YÖNETİM BİLİŞİM SİSTEMLERİ DERGİSİ
http://dergipark.org.tr/ybs

Yayın Geliş Tarihi: 30.01.2025
Yayına Kabul Tarihi: 10.06.2025
Online Yayın Tarihi: 30.06.2025

Cilt: 11, Sayı: 1, Yıl: 2025, Sayfa: 62-80
ISSN: 2630-550X

A RISK ASSESSMENT ON THE USAGE OF KALI TOOLS TO HACK AND MANIPULATE WEB-BASED MIS AND ERP APPLICATIONS

Ahmet EFE[1]

[1] Independent Researcher, Ankara, Türkiye

ABSTRACT

The increasing reliance on web-based Management Information Systems (MIS) and Enterprise Resource Planning (ERP) applications has made them an attractive target for cyber attackers. This study conducts a comprehensive risk assessment of the use of Kali Linux tools in hacking and manipulating web-based MIS and ERP applications. The study provides an in-depth analysis of prominent Kali Linux tools such as SQLMap, Burp Suite, Metasploit Framework, Nmap, and Nessus, which are commonly used for security testing but also pose significant risks when leveraged for malicious activities. Drawing on case studies and existing literature, the findings underscore the critical security gaps in web-based MIS and ERP applications, emphasizing the need for robust defense mechanisms. This study highlights the critical security vulnerabilities present in web-based MIS and ERP systems and demonstrates, through expert-guided theoretical analysis, how proactive risk mitigation strategies can significantly strengthen organizational cybersecurity posture. The study proposes proactive risk mitigation strategies, including regular security audits, implementation of least privilege access controls, security awareness training, deployment of advanced threat detection systems, and adherence to legal and compliance frameworks governing penetration testing. The research concludes that while Kali Linux serves as a valuable tool for ethical hacking and security assessments, its misuse with the support of AI algorithms and automated code generations of scanning and attacks necessitates a stringent cybersecurity framework to protect organizational assets. Future research should explore the integration of automated threat detection systems and the legal implications of penetration testing to enhance cybersecurity resilience.

Keywords: Kali Linux, Web-Based Applications, MIS, ERP, Hacking, Vulnerabilities, Cyber Risks

KALI ARAÇLARININ WEB TABANLI MIS VE ERP UYGULAMALARINI HACKLEME VE MANİPÜLE ETME AMACIYLA KULLANIMINA YÖNELİK BİR RİSK DEĞERLENDİRMESİ

ÖZET

Web tabanlı Yönetim Bilişim Sistemleri (MIS) ve Kurumsal Kaynak Planlama (ERP) uygulamalarına olan bağımlılığın artması, bu sistemleri siber saldırganlar için cazip bir hedef haline getirmiştir. Bu çalışma, Kali Linux araçlarının web tabanlı MIS ve ERP uygulamalarını hackleme ve manipüle etme amacıyla kullanılmasına yönelik kapsamlı bir risk değerlendirmesi yapmaktadır. Araştırma kapsamında SQLMap, Burp Suite, Metasploit Framework, Nmap ve Nessus gibi öne çıkan Kali Linux araçlarının detaylı bir analizi yapılmıştır. Bu araçlar, güvenlik testleri için yaygın olarak kullanılmakla birlikte, kötü niyetli faaliyetler için kullanıldığında önemli riskler teşkil etmektedir. Çalışmada vaka analizleri ve mevcut literatürden yararlanılarak, web tabanlı MIS ve ERP uygulamalarındaki kritik güvenlik açıkları ortaya konmuş ve güçlü savunma mekanizmalarının gerekliliği vurgulanmıştır. Bu çalışma, web tabanlı MIS ve ERP sistemlerinde mevcut olan kritik güvenlik açıklarını ortaya koymakta ve uzman rehberliğinde yürütülen teorik analizler aracılığıyla, proaktif risk azaltma stratejilerinin kurumsal siber güvenlik duruşunu önemli ölçüde güçlendirebileceğini göstermektedir. Düzenli güvenlik denetimleri, en az ayrıcalık ilkesine dayalı erişim kontrollerinin uygulanması, güvenlik farkındalığı eğitimi, ileri tehdit tespit sistemlerinin devreye alınması ve sızma testlerini düzenleyen yasal ve uyum çerçevelerine riayet edilmesi gibi proaktif risk azaltma stratejileri önerilmektedir. Araştırma, Kali Linux'un etik hackleme ve güvenlik değerlendirmeleri için önemli bir araç olduğunu, ancak yapay zekâ algoritmalarının desteğiyle tarama ve saldırı süreçlerinin otomatikleştirilmesiyle kötüye kullanım riskinin arttığını ortaya koymaktadır. Sonuç olarak, kurumsal varlıkları korumak için katı bir siber güvenlik çerçevesinin benimsenmesi gerektiği vurgulanmaktadır. Gelecek çalışmaların, otomatik tehdit tespit sistemlerinin entegrasyonu ve sızma testlerinin hukuki boyutlarını ele alarak siber güvenlik dayanıklılığını artırmaya odaklanması önerilmektedir.

Anahtar Kelimeler: Kali Linux, Web Tabanlı Uygulamalar, Yönetim Bilişim Sistemleri (YBS), Kurumsal Kaynak Planlaması (KKP), Hackerlık, Güvenlik Açıkları, Siber Riskler

1. INTRODUCTION

The increasing integration of Management Information Systems (MIS) within contemporary organizations has driven the widespread adoption of web-based applications for storing, processing, and disseminating critical information. However, this growing dependence on web-based MIS platforms has simultaneously heightened the risk of malicious attacks and unauthorized access, posing serious threats to the confidentiality, integrity, and availability of organizational data. This study seeks to investigate the potential vulnerabilities inherent in these systems and demonstrate how tools available within the Kali Linux distribution can be employed to exploit and manipulate them, while leaving the analysis of AI-driven defense mechanisms to future research.

MIS are computerized systems that provide organizations with essential information to aid in the management and decision-making processes (Laudon & Laudon, 2004). These systems integrate data from various sources, process it, and present it in an accessible and meaningful way to facilitate planning, controlling, and decision-making activities within organizations (Oz, 2008).

MIS tools focus on data analysis and reporting, enabling organizations to make data-driven decisions, while ERP applications integrate various business functions to ensure smooth operations across departments. Below is a table of reputable MIS and ERP applications available in the market as of September 2021, along with their respective companies and websites. Please note that there may be newer or updated applications available that are not included in this list. It is essential to research and evaluate each product based on your organization's unique needs before making a decision.

The increasing reliance on web-based MIS and ERP applications exposes organizations to evolving cyber threats. These applications manage critical business processes and store sensitive data, making them attractive targets for attackers. Kali Linux, an advanced penetration testing and security auditing toolset, contains numerous tools that can be leveraged to exploit vulnerabilities in these systems. This study aims to assess the risks associated with the use of Kali Linux tools in hacking and manipulating web-based MIS and ERP applications. By systematically analyzing various attack techniques—reconnaissance, scanning, enumeration, exploitation, and post-exploitation—the study will highlight the ease with which attackers can exploit vulnerabilities in these systems. The findings will contribute to a deeper understanding of cyber risks and inform risk mitigation strategies to enhance the security posture of organizations relying on web-based MIS and ERP applications.

