen, sie zu kontrollieren oder die Integrität deren Daten zu stehlen oder zu dominieren (NIST, 2020).

Cybersicherheit Dieser Domäne umfasst Kenntnisse und Methode für den Schutz, für die Prävention, für die Wiederherstellung von elektronischen Kommunikationsmittel und dessen Inhalt. Es konzentriert sich in ihrer Verfügbarkeit, Integrität, Authentizität, Vertraulichkeit und Verbindlichkeit (NIST, 2020).

Dirbuster, Gobuster, usw. sind Anwendungen, die mit Brute Force, versuchen, Dateien und Verzeichnisse innerhalb Webanwendungen zu finden (KALI TOOLS, 2022).

Javascript ist ein Programmiersprache, die in Webanwendungen verwendet wird, um komplexe Strukturen wie Animationen, Bilder, Tun und Interaktionen zu implementieren. (Mozilla Corporation, 2022).

nmap (Network Mapper) ist eine Open-Source Anwendung für die Netzwerkanalyse. Mit diesem Tool ist es möglich, Hosts, Diensten (und deren Versionen) schnell zu entdecke(Nmap.org, 2021).

Pentester Auch Ethical Hacker genannt ist ein Sicherheitsanalyst, der sich damit beschäftigt, Schwachstellen von IT-Systemen zu finden (ProSec, 2019).

Port ist eine Zahl, die ein Dienst oder eine Verbindung identifiziert. Es geht hier um eine logische Adressierung zur Identifizierung eines oder mehrere Prozessen (Tanembaum and Wetherall, 2011).

Schwachstelle Schwäche eines Systems (Wendzel, 2018).

Scout Suite Audi-Tool für Sicherheitsüberprüfungen von Cloud-Umgebungen. Mit dem Tool werden Einstellungsinformationen gelesen und in einem lesbaren Dateiformat ausgegeben (nccgroup, 2022).

Tennant Einige Webanwendungen werden so konzipiert, dass verschiedene unabhängige Gruppe verwenden können. Z.B. ein Plattform für Online-Shop kann von verschiedenen Anbieter benutzt werden. Obwohl jeder Anbieter seine eigenen Namen, Marken, Produkte haben, benutzen beide nur einen Plattform. Jeder von diesem Anbieter nennen wir Tennants.

TheHive Project TheHive ist ein Open Source Plattform für die Verwaltung und Weiterarbeitung von Sicherheitsvorfälle. Es integriert andere Plattform und Anwendungen, wie Cortex, um Informationen und Handlungen bereitzustellen, damit die Arbeit von Security Operation Center auf einem Plattform konzentriert bleibt (Project, 2021).

Verwundbarkeit Oder als *vulnerability* gekannt. Es beschreibt eine von Angreifer ausnutzbare Schwachstelle (Wendzel, 2018).

Webanwendung Internetseite, die eine Interaktion ermöglichen. Diese Interaktion kann beispielsweise Login, Einkauf, Erstellung und Manipulation von Daten. Die meisten Webanwendungen sind an einem Datenbank verbunden. Webseite sind seinerseits statische Seite, dessen Inhalt nicht dynamisch aktualisiert sind (Essential Designs, 2019).

Abkürzungen

CIA Confidentiality, Integrity and Availability.

CVSS Common Vulnerability Scoring System.

DVWA Damn Vulnerable Web Application .

DVWA Cross-Site Scripting.

FPO Fachspezifische Prüfungsordnung.

NIST National Institute of Standards and Technology.

OSINT Open-source intelligence.

OWASP® Open Web Application Security Project®.

PoF Proof of Concept.

ROE Rules of Engagement.

1 Einleitung

Mein Praxissemester findet im Rahmen der Fachspezifische Prüfungsordnung (FPO) 2008 für den Studiengang Angewandte Informatik B.Sc. und dessen Modulhandbuch (Hochschule Worms, 2018) stat. Die Stellensuche orientierte sich auf dem Schwerpunkt Networks & Security und spezifischer auf Cybersicherheit und Penetration Testing.

Da es sich um einen spezifischen Bereich geht, war die Suche auf wenigen Firmen eingeschränkt, wo ich meine Bewerbungsunterlagen schickte. Die Firma Wallsec GmbH in Wiesloch-Walldorf hatte eine offene Stelle für Student. Da diese Stelle genau meinen Ziele entsprachen, bewarb ich mich für die Position. Die Aufgabe in der Stellenbeschreibung ging hauptsächlich um Durchführung von Penetrationstests, um Source-Code-Analyse, um Dokumentation Analyse und Erstellen, um Evaluation und Entwicklung von Sicherheitsprozessen. Als Voraussetzung verlangte Wallsec zwar wenig Fachkenntnis im Bereich Sicherheit und Penetration Testing, aber wollte große Interesse von den Kandidaten für das Lernen (Wallsec Security, 2022).