We have identified the research assumptions as such:

1. The study assumes that an attacker with basic cybersecurity knowledge can access and utilize Kali Linux tools to exploit vulnerabilities in MIS and ERP applications.

2. Many web-based MIS and ERP applications contain known vulnerabilities that can be identified through databases such as Exploit-DB, NIST NVD, and CVE Details.
3. Attack methodologies involving reconnaissance, scanning, enumeration, exploitation, and post-exploitation are assumed to be effective in compromising security mechanisms of these applications.
4. Many organizations lack robust controls, such as regular patching, intrusion detection systems (IDS), web application firewalls (WAFs), and strong authentication mechanisms, making them susceptible to Kali Linux-based attacks.
5. The study assumes that all penetration testing activities conducted for analysis will be performed in a controlled and ethical testing environment, without violating legal or organizational policies.

Table 1: Information on the Most Reputable Web-Based MIS and ERP Systems

Application Name	Type	Company	Website
SAP S/4HANA	ERP	SAP	https://www.sap.com/products/s4hana-erp.html
Oracle NetSuite	ERP	Oracle	https://www.netsuite.com
Microsoft Dynamics 365	ERP	Microsoft	https://dynamics.microsoft.com/en-us/
Infor CloudSuite	ERP	Infor	https://www.infor.com/products/cloudsuite
Epicor ERP	ERP	Epicor	https://www.epicor.com/en-us/erp-systems/epicor-erp/
Sage X3	ERP	Sage	https://www.sage.com/en-us/products/sage-x3/
Odoo	ERP	Odoo	https://www.odoo.com
Acumatica Cloud ERP	ERP	Acumatica	https://www.acumatica.com/cloud-erp-software/
SYSPRO	ERP	SYSPRO	https://www.syspro.com/erp-software/
Deltek Costpoint	ERP	Deltek	https://www.deltek.com/en/products/project-erp/costpoint
Zoho One	MIS/ ERP	Zoho	https://www.zoho.com/one/
Tableau	MIS	Salesforce	https://www.tableau.com
Power BI	MIS	Microsoft	https://powerbi.microsoft.com/en-us/
Qlik	MIS	Qlik	https://www.qlik.com
Sisense	MIS	Sisense	https://www.sisense.com

In line with the above-mentioned assumptions, we have 2 hypotheses to ascertain**:**

H1: Web-based MIS and ERP applications contain exploitable vulnerabilities that can be identified and leveraged using Kali Linux tools.

H2**:** Implementing proactive risk mitigation strategies, including regular security patching, network segmentation, and real-time threat monitoring, can significantly reduce the risks associated with the exploitation of MIS and ERP vulnerabilities.

To test the first hypothesis (H1)—that web-based MIS and ERP applications contain exploitable vulnerabilities that can be identified and leveraged using Kali Linux tools—this study adopts a structured penetration testing methodology grounded in widely recognized cybersecurity practices. The analysis systematically applies the five phases of ethical hacking: reconnaissance, scanning, enumeration, exploitation, and post-exploitation. Each phase is implemented using specialized tools available within the Kali Linux distribution, including SQLMap, Metasploit, Burp Suite, and Nmap, among others.

The research draws on publicly accessible vulnerability databases such as Exploit-DB, NIST National Vulnerability Database (NVD), and CVE Details to identify known weaknesses in widely used MIS and ERP platforms. These databases offer critical insights into existing software flaws, thereby supporting realistic test scenarios during the penetration testing exercises. Through a series of controlled case studies—such as exploiting SQL injection flaws, enabling remote code execution, and triggering cross-site scripting (XSS) vulnerabilities—the study demonstrates the feasibility of compromising real-world MIS and ERP applications using Kali Linux tools under simulated adversarial conditions.

To address the second hypothesis (H2)—that implementing proactive risk mitigation strategies can significantly reduce exploitation risks—the study reviews and contrasts the security configurations of target systems before and after applying layered defenses. This includes analyzing the impact of regular patch management, intrusion detection systems (IDS), web application firewalls (WAFs), multi-factor authentication, and network segmentation. By documenting the effectiveness of these controls in either preventing or mitigating successful attacks, the study presents evidence of how organizations can enhance their security posture.

This hypothesis-testing framework not only validates the technical exploitability of MIS and ERP systems (H1) but also provides empirical insight into the efficacy of standard and advanced defense mechanisms (H2). The findings underscore the critical need for ethical hacking, continuous vulnerability management, and cybersecurity training as integral components of organizational risk management strategies.

2. LITERATURE DISCUSSIONS

The dual-use nature of penetration testing tools in Kali Linux, often employed for both ethical and malicious purposes, raises urgent questions around their risks when deployed against such systems. The proliferation of artificial intelligence (AI)-enhanced cyber tools compounds these risks by automating, optimizing, and personalizing attack vectors.

Kali Linux, a Debian-based penetration testing distribution, is extensively used in both professional security testing and malicious hacking scenarios. Numerous studies illustrate how it serves as a launchpad for Distributed Denial of Service (DDoS), SQL injection, backdoor access, and privilege escalation attacks on various platforms (Khan et al., 2022; Pamarthi, 2020). Tools like Metasploit, SQLMap, and Aircrack-ng within Kali are instrumental in simulating or conducting real-time attacks on ERP interfaces and databases.

Khan et al. (2022) demonstrate that the DDoS attacks initiated from Kali environments generate observable impacts on system load and network latency, which can be visualized through traffic flow in network maps. Such empirical insights highlight how easily a network’s operational integrity can be compromised using Kali tools in targeted scenarios against web-based services.

Recent advancements in AI have dramatically shifted the landscape of penetration testing and cyber threat simulation. Studies show that machine learning models trained on attack datasets, when integrated into Kali Linux environments, can predict vulnerabilities and autonomously exploit them (Moorthy & Nathiya, 2023; Ordoñez & Guerra, 2018). He et al. (2023) particularly emphasizes how AI-based ethical hacking methods using Kali tools can simulate real-world attacks on health information systems—systems analogous in complexity and data sensitivity to ERP systems. This convergence of AI and hacking capabilities enables not only faster but also more nuanced cyberattacks that adapt in real-time, complicating detection and response strategies.

As hacking tools become smarter, defensive mechanisms are also evolving. AI-driven intrusion detection systems (IDS), especially those using neural networks or evolutionary algorithms, attempt to counteract the risks posed by AI-enhanced attacks from platforms like Kali Linux (Moustafa, 2022; Ciric et al., 2024). However, these systems are still vulnerable to adversarial inputs and self-poisoning data strategies, where attackers feed corrupted data into AI systems to mislead detection algorithms (Mahmood et al., 2024). This arms race suggests that while AI can strengthen defense mechanisms, it simultaneously enhances offensive capabilities, particularly when attackers use tools like Metasploit (Wang & Johnson, 2024) to automate reconnaissance and exploit phases against enterprise systems.

Ethical hacking simulations, such as those conducted using vulnerable systems like Metasploitable or VulnHub environments (Martínez et al., 2025), provide critical insights into potential attack paths and defense strategies. However, the blurred boundary between simulated and live systems, especially when practitioners misuse educational tools in production environments, presents a major risk. For instance, Nilă et al. (2021) and Zhuravchak et al. (2024) demonstrate the effectiveness of tools like CoWPAtty and malware simulation techniques in controlled environments, which, if replicated maliciously in enterprise settings, can have disastrous consequences. The ease of transferring exploits from a virtual machine to real ERP systems amplifies this risk.

MIS and ERP platforms are particularly susceptible to attacks involving SQL injections, session hijacking, and remote code execution—vectors frequently demonstrated using Kali Linux (Herman et al., 2023; Pamarthi, 2020). These applications often run with high privileges and contain sensitive organizational data, making successful exploitation both impactful and difficult to recover from. Advanced privilege escalation strategies using reinforcement learning (Kujanpää et al., 2021) further complicate security, especially when coupled with poor access control policies in MIS/ERP configurations.

Kali Linux is a widely adopted open-source distribution designed for penetration testing, security auditing, and ethical hacking (Najera-Gutierrez & Ansari, 2018). The operating system provides a comprehensive suite of tools for security professionals to identify vulnerabilities in systems, including Web-Based MIS and ERP applications. These tools facilitate ethical hacking activities but also introduce significant risks when leveraged for malicious purposes (Messier, 2024).

MIS and ERP applications are critical to business operations, as they manage key processes such as finance, supply chain, and human resources. However, these systems are often targeted by cybercriminals due to their extensive data repositories and integrations with multiple third-party applications (Velu & Beggs, 2019). Penetration testing with Kali Linux exposes these systems' vulnerabilities, enabling organizations to address security flaws before attackers can exploit them (Johansen, Allen, Heriyanto, & Ali, 2016). However, unauthorized usage of these tools poses a serious threat, leading to data breaches and system manipulation (Singh, 2019).

Kali Linux includes a variety of tools specifically designed for web application security testing. Some of the most notable ones include:

- SQLMap: Used for SQL injection attacks, allowing unauthorized access to databases (Akhtar & Rawol, 2024).
- Burp Suite: Assists in security testing of web applications, identifying vulnerabilities in authentication and session management (Velu, 2022).
- Metasploit Framework: Provides a structured approach to exploiting security weaknesses in web-based applications (Tabassum et al., 2021).
- Nmap and Nessus: Essential for network and vulnerability scanning, identifying open ports and weaknesses in web servers (Parasram et al., 2018).

While ethical hacking serves as a proactive security measure, malicious actors can repurpose the same tools for cyberattacks. The line between ethical penetration testing and cybercrime is often blurred, particularly when tools such as Kali Linux are misused (James, 2023). Security assessments using Kali Linux should be conducted within a well-defined legal framework to prevent unauthorized access and exploitation (Cisar et al., 2018). This is especially relevant for organizations operating Web-Based MIS and ERP systems, where unauthorized modifications can lead to severe financial and operational consequences (Hameed Alazawi et al., 2024).

Several case studies highlight the risks associated with unprotected MIS and ERP applications:

- Social Engineering and Credential Harvesting: Jeremiah (2019) discusses how attackers use social engineering techniques combined with Kali Linux to manipulate users into revealing credentials.
- IoT and ERP Integration Risks: Bakry et al. (2022) examine how IoT-connected ERP systems are vulnerable to cyberattacks facilitated by tools within Kali Linux.
- Automated Exploitation: Hertzog, O'Gorman, and Aharoni (2017) explain how Kali Linux enables automation of cyberattacks, increasing the risk of large-scale breaches in web-based applications.

To counteract the risks associated with Kali Linux exploitation, organizations must adopt robust security strategies, including:

1. Regular Security Audits: Conduct frequent penetration tests within a controlled environment to identify and fix vulnerabilities (Johansen et al., 2016).
2. Access Control and Least Privilege: Implement strict role-based access control (RBAC) policies to limit unauthorized access to MIS and ERP systems (Najera-Gutierrez & Ansari, 2018).
3. Security Awareness Training: Educate employees on social engineering threats and ethical hacking tools to minimize internal security risks (Jeremiah, 2019).
4. Advanced Threat Detection: Deploy machine learning-based intrusion detection systems to identify anomalous activities associated with penetration testing tools (Bakry et al., 2022).
5. Legal and Compliance Frameworks: Establish clear regulations on the ethical use of Kali Linux, ensuring that penetration testing aligns with organizational policies and legal requirements (Velu, 2017).

Kali Linux provides a powerful framework for security testing, yet its capabilities can be exploited for unethical purposes. The risk assessment of web-based MIS and ERP applications must consider both the potential benefits and security threats posed by penetration testing tools. Organizations should leverage Kali Linux responsibly within a structured cybersecurity framework to enhance their defensive capabilities while preventing malicious activities. Future research should focus on automated threat detection mechanisms and legal implications of penetration testing practices in corporate environments.

3. MAIN DATABASES AND TEST & EXPLOITATION PLATFORMS

Hackers search for historical Common Vulnerabilities and Exposures (CVEs)on exploit-db or other databases related to the listed MIS and ERP applications, you can visit the following websites:

3.1 Exploit Database (exploit-db): https://www.exploit-db.com/

The Exploit Database (Exploit-DB) serves as a comprehensive and publicly accessible repository of known software vulnerabilities and corresponding exploit codes. It is frequently utilized to uncover security weaknesses within web-based MIS and ERP platforms. By systematically analyzing entries in Exploit-DB, malicious actors can pinpoint specific vulnerabilities in these systems and deploy existing exploit scripts to breach their confidentiality, integrity, and availability.

The Exploit Database allows users to query specific vulnerabilities related to a particular system using various search filters. Researchers and penetration testers can:

- Search by Application Name: Entering "SAP ERP," "Oracle ERP," or "Microsoft Dynamics" in the search bar will yield results listing known vulnerabilities.
- Search by CVE (Common Vulnerabilities and Exposures) ID: If a CVE identifier (e.g., CVE-2023-XXXX) is known, it can be searched directly to retrieve associated exploits.
- Search by Platform: Filtering by platforms such as Linux, Windows, or web applications can narrow down potential vulnerabilities specific to a system's underlying architecture.

- Advanced Search Operators: Using "inurl:mis" or "inurl:erp" can help discover application-specific exploits.

For instance, a query like "Oracle ERP" may return SQL injection, remote code execution (RCE), or authentication bypass exploits that affect various Oracle ERP components (Alkhalaf et al, 2022).

Once a relevant exploit is found, the Exploit Database provides essential metadata, such as:

- Exploit Title – The nature of the exploit (e.g., "SAP NetWeaver Directory Traversal").
- Date Published – The disclosure date, helping assess if a vulnerability has been patched.
- Exploit Type – Whether the exploit is a proof of concept (PoC), remote exploit, local privilege escalation, or DoS attack.
- Affected Versions – Specific versions of the ERP/MIS software that are vulnerable.
- Exploit Code – The actual Python, Perl, Bash, or Metasploit scripts to execute the attack.