Das Bewerbungs- bis zum Einstiegsverfahren dauert ungefähr einen Monat. Am 15ten Juli 2022 fing ich an bei Wallsec als Praktikant im Vollzeit zu arbeiten. In diesem Bericht werden wir folgenden Themen bearbeiten:

- Informationen über die Firma Wallsec
- Konzepte von Penetration Testing
- Aufgabebereich der Tätigkeit
- Ergebnis des Praxissemesters

1.1 Wallsec

Die Firma wurde im Jahr 2020 von Peter Todorov in Walldorf-Wiesloch, Baden Württemberg, gegründet. Laut der Beschreibung der Webseite fokussierte sie auf die Planung,

Bereitstellung und Risikoanalysen von Sicherheits-Infrastrukturen von Firmen verschiedenen Größen (Wallsec Security, 2022). Wallsec bietet folgenden Leistungen an:

- Penetrationstests
- Schwachstellenmanagement
- Richtlinien
- Automatisierung
- DevOps und CI/CD Pipeline Sicherheit
- Beratung im Bereich Cyberabwehr

2 Anwendungsdomäne

Oft gibt es Nachrichten über Firmen oder Regierungen, deren Geheimnis im Netz von Angreifer veröffentlicht wurden oder deren Diensten wegen Cyberangriff unerreichbar sind. Da solche Situationen öfter als je vorkommen, ist das Interesse und die Forschung in dem Bereich Cybersicherheit in den letzten Jahren rasant gestiegen (Tanembaum, 2009).

Diese Angriffe verletzen die drei wichtigsten Ziele der Cybersicherheit, und zwar die Vertraulichkeit, die Integrität und die Verfügbarkeit (aus dem Englisch Confidentiality, Integrity and Availability (CIA)). Diese Zielen bekommen in den Fachliteraturen folgenden Beschreibungen: Schutz gegen unautorisierte Informationsgewinnung; Schutz gegen unautorisierte Datenmanipulation und Zugriffsgewährleistung für authentifizierte und autorisierte Subjekten (Wendzel, 2018). Ein Angriff zielt Schwachstelle oder Verwundbarkeit eines Systems. Diese wird dann zu einer Bedrohung, wenn es möglich ist, dieses System auszunutzen.

2.1 Theorie über Penetration Testing

Eine heutzutage sehr verwendete Methode, um Verwundbarkeit zu finden und zu analysieren ist durch Schwachstellenanalyse und Penetration Testing. Der National Institute of Standards and Technology (NIST) beschreibt das erste als systematische Analyse eines Systems oder Produktes in Bezug auf ihrer Sicherheit, um dessen Schwachstelle zu finden; und das zweite als Methode für die Verifizierung von binärischen Komponenten oder Anwendungen im Ganzen, um zu finden, ob dessen Verwundbarkeiten in Bezug auf ihre Daten oder Resources ausnutzbare sind (NIST, 2020). Man kann auch sagen, dass es bei der Schwachstellenanalyse eine Datenerhebung stattfindet, die später in der Penetration Testing in eine autorisierte Weise ausgenutzt wird(Goel and Mehtre, 2015). Das Ergebnis dieser zwei Prozessen werden später dem Beauftragter bekanntgemacht, damit Sicherheitsmaßnahmen genommen werden können.

Der Begriff ist auch als “Red Teaming”, als “Ethikal Hacking”, als “Pentest” oder als “white hats (weiße Huts)” bekannt. Es umfasst spezifische Analyse von Drohungen und von Verwundbarkeiten eines Produktes und es findet im Rahmen einen Vertrag oder ROE zwischen einen Kunde und die Firma oder Person statt, die für die Tests verantwortlich sind (Bishop, 2007).

2.2 Phase und Methodologie eines Penetration Testing

Ein Penetration Testing findet in einer systematischen Reihenfolge mit drei Hauptphasen: Vorbereitung, Implementation und Analysis (Shebli and Beheshti, 2018). Während der Vorbereitung werden der Umfang, die Ziele und den Dauer definiert. Bei der Implementation wird das System oder das Produkt in ihren Aufbau erkannt, analysiert und ausgenutzt (*exploited*). In der letzten Phase werden die gefundenen Verwundbarkeiten sämtliche Lösungsvorschlägen dem Beauftragter mitgeteilt. In manchen Fälle kann das getestete Objekt bezüglich seiner Sicherheit mithilfe der Common Vulnerability Scoring System (CVSS) bewertet werden. Diese Punktmechanismus stellt eine internationale anerkannte Evaluation eines Objekts dar.