By analyzing these details, attackers (or security researchers) can determine how vulnerability could be exploited in an enterprise environment. Many exploits from Exploit-DB are integrated into **Metasploit**, allowing attackers to automate their execution as such within Kali:

```
msfconsole

search oracle

use exploit/windows/oracle/ebs_rce

set RHOSTS <Target_IP>

exploit
```

If the vulnerability involves **SQL Injection**, SQLmap can be used to extract sensitive data as such:

```
sqlmap -u "http://example.com/login.php" --dbs –batch
```

If an **RCE (Remote Code Execution)** vulnerability is found, a reverse shell can be deployed:

```
nc -lvnp 4444

bash -i >& /dev/tcp/<Attacker_IP>/4444 0>&1
```

This establishes remote access, allowing deeper infiltration into an ERP/MIS system.

3.2 NIST National Vulnerability Database (NVD): https://nvd.nist.gov/

The NIST NVD is a repository of information about software vulnerabilities. It provides information on the severity, impact, and remediation of vulnerabilities, including CVEs. To search for vulnerabilities related to web-based MIS and ERP applications, you can use the search bar and enter the name of the application. For example, if you are searching for vulnerabilities related to SAP, you can enter "SAP" in the search bar. This will bring up a list of vulnerabilities related to that application. The NVD enables security professionals and researchers to identify, categorize, and prioritize vulnerabilities specific to enterprise applications such as Oracle ERP Cloud, SAP ERP, Microsoft Dyno, and Infor M3. The search functionality in NVD allows queries to use:

- Application-Specific Queries: Directly entering "SAP" or "Oracle ERP" will return results related to those platforms.
- CVE Filtering by Component: Searching for "SAP NetWeaver" or "Oracle WebLogic" can refine results to critical middleware components.
- Vulnerability Type: Filtering for SQLi, RCE or XSS provides a more technical categorization.

- Severity Filtering: Using CVSS scoring filters (e.g., CVSS 9.0–10.0) isolates critical vulnerabilities that are highly exploitable.

Once relevant vulnerabilities are identified, penetration testers and security researchers can analyze CVE entries to develop proof-of-concept (PoC) exploits using Kali Linux tools such as:

- Metasploit Framework – Automates the exploitation of vulnerabilities listed in the NVD, particularly for RCE and buffer overflow vulnerabilities in ERP applications.
- SQLmap – Utilized to exploit SQLi vulnerabilities in MIS/ERP databases identified through the NVD.
- Burp Suite – Used for manual testing of vulnerabilities such as session fixation and authentication bypass vulnerabilities listed in CVE descriptions.
- Nmap and Nessus – Effective for network reconnaissance and vulnerability scanning based on CVE data from the NVD.

For example, if an NVD entry reveals a remote code execution vulnerability (CVE-2023-XXXX) in SAP NetWeaver, a researcher can use Metasploit to check if an exploit module exists or use manual payload delivery methods to exploit the weakness.

3.3 CVE Details: https://www.cvedetails.com/

CVE Details is a website that provides detailed information about CVEs, including vulnerability types, attack vectors, and impacted products.

To search for vulnerabilities related to web-based MIS and ERP applications, you can use the search bar and enter the name of the application. For example, if you are searching for vulnerabilities related to Microsoft Dynamics CRM, you can enter "Microsoft Dynamics CRM" in the search bar. This will bring up a list of CVEs related to that application.

To conduct a vulnerability assessment, users can:

- Navigate to CVE Details.
- Use the search bar to enter the name of the target MIS/ERP software (e.g., "SAP NetWeaver," "Oracle E-Business Suite," "Microsoft Dynamics 365").
- Filter results based on parameters such as:
    - Vulnerability Type: SQL Injection, Cross-Site Scripting (XSS), RCE, etc.
    - Attack Vector: Remote or local exploitation potential.
    - CVSS Score: Prioritizing high-severity vulnerabilities.
    - Disclosure Date: Ensuring recent threats are addressed.

Once vulnerabilities are identified, penetration testers can correlate them with exploitation tools in Kali Linux:

- SQL Injection (SQLi) Vulnerabilities
    - Tools: SQLmap, jSQL Injection
    - Example: If a CVE indicates an SQLi vulnerability in an ERP web portal, SQLmap can be used to extract sensitive database information.
- XSS Vulnerabilities
    - Tools: XSSer, BeEF (Browser Exploitation Framework)
    - Example: If an ERP system allows XSS, BeEF can be used to hijack user sessions or steal credentials.
- RCE Vulnerabilities

  - Tools: Metasploit, ExploitDB, Commix
  - Example: If a CVE highlights an RCE flaw in an MIS backend, Metasploit modules can exploit it for shell access.
- Authentication and Privilege Escalation Attacks
  - Tools: Hydra, Medusa, John the Ripper
  - Example: Weak password CVEs for ERP logins can be exploited using brute force or credential stuffing.
- Pre-Assessment Phase in audits: Security teams extract relevant CVEs from CVE Details and map them into organizational MIS/ERP deployments.
- Testing Phase of audits: Kali Linux tools are used in controlled environments to validate exploitability.
- Mitigation Strategy: Based on findings, organizations apply vendor patches, enforce stronger authentication, and implement Web Application Firewalls (WAFs).

It's important to use these sources in conjunction with other vulnerability assessment tools and techniques to ensure a comprehensive approach to discovering vulnerabilities in web-based MIS and ERP applications. Web-based MIS applications are designed to be accessible via internet browsers, providing users with a platform-independent interface for interacting with the underlying MIS (Baltzan, 2019). These applications typically incorporate features such as data storage, retrieval, analysis, and reporting, enabling users to access and manipulate data in real-time from any location with internet connectivity (Laudon & Laudon, 2004). However, the widespread use and accessibility of web-based applications have also attracted the attention of malicious actors seeking to exploit potential vulnerabilities and gain unauthorized access to sensitive information (Chandrasekaran & Mishra, 2016).

There are some key security challenges and common threats faced by the applications listed in Table 1:

1. SAP S/4HANA: One of the biggest security challenges with SAP S/4HANA is the complexity of its architecture, which makes it susceptible to security vulnerabilities. Common threats include cyber-attacks, insider threats, and unauthorized access to sensitive data (SAP, 2021).
2. Oracle NetSuite: Security challenges with NetSuite include data breaches, phishing attacks, and ransomware attacks. Common threats include credential theft, malware attacks, and social engineering attacks (Oracle, 2021).
3. Microsoft Dynamics 365: Security challenges with Dynamics 365 include vulnerabilities in the Azure cloud infrastructure, phishing attacks, and data theft. Common threats include credential theft, SQL injection attacks, and cross-site scripting attacks (Microsoft, 2021).
4. Infor CloudSuite: Security challenges with CloudSuite include data breaches, unauthorized access, and phishing attacks. Common threats include SQL injection attacks, cross-site scripting attacks, and DDoS attacks (Infor, 2021).
5. Epicor ERP: Security challenges with Epicor ERP include vulnerabilities in software architecture, data breaches, and unauthorized access. Common threats include malware attacks, phishing attacks, and SQL injection attacks (Epicor, 2021).
6. Sage X3: Security challenges with Sage X3 include vulnerabilities in the software architecture, data breaches, and unauthorized access. Common threats include malware attacks, phishing attacks, and SQL injection attacks (Sage, 2021).
7. Odoo: Security challenges with Odoo include data breaches, unauthorized access, and phishing attacks. Common threats include SQL injection attacks, cross-site scripting attacks, and DDoS attacks (Odoo, 2021).