Es gibt drei bekannte Methodologie, wo ein Penetration Testing stattfindet, und zwar *white box*, *black box* oder *zero-knowledge* und *grey box*. In der ersten Methodologie bekommen die Tester ausführliche Informationen über das getestet Objekt, wie Quellcode, interne Logik und Struktur. In der zweiten haben die Tester nur Open-source intelligence (OSINT) Informationen. Die dritte Variante ist eine Mischung aus den ersten zwei, in diesem Fall bekommen die Tester beschränkte Informationen über das zu testende Objekt (Ehmer and Khan, 2012).

2.3 Penetration Testing in Webanwendungen

Webanwendungen bieten ihren Nutzer eine dynamische und interaktive Umgebungen, ohne dass man Anwendungen in dem eigenen Rechner installieren muss. Im Vergleich

zum Desktop-Anwendungen erlauben Webanwendungen, dass mehrere Nutzer die Anwendungen gleichzeitig benutzen können, dass der Zugriff über verschiedene Plattformen, wie Handys, Desktop, Tablet und Laptops, stattfinden kann und dass die Wartungskosten niedriger werden, da die Hardwarekonfiguration nicht ständig aktualisiert werden muss(Techtarget, 2019).

Da die Häufigkeit der Transaktionen mit Webanwendungen ständig steigt, müssen Anbieter die Sicherheit dieser Anwendungen gewährleisten. Eine Organisation, die sich darum kümmert, über die Sicherheit in Webanwendungen zu recherchieren die Open Web Application Security Project®(OWASP®). Die Publikationen von der Organisation werden weltweit von Sicherheitsfirmen, Entwickler und Pentester verwendet, um die Tests durchzuführen.

Jährlich veröffentlicht OWASP® eine Liste mit den zehn häufigsten Angriffen in Webanwendungen und sichere Maßnahmen, um sie zu vermeiden. Die Organisation bietet auch eine eigene *Security Testing Guide* an, die die Arbeit von Penetration Testing unterstützt, um die Schwachstellen von Anwendungen zu finden, zu überprüfen und zu härten.

Innerhalb meines Praxissemesters spielten die Publikationen von der Organisation eine wichtige Rolle, um spezifische Kenntnisse zu erwerben und um meine Arbeitsweise an dem heutigen Anforderungen anzupassen.

3 Durchführung der Aufgabe

In diesem Kapitel beschreiben wir konkret, wie die Arbeit während meines Praxissemesters sich entwickelte. Die ersten zwei Wochen dienten als Einarbeitung und Einstieg. Nach dieser Phasen bekam ich langsam und unter Betreuung mehr Verantwortlichkeit und mehr Freiheit, um die Arbeit durchzuführen. In der folgenden Tabellen wird der Ablauf systematisch und ohne Einzelheit beschrieben. In dem zweiten Teil dieses Kapitels geben wir eine ausführliche Beschreibung eines Projekts.

Jedes Projekt besitzt innerhalb von Wallsec einen festgelegten Aufbau. Dieser kann in den folgenden Punkten zusammengefasst werden:

1. *Kick-off Meeting* mit den Kunden, um grundsätzliche Information über die Anwendung zu bekommen

2. Definition der Umfang des Tests, wie Anmeldedaten, Rolle der zu getestete Nutzer, Tennants und Einschränkungen

3. Durchführung von Tests nach einem vorgegebenen Checklist

4. Dokumentation der durchgeführten Testen, dessen gefundene Schwachstellen und Vorschläge zur Härtung der Anwendung

5. Abschlussmeeting mit dem Kunden, um die Schwachstellen und deren Ausnutzung zu präsentieren und zu demonstrieren

4 Wöchentliche Zusammenfassung meines Praxissemesters

Auflistung der Aufgabe	
Woche	**Aufgabeschreibung**
1 - 2	Einarbeitung: • Installation von einer virtuellen Maschine für die Testumgebungen • Einführung in der Arbeitsablauf der Firma • Einführung, Installation und Einstellungen von Burp Suite • Einführung in einem laufenden Projekt, um über das Ablauf- und Dokumentationsverfahren zu lernen • Durchführung und Wiederholungen von einigen Tests, um mich an den gegebenen Tools zu gewöhnen • Teilnahmen an einer Abschlussmeeting des laufenden Projekts, um das Verfahren und den Ablauf des Kundenkontakt zu erkennen und später zu wiederholen
3 - 4	Start, Durchführung und Abschluss eines neuen Pentesting-Projekts an einem Versicherungsanwendung mit dem obigen beschriebenen Schritte (3)
5 - 6	Weiterarbeitung an der Installation, an der Einstellungen und an der Nutzung der Tools TheHive Project und Cortex. Bereitstellung von Skripts zum Herunterladen von statistische Daten der Anwendungen und zur Automatisierung deren Nutzung.
7 - 8	Start, Durchführung und Abschluss eines neuen Pentesting-Projekts an einem Marketing-Webanwendung mit dem obigen beschriebenen Schritte (3)
9 - 10	Start, Durchführung und Abschluss eines neuen Pentesting-Projekts an Netzwerk-Umgebungen mit dem obigen beschriebenen Schritte (3). Die durchgeführten Tests konzentrieren sich auf die Sicherheit einer Netzwerk in einer Cloud-Umgebung. Für dieses Projekt spielen die Tools nmap und Scout Suite eine wichtige Rolle, da das Ziel war, Hosts, Dienste und dessen Einstellungen und Schwachstelle zu erkennen
13 - 14	Start, Durchführung und Abschluss eines Pentesting bei einem umfangreichen Webanwendung mit verschiedenen Tennants und Nutzungsrolle für die Verwaltung von Business-Processes.
15 - 16	xxxxxx
17 - 18	xxxxxx
19 - 20	xxxxxx

5 Ausführliche Beschreibung eines Penetration Testing innerhalb des Praxissemesters

Bevor der Durchführung jedes Tests ist Wallsec und ihre Mitarbeiter dazu verpflichtet, eine Vertraulichkeitserklärung zu unterschreiben. Keine Information weder über die Firma noch über die verwendeten Methode dürfen aus irgendwelcher Form veröffentlicht werden. Aus diesem Grund werden die hier demonstrierten Methode und Tests in der Test- und Lernumgebung Damn Vulnerable Web Application (DVWA) gezeigt. In den realen Tests verwenden wir ähnliche Methode, manchmal mit mehr oder weniger Details, um die Sicherheit der Anwendungen zu überprüfen.

Da dieses Bericht eingeschränkte Platz hat und das dieses Thema sehr umfangreich hier, demonstrieren wir in den nächsten zwei Unterkapitel einige Methode, die wir verwenden, um die Zielanwendung in ihrer Struktur zu kennen und auszunutzen.

5.1 Sammlung von Informationen von dem Zielsystem

Obwohl jede Webanwendungen ihre eigene Eingenschaften und Ziele haben, besitzen fast alle eine ähnliche Struktur und Aufbau. Beim jeden Test fangen wir damit an, diese gemeinsame Struktur zu erkennen, indem wir nach öffentlichen Informationen suchen. Viele kritische Informationen, wie Username, Passwörter, Versionen, Systemen verbundenen IP-Adressen, lassen sich sehr leicht nach einer Online-Suche finden. Auch mit eingebauten Tools eines Betriebssystems können wir auf solche Informationen zugreifen. Dieses Verfahren nennen wir Banner Grabbing. Das folgende Bild zeigt ein Beispiel von einer einfachen Durchführung von Banner Grabbing:

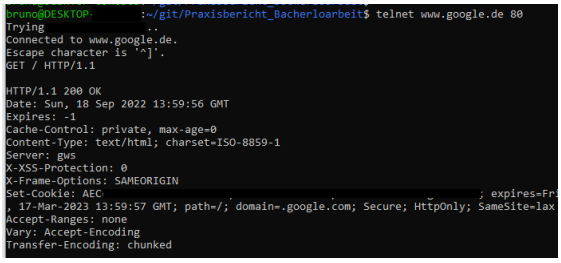

Abbildung 1: Banner Grabbing mithilfe von dem Tool telnet

Eine Webanwendung ist eine Gruppierung von verschiedene Verzeichnisse. Jedes Verzeichnis soll den Nutzer eine Information oder Interaktion anbieten. Manche sind aber nicht für Nutzer gedacht und dient hauptsächlich dazu, Einstellungen zu verwalten. Da solche Verzeichnisse nicht dazu konzipiert, direkt aufrufbar zu sein, benutzen wir Testers andere Methode, um zu finden, was die Entwickler im Hintergrund beibehalten wollte. Es gibt verschiedene Tools, die mithilfe von sogenannten *wordlists*, viele Anfrage an eine Anwendung schickt, um herauszufinden, was nicht direkt von dem Browser aufrufbar ist. Solche *wordlists* sind Textdateien, die häufige verwendete Wörter beinhalten, die für Webanwendungen, für Nutzername oder für Passwörter verwendet werden. Da viele Webanwendungen ähnliche Strukturen haben, ist auch meistens erwartet, dass gewöhnliche Wörter auch zu finden sind. Das nächste Beispiel zeigt uns, dieses Entdeckungsverfahren auf unserem Ziel, DVWA Tool. Hier benutzen wir das Tool Dirbuster, Gobuster, usw., um herauszufinden, welche Verzeichnisse in dieser Anwendung existieren. Dieses und andere Tool arbeiten mit der Brute-Force Methode. In diesem Fall werden verschiedene Anfrage geschickt, jede mit einem verschiedenen Wort, um zu sehen, welche positive Antworten liefern. Das folgende Bild zeigt die Durchführung und das Ergebnis des Scanverfahren mithilfe des Tools Dirbuster, Gobuster, usw.:

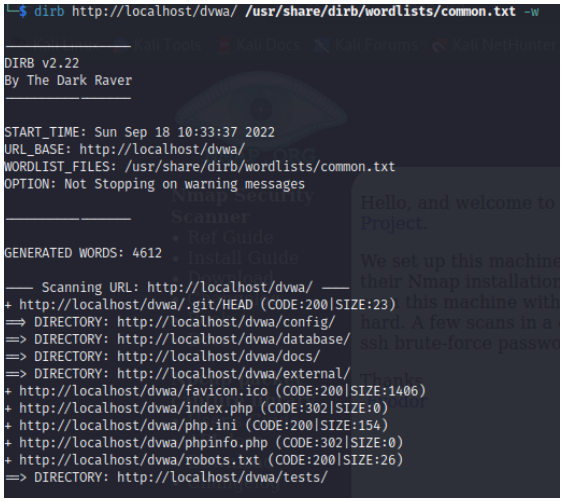

Abbildung 2: Brute force Scan für Verzeichnisentdeckung

Der nächste Schritte wäre eine manuelle Beobachtung der entdeckten Material, um zu finden, ob irgendwelche sensitiven Information ausgelieferte wurde. Falls ja, wurden wir dann versuchen diese Schwachstelle, zu erkennen und auszunutzen.

Ein Netzwerk-Scan ist auch eine häufige verwendete Methode, um zu finden, auf welche Dienste oder Anwendungen unser Ziel gebaut ist. Dieses Scan schickt an dem Ziel verschiedene Anfrage, damit wir das Reaktion des Zielsystems beobachten können. Während bei dem ersten Scan wir uns auf der Webanwendung fokussierten, hier bearbeiten wir eine Ebene, die nicht für normale Nutzung gezielt ist. Unser Fokus hier liegt auf dem Server, wo die Anwendung läuft. Dafür testen wir die sogenannte Port. Aus diesem Scan lassen sich meistens viele nützliche Informationen herausfinden, wie Betriebssystems, wo die Webanwendung läuft, Name und Versionen der existierenden Dienste. Mit dieser Information ist es möglich dann zielgerichtete Angriffe vorzubereiten, um Schwachstellen auszunutzen.

Das nächste Bild zeigt das Ergebnis der Durchführung von nmap gegen das Testziel *scanme.nmap.org*:

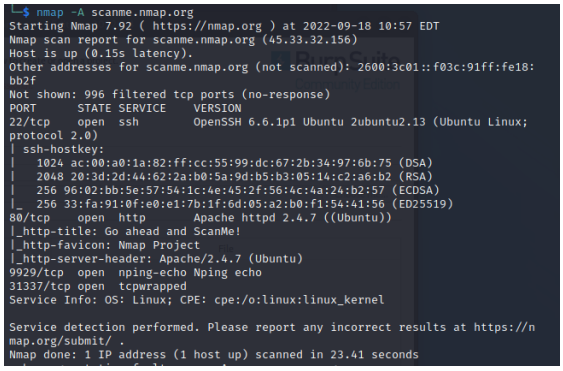

Abbildung 3: Brute force Scan für Verzeichnisentdeckung

Aus diesem Scan erfahren wir welche Betriebssystem und welche Version der Webanwendung benutzt werden. Auch wenn solche Versionen gegen Angriffe geschützt sind, ist es unsere Aufgabe dem Kunde zu informieren, dass sensitive Informationen für alle sichtbar sind. Eine böse absichtliche Nutzer könnte diese Information nutzen, um eine Schwachstellen für diese Anwendungen zu erfinden und dann auszunutzen.