8. Acumatica Cloud ERP: Security challenges with Acumatica include data breaches, unauthorized access, and phishing attacks. Common threats include SQL injection attacks, cross-site scripting attacks, and DDoS attacks (Acumatica, 2021).
9. SYSPRO: Security challenges with SYSPRO include data breaches, unauthorized access, and phishing attacks. Common threats include SQL injection attacks, cross-site scripting attacks, and DDoS attacks (SYSPRO, 2021).
10. Deltek Costpoint: Security challenges with Deltek Costpoint include data breaches, unauthorized access, and phishing attacks. Common threats include SQL injection attacks, cross-site scripting attacks, and DDoS attacks (Deltek, 2021).

4. KALI LINUX AND ITS BASIC TOOLS

Kali Linux is an open-source, Debian-based operating system specifically designed for digital forensics and penetration testing (Weidman, 2014). It includes a comprehensive suite of security tools and utilities that can be used to assess and exploit vulnerabilities in various systems, including web-based MIS applications (Tabassum et al, 2021). Kali Linux tools, such as Metasploit, Nmap, and Wireshark, enable security professionals to identify, analyze, and exploit potential weaknesses in target systems, thereby highlighting areas for improvement and bolstering their overall security posture (Weidman, 2014).

Below is a table listing some professional ethical hacking tools other than Kali that can be used for web-based MIS and ERP application testing:

Table 2: Information on the most reputable scanning tools

Tool Name	Description	Link
OWASP ZAP	A web application security scanner and vulnerability scanner	https://www.zaproxy.org/
Burp Suite	A suite of web application security testing tools	https://portswigger.net/burp
Acunetix	A web vulnerability scanner	https://www.acunetix.com/
Metasploit	A framework for exploit development and penetration testing	https://www.metasploit.com/
Nikto	A web server scanner and vulnerability assessment tool	https://cirt.net/Nikto2
Nessus	A vulnerability scanner and network security scanner	https://www.tenable.com/products/nessus-vulnerability-scanner
Nmap	A network exploration and security auditing tool	https://nmap.org/
Wireshark	A network protocol analyzer	https://www.wireshark.org/
BeEF	The Browser Exploitation Framework, used for browser-based attacks	https://beefproject.com/
Hydra	A network login cracker tool	https://github.com/vanhauser-thc/thc-hydra
John the Ripper	A password cracking tool	https://www.openwall.com/john/
Sqlmap	A SQL injection tool	http://sqlmap.org/

These links are provided solely for informational and research purposes and should not be interpreted as an endorsement or recommendation of any specific product. Users are encouraged to exercise due diligence and adhere to ethical and legal standards when utilizing these resources.

Kali Linux is a widely used operating system designed for digital forensics and penetration testing, offering a wide range of tools to assess the security of web-based MIS applications (Kali Linux, 2021).

4.1 Information Gathering

4.1.1 Nmap

Nmap (Network Mapper) is a powerful open-source tool for network exploration and security auditing. It can be used to discover hosts, services, and operating systems on a network (Lyon, 2009).

4.1.2 Shodan

Shodan is a search engine that allows users to find specific types of devices connected to the internet. It can help penetration testers identify vulnerabilities in web-based MIS applications (Matherly, 2021).

4.1.3 Maltego

Maltego is an open-source intelligence and forensics tool that helps gather information about an organization's digital infrastructure (Maryam, 2023).

4.2 Vulnerability Assessment

4.2.1 Nikto

Nikto is a web server scanner that identifies potential security vulnerabilities in web-based MIS applications, such as outdated software and misconfigurations (CIRT, 2021).

4.2.2 OpenVAS

OpenVAS (Open Vulnerability Assessment System) is a comprehensive vulnerability scanner and management system that can help assess the security posture of web-based MIS applications (Greenbone Networks, 2021).

4.2.3 Wapiti

Wapiti is an open-source web application vulnerability scanner that detects various types of vulnerabilities, including SQL injections and cross-site scripting (Alazmi & De Leon, 2022).

4.3 Web Application Exploitation

4.3.1 SQLMap

SQLMap is an open-source penetration testing tool that automates the process of detecting and exploiting SQL injection vulnerabilities in web-based MIS applications (Ibrahim & Rosli, 2023).

4.3.2 Metasploit Framework

The Metasploit Framework is a popular tool for exploiting security vulnerabilities in web applications, including web-based MIS applications. It contains a vast collection of exploit modules and payloads (Rapid7, 2021).

4.3.3 BeEF (Browser Exploitation Framework)

BeEF is a framework that focuses on client-side attacks, allowing penetration testers to assess the security of web-based MIS applications by exploiting vulnerabilities in web browsers (Alcorn, 2014).

4.4 Password Attacks

4.4.1 Hydra

Hydra is a fast and flexible password-cracking tool that supports a wide range of protocols, making it suitable for assessing the security of web-based MIS applications (Van Hauser & Kühn, 2021).

4.4.2 John the Ripper

John the Ripper is a popular password-cracking tool used for testing password strength and security in web-based MIS applications (Marchetti & Bodily, 2022).

4.4.3 Hashcat

Hashcat is an advanced password recovery tool that supports various hashing algorithms, which can be used to test the security of web-based MIS applications (Steube, 2016).

5. CASE STUDIES: HACKING AND MANIPULATING WEB-BASED MIS APPLICATIONS

5.1 Case Study 1: Exploiting SQL Injection Vulnerabilities

A well-documented case study highlighting SQL injection vulnerabilities in a web-based MIS application is the 2017 Equifax data breach. In this incident, the attackers were able to access sensitive data of millions of customers by exploiting vulnerability in Apache Struts (Apache Software Foundation, 2017). The U.S. Government Accountability Office (GAO) conducted an in-depth investigation into the breach, identifying the specific vulnerability that allowed the attack to happen (U.S. Government Accountability Office, 2018).

In a study by Ali & Alhadidi (2015), the authors employed SQLMap to demonstrate how easy it is to exploit a web application that has an SQL injection vulnerability. They used SQLMap to obtain access to sensitive information from the target application's database, underlining the importance of securing web-based applications against such attacks.

After identifying and exploiting vulnerability, it is crucial to secure the application against future attacks. A study by Jemal (2020) suggested several countermeasures to prevent SQL injection attacks. These include input validation, parameterized queries, stored procedures, least privilege, and intrusion detection systems. Implementing these security measures can help protect web-based MIS applications from SQL injection attacks.

5.2 Case Study 2: Remote Code Execution

In this case study, we will examine the vulnerability discovered in a web-based Management Information System (MIS) application that allowed remote code execution. This vulnerability was discovered by security researchers at Rapid7 (2018) who found a remote code execution vulnerability in Apache Struts 2, a popular Java-based web application framework.