5.2 Ausnutzung der Zielanwendung

Nachdem die vorherigen Scans durchgeführt wurde und öffentliche und Serverseite Informationen gesammelt wurden, fangen wir damit an, die Webanwendung direkt zu testen. In diesem Fall ist es unser Ziel zu wissen, welche versteckte Daten oder unerlaubte Aktionen ein Angreifer durchführen kann, um die CIA der Anwendung zu verletzen. Für die folgenden Tests benutzen wir unter anderen auch das Tool Burp Suite.

Unser erster Test will überprüfen, ob es möglich ist, in ein Eingabefeld Daten einzutragen und das normale Verhalten der Anwendung zu ändern. Wir wollen eigenen Code hinzuführen und in dem Falle, dass es uns gelingt, das zu tun, können wir dann weitere Code hinzufügen, um Daten von Nutzer zu stehlen oder das normale Verhalten der Anwendung beschädigen. Wir prüfen hier, ob die gegen Cross-Site Scripting (XSS) anfällig ist. Um diesen Test durchzuführen, fügen wir erwartete Daten hinzu, um das Verhalten der Anwendung zu beobachten. Nachdem wir das normale Verhalten erkannt haben, versuchen

wir eigenen Code hinzufügen und beobachten, ob wir die Anwendung ausnutzen können.

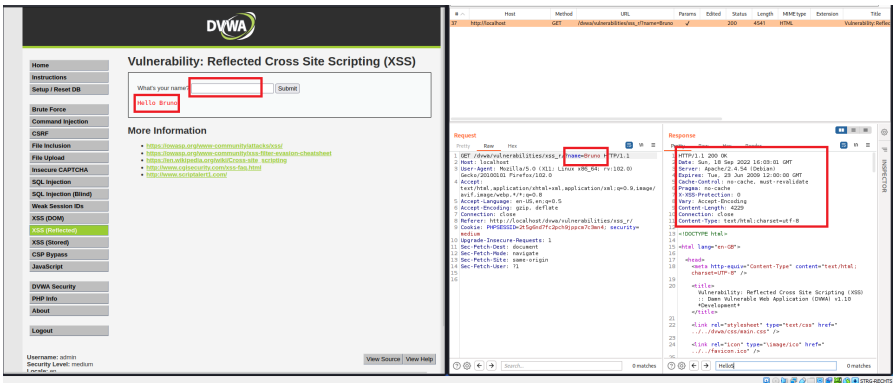

Abbildung 4: Beobachtung der Anwendung unter normale Nutzung

Das Bild 4 zeigt den ersten Test. Aus dieser Aufnahme der Anfrage können wir sehen, sehen, dass die Nutzereingabe direkt in dem Browser stattfindet. Wir sehen in der Antwort Informationen über den Aufbau der Anwendung und wie sie auf unsere Anfrage reagiert. Auf dem nächsten Bild versuchten wir einige bösartige Code hinzufügen, um den normalen Ablauf der Anwendung zu verletzten. Dafür verwenden Javascript Code. Wir können unseren Code direkt in die Anwendung, in eine selbst gebastelte Request oder in Burp Suite eingabe:

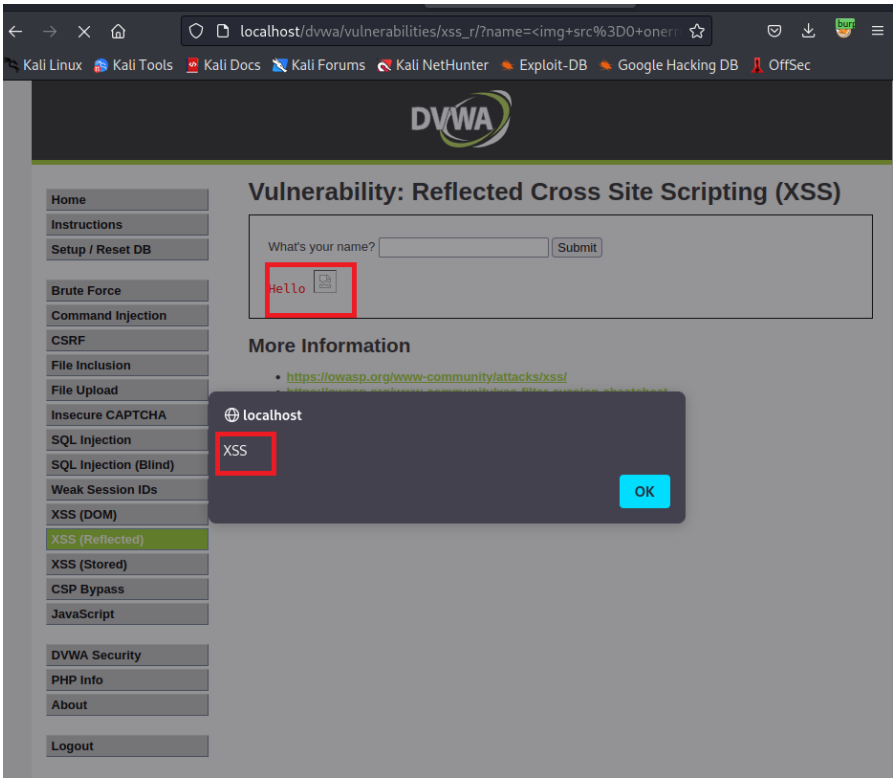

Abbildung 5: Einführung von bösartigen Code und Beobachtung der Reaktion der Anwendung.

Für diesen Test haben wir den Code *<img src=0 onerror='alert("XXS")'\>* hinzugefügt. Das Ziel dieses Codes ist ein nicht existierende Bild hinzufügen, um einen absichtlichen Fehler zu provozieren. Dieser Fehler zeigt ein kleines Fenster in der Anwendung mit dem Text "'XSS"'. Eine geschützte Anwendung würde entweder den Code und ihren Zeichen "< >" ignorieren oder diese zu anderen übersetzen. Es kann auch sein, dass die Anwendung den Nutzer sagt, dass die eingegebenen Zeichen nicht erlaubt nicht. Aus dem Bild sehen wir aber, dass die Anwendung alle Zeichen akzeptiert und sogar erlaubt, dass der Code ausgeführt wird. Aus dieser Situation hätten wir einen Proof of Concept (PoF), dass die Anwendung gegen diese Art von Angriff anfällig ist.

5.3 Kundebericht

Je nachdem wie lange das Projekt läuft, können wir mehr oder weniger zeitintensive Tests durchführen. Am Ende des Projekts präsentieren wir unseren Kunde in einem Meeting unser Ergebnis und liefern wir ein Bericht mit detaillierte Informationen über die gefundenen Schwachstellen und die dazu verwendete Methode, um sie zu finden. Dieses Bericht wird so geschrieben, damit auch solche, die nicht aus dem Sicherheitsbereich hingehören verstehen können.

In den ersten Abschnitten erklären wir mit weniger technischen Begriffe, wie was für Tests durchgeführt wurden. In den folgenden Kapitel erklären wir mit mehr Einzelheiten und technische Details, wie wir zu unserem Ergebnis kamen. Anschließend geben wir Vorschläge für die Verbesserung der Sicherheit der Webanwendung und am Ende geben wir mithilfe von CVSS oder von den Kunden ausgewählten Metrik eine allgemeine Bewertung.

6 Fazit

Das Praxissemesters bat mir eine sehr gute Möglichkeit an, die theoretische Kenntnisse der Uni in der Praxis anzuwenden. Zusätzlich konnte ich weitere und tiefere Kenntnisse in dem Bereich Sicherheit und Penetration Testing entwickeln. Alle Module meines Studiums haben dazu beitragen, dass ich mich in den Sicherheitsthemen vertiefen kann. Sie gaben mir eine solide Grundlage und die richtigen Soft Skills für die Weiterentdeckung des Sicherheitsbereiches Penetration Testings.

Die entwickelten Eigenschaften werden meine zukünftige Berufsleben prägen und werden viel Bedeutung für die Durchführung einer hochwertigen Arbeit sein. Da aber diese Kenntnisse sich hier nicht ausschließen, stehe ich metaphorisch nur am Anfang eines Entdeckungsweges, der mir herbeiführt, eine Karriere in Ethical Hacking aufzubauen.

Literaturverzeichnis

Bishop, M. (2007). About penetration testing. *IEEE Security & Privacy*, 5(6):84–87.

DVWA TEAM (2016). Damn vulnerable web application. https://github.com/digininja/DVWA. Zugriff am 18te September 2022.

Ehmer, M. and Khan, F. (2012). A comparative study of white box, black box and grey box testing techniques. *International Journal of Advanced Computer Science and Applications*, 3. http://dx.doi.org/10.14569/IJACSA.2012.030603. Zugriff am 31 Juni 2022.

Essential Designs (2019). Website vs web app: What's the difference? https://www.triaxiomsecurity.com/rules-of-engagement-important-to-penetration-test/. Zugriff am 7te August 2022.

Goel, J. N. and Mehtre, B. (2015). Vulnerability assessment & penetration testing as a cyber defence technology. *Procedia Computer Science*, 57:710–715. https://www.sciencedirect.com/science/article/pii/S1877050915019870/. Zugriff am 31 Juni 2022.