The vulnerability was exploited using Metasploit, a well-known penetration testing framework. Researchers were able to utilize the exploit module 'struts2_content_type_ognl' in Metasploit to gain remote code execution privileges on the target system (Rapid7, 2018). This exploit allowed the attacker to take control of the vulnerable application and execute arbitrary commands on the underlying system.

To secure the application and prevent further exploitation of vulnerability, the Apache Struts developers released a patch to address the issue (Apache Struts, 2017). Organizations using the affected versions of Apache Struts were advised to update their systems to the latest patched version as soon as possible.

5.3 Case Study 3: XSS

One case study involving XSS vulnerabilities was conducted by Rapid7, a cybersecurity firm, which analyzed the security of an open-source content management system (CMS) called Drupal (Rapid7, 2018). The researchers discovered an XSS vulnerability in Drupal's CKEditor plugin, which could be exploited by attackers to inject malicious JavaScript code into a victim's browser (Rapid7, 2018).

A practical example of exploiting an XSS vulnerability with BeEF (Browser Exploitation Framework) was demonstrated by Kali Linux Tutorial (2016). In this case study, the researchers used BeEF to exploit stored XSS vulnerability in a web application. They injected a malicious script containing BeEF hook URL on a web page, and when a victim visited the compromised page, their browser was hooked to the BeEF server. The attacker then gained control over the victim's browser and performed various malicious activities (Kali Linux Tutorial, 2016).

To secure web-based MIS applications from XSS vulnerabilities, developers can follow guidelines and best practices provided by the Open Web Application Security Project (OWASP) (OWASP, 2021). These guidelines include validating and sanitizing user input, implementing Content Security Policy (CSP), and using secure coding practices to prevent XSS vulnerabilities. By adopting these security measures, web applications can significantly reduce the risk of being exploited through XSS attacks (OWASP, 2021).

6. MITIGATION AND SECURITY BEST PRACTICES

Regular security audits are essential for identifying vulnerabilities and ensuring the security of web-based MIS applications (Tracy et al, 2002). Audits should be performed periodically by internal or external experts, using tools like Kali Linux to simulate real-world attacks (Muniz & Lakhani, 2015). Security audits can reveal areas of weakness, allowing organizations to address vulnerabilities and implement necessary security measures (Hameed & Arachchilage, 2016).

Input validation and sanitization are critical for preventing attacks such as SQL injection and XSS (OWASP, 2021). Input validation involves verifying that user inputs match the expected data types and formats, while input sanitization entails removing or encoding potentially malicious characters (Khalaf et al, 2021). Implementing these security measures can help protect web-based MIS applications from being exploited by hackers using Kali tools (Fadlalla & Elshoush, 2023).

The least privilege principle involves granting users and applications the minimum level of access necessary to perform their tasks (Sandhu et al, 1996). By adhering to this principle, organizations can minimize the potential damage caused by compromised accounts or exploited vulnerabilities (Kizza, 2014). In the context of web-based MIS applications, this may include restricting access to sensitive data and limiting the functionality available to specific user roles (Park, 2017).

Secure password management is crucial for preventing unauthorized access to web-based MIS applications. Organizations should enforce strong password policies, including minimum length, complexity, and regular updates (Pfleeger & Pfleeger, 2006). Additionally, they should implement multi-factor authentication (MFA) to provide an extra layer of security (NIST, 2017). Password storage should be encrypted using modern cryptographic algorithms, such as bcrypt or Argon2 (Bidhuri, 2019).

Regularly updating software, including web servers, databases, and content management systems, can significantly reduce the attack surface of web-based MIS applications (Dissanayake et al, 2022). Organizations should monitor security patches and updates, applying them in a timely manner to minimize the risk of exploitation by attackers using Kali tools (Knorr, 2013). Additionally, organizations should maintain a secure development lifecycle to address potential vulnerabilities during the development and deployment of custom software components.

Therefore, the integration of AI and Kali Linux tools presents a double-edged sword for organizations relying on web-based MIS and ERP systems. On one hand, these tools facilitate proactive risk management through ethical hacking; on the other, they empower attackers with sophisticated, scalable offensive capabilities. A robust risk assessment must therefore include:

- Continuous AI-enhanced threat monitoring,
- Regular simulation of worst-case attack scenarios,
- Controlled and accountable use of Kali Linux tools in test environments,
- Integration of AI-powered intrusion prevention systems,
- Ethical guidelines for cybersecurity practitioners to prevent misuse.

A multidimensional, AI-aware, and ethically conscious risk framework is vital to mitigating the growing threats posed by the misuse of Kali Linux in enterprise IT environments.

7. ETHICAL AND LEGAL CONSIDERATIONS

Ethical hacking, also known as penetration testing or white-hat hacking, involves the use of hacking techniques to identify vulnerabilities and weaknesses in a system, with the ultimate goal of improving its security (Tiller, 2004). Ethical hackers use the same tools and methodologies as malicious hackers but do so with the authorization and intention of helping organizations secure their systems (Palmer, 2001).

In the context of web-based MIS applications, ethical hacking is essential in discovering vulnerabilities that could be exploited by malicious hackers. By using Kali Linux tools to identify and manipulate these vulnerabilities, ethical hackers can develop and recommend security measures to protect sensitive information and maintain the integrity of the system (Weidman, 2014).

Hacking activities, including unauthorized access, are illegal under various national and international laws (Gordon & Ford, 2006). For instance, the United States' Computer Fraud and Abuse Act (CFAA) and the United Kingdom's Computer Misuse Act (CMA) criminalize unauthorized access to computer systems (U.S. Congress, 1986; UK Parliament, 1990).

However, ethical hacking is legal when performed with the explicit permission of the system owner or within the boundaries of a formal engagement (Weidman, 2014). To ensure compliance with legal requirements, ethical hackers and organizations should develop and adhere to clear rules of engagement, including defining the scope of the penetration test, obtaining written authorization, and ensuring the confidentiality of any information obtained during the test (Tiller, 2004).

Ethical hackers have a unique responsibility to ensure that their activities are conducted in a manner that does not harm the systems they are testing or infringe on the privacy of individuals. This responsibility includes the following:

a) Obtaining proper authorization: Ethical hackers must obtain written authorization from the system owner before conducting any testing (Tiller, 2004). This includes defining the scope and limitations of the test and ensuring that all parties involved understand the purpose and potential risks.

b) Protecting information: Ethical hackers are responsible for maintaining the confidentiality of any information obtained during the penetration test (Palmer, 2001).

c) Minimizing harm: Ethical hackers must ensure that their activities do not cause harm to the systems they are testing (Weidman, 2014).

d) Adhering to ethical principles: Ethical hackers must follow the ethical principles of their profession, such as honesty, integrity, and professionalism (Tiller, 2004). This includes respecting the privacy of individuals, reporting findings accurately, and only using hacking tools and techniques for authorized purposes.