Healthcare Computing (2021). Was ist bzw. tut das national institute of standards and technology (nist)? https://www.healthcare-computing.de/was-ist-bzw-tut-das-national-institute-of-standards-and-technology-nist-a-1022210/. Zugriff am 31 Juni 2022.

Hochschule Worms (2018). Fachspezifische prüfungsordnung (fpo 2018). https://www.hs-worms.de/fileadmin/media/fachbereiche/informatik/AInf/Pruefungsordnung/AnInf_FPO_2017-12-19_FINAL.pdf. Zugriff am 31 Juni 2022.

Junmei, W. and YanChengkang (2021). Automation testing of software security based on burpsuite. In *2021 International Conference of Social Computing and Digital Economy (ICSCDE)*, pages 71–74. https://doi.org/10.1109/ICSCDE54196.2021.00025. Zugriff am 7te August 2022.

KALI TOOLS (2022). dirb, gobuster. https://gitlab.com/kalilinux/packages/dirbuster. Zugriff am 18te September 2022.

Mahmoud, S. K., Alfonse, M., Roushdy, M. I., and Salem, A.-B. M. (2017). A comparative analysis of cross site scripting (xss) detecting and defensive techniques. In *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pages 36–42. https://doi.org/10.1109/INTELCIS.2017.8260024. Zugriff am 18te September 2022.

Malwarebytes (2022). Proof of concept. https://www.malwarebytes.com/glossary/proof-of-concept#:~:text=A%20proof%20of%20concept%20(PoC,12th%20Floor. Zugriff am 18te September 2022.

Mozilla Corporation (2022). What is javascript?

https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript. Zugriff am 18te September 2022.

nccgroup (2022). Scoutsuite.
https://github.com/nccgroup/ScoutSuite. Zugriff am 11 September 2022.

NIST (2020). Cyber attacke.
https://csrc.nist.gov/glossary/term/Cyber_Attack. Zugriff am 31 Juni 2022.

Nmap.org (2021). Nmap-referenz-handbuch (man page)m.
https://nmap.org/man/de/index.html. Zugriff am 21 August 2022.

Openvpn (2022). How cybersecurity has changed in the last decade.
https://openvpn.net/blog/how-cybersecurity-has-changed-in-the-last-decade/. Zugriff am 31 Juni 2022.

OWASP (2001). Who is the owasp® foundation?
https://owasp.org/. Zugriff am 7te August 2022.

Project, T. (2021). Thehive - a 4-in-1 security incident response platform.
https://thehive-project.org/. Zugriff am 14 August 2022.

ProSec (2019). Der job als penetration tester.
https://www.prosec-networks.com/blog/der-job-als-penetration-tester/. Zugriff am 7te August 2022.

Security Insider (2019). Was ist cvss?
https://www.security-insider.de/was-ist-cvss-a-853465/. Zugriff am 31 Juni 2022.

Shebli, H. M. Z. A. and Beheshti, B. D. (2018). A study on penetration testing process and tools. In *2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, pages 1–7.
https://doi.org/10.1109/LISAT.2018.8378035. Zugriff am 31 Juni 2022.

Tanembaum, A. S. (2009). *Moderne Betriebssysteme*. Pearson, München.

Tanembaum, A. S. and Wetherall, D. (2011). *Computer Networks*. Prentice Hall, München, 5 edition.

Techtarget (2019). Web application (web app).
https://www.techtarget.com/searchsoftwarequality/definition/Web-application-Web-app. Zugriff am 7te August 2022.

Triaxiom Security (2018). Why are rules of engagement important to my penetration test?
https://www.triaxiomsecurity.com/rules-of-engagement-important-to-penetration-test/. Zugriff am 31 Juni 2022.

Wallsec Security (2022). About us.
https://www.wallsec.de. Zugriff am 31 Juni 2022.

Wendzel, S. (2018). *IT-Sicherheit für TCP/IP- und IoT-Netzwerke*. Springer Vieweg, Wiesbaden.

Wendzel, S. and Plötner, J. (2007). *Praxisbuch Netzwerk-Sicherheit: Risikoanalyse, Methoden und Umsetzung; für Unix-Linux und Windows; VPN, WLAN, Intrusion Detection, Disaster Recovery, Kryptologie*. Galileo Computing, Bonn.

Yeboah-Ofori, A. (2018). Cyber intelligence and osint: Developing mitigation techniques against cybercrime threats on social media. *International Journal of Cyber-Security and Digital Forensics*, 7:87–98.
`http://dx.doi.org/10.17781/P002378`. Zugriff am 31 Juni 2022.