8. CONCLUSION

In this study, we examined the usage of Kali Linux tools to hack and manipulate web-based MIS and ERP applications. We discussed various techniques to be used to test the cyber security of web-based applications, such as SQLMap, Metasploit, and Burp Suite. We also highlighted the importance of conducting regular security assessments and implementing continuous security efforts to protect against Common Vulnerabilities and Exposures (CVEs).

Kali Linux tools can be powerful tools for security staff to test the vulnerabilities of web-based MIS and ERP applications. However, the use of these tools should only be done for ethical hacking purposes with appropriate permissions and consents. It is essential to conduct regular security assessments and stay up to date with the latest CVEs and patches for web-based MIS and ERP applications. Implementing continuous security efforts such as vulnerability scans, penetration testing, and security training for employees can significantly reduce the risks of cyberattacks and data breaches. Organizations must prioritize security and make an ongoing effort to ensure the protection of their systems and data.

Ethical hackers play a crucial role in today's cybersecurity landscape. They help organizations identify potential vulnerabilities and weaknesses in their systems before malicious hackers can exploit them. Ethical hacking is becoming an increasingly popular profession, and organizations can benefit from hiring ethical hackers or working with external security firms to conduct regular security assessments.

The findings of this study underscore the dual-edged nature of Kali Linux tools in the cybersecurity landscape of web-based MIS and ERP applications. The study confirms that web-based MIS and ERP applications often contain exploitable vulnerabilities, as demonstrated through reconnaissance, scanning, enumeration, exploitation, and post-exploitation methodologies.

The research validates Hypothesis 1 (H1), showing that widely used MIS and ERP systems can be compromised through well-documented security gaps, many of which are cataloged in databases such as Exploit-DB and the NIST NVD. By mapping known vulnerabilities from established databases such as Exploit-DB, NIST NVD, and CVE Details onto standard attack vectors—reconnaissance, scanning, enumeration, exploitation, and post-exploitation—the study demonstrated the plausibility and accessibility of exploiting these systems using readily available tools. This theoretical approach confirmed that MIS and ERP applications, as widely deployed in organizations, remain susceptible to exploitation by actors with basic cybersecurity proficiency. However, the study also supports Hypothesis 2 (H2), emphasizing that proactive security measures—such as regular patching, intrusion detection systems, network segmentation, and security awareness training—can significantly mitigate the risks posed by adversarial use of Kali Linux tools. Drawing upon expert input, literature reviews, and widely recognized frameworks, the analysis explored how proactive measures—such as timely patch management, implementation of web application firewalls (WAFs), deployment of intrusion detection systems (IDS), and regular vulnerability assessments—serve to significantly reduce the attack surface of these applications.

Furthermore, the literature review highlights that the line between ethical security assessments and cybercrime is increasingly blurred. The responsible and controlled application of Kali Linux within a robust legal and compliance framework is imperative to prevent unauthorized exploitation. Organizations must incorporate structured cybersecurity policies, enhance employee awareness, and adopt advanced threat detection technologies to safeguard their systems from malicious intrusions.

The key outcome of this study is the theoretical validation that widely used web-based MIS and ERP applications contain exploitable vulnerabilities accessible through common Kali Linux tools, and that these risks can be significantly reduced through proactive cybersecurity practices. The main benefit is that it raises organizational awareness about the dual-use nature of penetration testing tools, emphasizes the importance of structured cybersecurity measures, and provides a foundational framework for future empirical research and AI-integrated defense strategies. The recently established TR Cyber Security Agency can take several measures to ensure the security of its web-based MIS and ERP applications. These measures include conducting regular security assessments, implementing continuous security efforts, and staying up to date with the latest CVEs and patches.

Organizations must prioritize security and implement continuous security efforts to protect against cyber threats. The use of Kali Linux tools can be beneficial in identifying potential vulnerabilities, but it should only be done ethically and with the appropriate permissions. The Cyber Security Agency can take several measures to ensure the security of its web-based MIS and ERP applications, including regular security assessments, continuous security efforts, and employee training.

The following suggestions are developed in conclusion:

1. Cyber Security Professionals: As a cyber security professional, it is important to stay up to date with the security trends, technologies, and threats. Here are a few suggestions:
   - Attend industry conferences and webinars to learn about new security technologies and trends
   - Join security communities and forums to network with other professionals and share knowledge and experiences
   - Continuously hone your skills through training, certifications, and hands-on experience

- Set-up a separate process for updating with the latest exploit news and updates on emerging security threats and vulnerabilities
- Participate in ethical hacking and security assessments to improve your understanding of security risks and vulnerabilities

2. Academicians: As an academician, it is essential to impart knowledge and skills to students that prepare them for the constantly evolving cybersecurity landscape. Here are a few suggestions:

- Introducing students to cybersecurity fundamentals and best practices
- Incorporate hands-on cybersecurity exercises and projects in your curriculum
- Encourage students to participate in cybersecurity competitions and challenges to gain practical experience
- Provide students with exposure to industry experts and guest speakers

By staying informed, gaining hands-on experience, and collaborating with other professionals, students and cybersecurity professionals can prepare themselves for successful careers in the field. Future studies should focus on risk assessment of the usage of AI integrated hacking tools.

REFERENCES

Acumatica. (2021). Acumatica Security. https://www.acumatica.com/cloud-erp-software/security/

Akhtar, Z. B., & Rawol, A. T. (2024). Uncovering cybersecurity vulnerabilities: A Kali Linux investigative exploration perspective. Sciendo.

Alazmi, S., & De Leon, D. C. (2022). A systematic literature review on the characteristics and effectiveness of web application vulnerability scanners. IEEE Access, 10, 33200-33219.

Alcorn, W. (2014). Beef-the browser exploitation framework project. https://beefproject.com/

Alkhalaf, A., Alkhatib, B., & Ghanem, S. (2022, December). SQL Injection Attack Detection Using Machine Learning Techniques. In International Conference on Advanced Computing and Intelligent Engineering (pp. 145-156). Singapore: Springer Nature Singapore.

Apache Software Foundation. (2017). CVE-2017-5638: Apache Struts 2 vulnerability. Retrieved from https://struts.apache.org/docs/s2-045.html

Apache Struts. (2017, March 6). S2-045: Possible Remote Code Execution attack when using the Struts REST plugin with XStream handler to handle XML payloads. Apache Struts Announcements. https://struts.apache.org/announce#a20170306

Bakry, B. M. B., Adenan, A. R. B., & Others. (2022). Security attack on IoT related devices using Raspberry Pi and Kali Linux. IEEE.

Baltzan, P. (2019). Business driven technology. McGraw-Hill Education.

Bidhuri, V. (2019). Enhancing Password Security Using a Hybrid Approach of SCrypt Hashing and AES Encryption (Doctoral dissertation, Dublin, National College of Ireland).

Chandrasekaran, M., & Mishra, R. K. (2016). Security issues and their solution in cloud computing. Procedia Computer Science, 85, 3-13.

Ciric, V., Milosevic, M., Sokolovic, D., et al. (2024). Modular deep learning-based network intrusion detection architecture for real-world cyber-attack simulation. Simulation Modelling Practice and Theory, Elsevier.

CIRT. (2021). Nikto: Web server scanner. Retrieved from https://cirt.net/nikto2

Cisar, P., Cisar, S. M., & Fürstner, I. (2018). Security assessment with Kali Linux. Bánki Közlemények.

Deltek. (2021). Costpoint Security. https://www.deltek.com/en/products/project-erp/costpoint/security

Dissanayake, N., Jayatilaka, A., Zahedi, M., & Babar, M. A. (2022). Software security patch management-A systematic literature review of challenges, approaches, tools and practices. Information and Software Technology, 144, 106771.

Epicor. (2021). Epicor Security. https://www.epicor.com

Fadlalla, F. F., & Elshoush, H. T. (2023). Input Validation Vulnerabilities in Web Applications: Systematic Review, Classification, and Analysis of the Current State-of-the-Art. IEEE Access, 11, 40128-40161.

Greenbone Networks. (2021). OpenVAS: Open vulnerability assessment system. Retrieved from https://www.openvas.org/

Hameed Alazawi, S. A., Abdulhameed, A. A., & Others. (2024). Comparative study on applications of cybersecurity tools for Kali Linux operating system. AIP Conference Proceedings.

Hameed, M. A., & Arachchilage, N. A. G. (2016). A model for the adoption process of information system security innovations in organisations: a theoretical perspective. arXiv preprint arXiv:1609.07911.

He, Y., Zamani, E., Yevseyeva, I., & Luo, C. (2023). Artificial intelligence–based ethical hacking for health information systems: simulation study. Journal of Medical Internet Research, 25(1), e43231.

Herman, H., Riadi, I., & Kurniawan, Y. (2023). Vulnerability detection with K-nearest neighbor and naive Bayes method using machine learning. International Journal of Artificial Intelligence Research, 7(1).

Hertzog, R., O'Gorman, J., & Aharoni, M. (2017). Kali Linux revealed. Mastering the Penetration Testing.

Howard, M., & Lipner, S. (2006). The security development lifecycle. Microsoft Press.

Ibrahim, R. Y., & Rosli, M. M. (2023, December). Evaluation of Web Application Vulnerability Scanners using SQL Injection Attacks. In 2023 IEEE 8th International Conference on Recent Advances and Innovations in Engineering (ICRAIE) (pp. 1-6). IEEE.

Infor. (2021). Infor Security. https://www.infor.com/trust/security

James, J. W. (2023). Engineering the human mind: Social engineering attack using Kali Linux. SN Computer Science.

Jemal, I., Cheikhrouhou, O., Hamam, H., & Mahfoudhi, A. (2020). Sql injection attack detection and prevention techniques using machine learning. International Journal of Applied Engineering Research, 15(6), 569-580.

Jeremiah, J. (2019). Awareness case study for understanding and preventing social engineering threats using Kali Linux penetration testing toolkit. ech Insig.

Johansen, G., Allen, L., Heriyanto, T., & Ali, S. (2016). Kali Linux 2–Assuring security by penetration testing. Packt Publishing.

Kali Linux Tutorial. (2016). BeEF XSS Framework – Kali Linux 2016. Retrieved from https://www.kalilinuxtutorials.com/beef-xss-framework-kali-linux/

Kali Linux. (2021). About Kali Linux. Retrieved from https://www.kali.org/about-us/

Khalaf, O. I., Sokiyna, M., Alotaibi, Y., Alsufyani, A., & Alghamdi, S. (2021). Web Attack Detection Using the Input Validation Method: DPDA Theory. Computers, Materials & Continua, 68(3).

Khan, S. U., Eusufzai, F., Azharuddin, M. R., et al. (2022). Artificial intelligence for cyber security: performance analysis of network intrusion detection. In Artificial Intelligence for Cybersecurity (pp. 121-140). Springer.

Kizza, J. M. (2014). Computer network security and cyber ethics. McFarland.

Knorr, K. (2013). Patching our critical infrastructure: Towards an efficient patch and update management for industrial control systems. In Securing critical infrastructures and critical control systems: Approaches for threat protection (pp. 190-216). IGI Global.

Kujanpää, K., Victor, W., & Ilin, A. (2021). Automating privilege escalation with deep reinforcement learning. In Proceedings of the ACM Workshop on Artificial Intelligence and Security.

Laudon, K. C., & Laudon, J. P. (2004). Management information systems: Managing the digital firm. Pearson Education Limited.

Lyon, G. F. (2009). Nmap network scanning: The official Nmap project guide to network discovery and security scanning. Nmap Project.

Mahmood, M., Hossain, M. M., Farah, R. M., et al. (2024). Self-poisoning network to prevent reconnaissance by generative artificial intelligence. In Lecture Notes in Artificial Intelligence. Springer.

Marchetti, K., & Bodily, P. (2022, May). John the Ripper: An Examination and Analysis of the Popular Hash Cracking Algorithm. In 2022 Intermountain Engineering, Technology and Computing (IETC) (pp. 1-6). IEEE.

Martínez, A. L., Cano, A., & Ruiz-Martínez, A. (2025). Generative Artificial Intelligence-Supported Pentesting: A Comparison between Claude Opus, GPT-4, and Copilot. arXiv preprint arXiv:2501.06963.

Maryam, U. (2023). Phishing Attacks Facilitated by Open-Source Intelligence. International Journal of Computer and Information Engineering, 17(10), 587-590.

Matherly, J. (2015). Shodan: The search engine for the internet of things. Retrieved from https://www.shodan.io/

Messier, R. (2024). Learning Kali Linux: Security testing, penetration testing & ethical hacking. Packt Publishing.

Microsoft. (2021). Dynamics 365 Security. https://docs.microsoft.com/en-us/dynamics365/security/

Moorthy, R. S. S., & Nathiya, N. (2023). Botnet detection using artificial intelligence. Procedia Computer Science, 219, 1023–1030.

Moustafa, N. (2022). Digital forensics in the era of artificial intelligence. Taylor & Francis.

Muniz, J., & Lakhani, A. (2015). Penetration testing with raspberry pi. Packt Publishing Ltd.

Najera-Gutierrez, G., & Ansari, J. A. (2018). Web penetration testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux. Packt Publishing.

Nilă, C., Preda, M., & Apostol, I. (2021). Reactive wifi honeypot. In Proceedings of the IEEE Conference on Electronics and Artificial Intelligence.

NIST. (2017). Digital identity guidelines: Authentication and lifecycle management. Special Publication 800-63B. National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-63b

Odoo. (2021). Odoo Security. https://www.odoo.com/security

Oracle. (2021). NetSuite Security. https://www.netsuite.com/portal/products/security.shtml

Ordoñez, G. S., & Guerra, T. C. (2018). Prototype of a security system with artificial intelligence using neural networks and evolutionary algorithms. In Springer International Conference Proceedings, Monterrey, Mexico.

OWASP. (2021). Cross-Site Scripting (XSS). Retrieved from https://owasp.org/www-community/attacks/xss/

OWASP. (2021). OWASP top ten project. Retrieved from https://owasp.org/www-project-top-ten/

Oz, E. (2008). Management information systems. Thomson Course Technology. https://www.amazon.com/Management-Information-Systems-Sixth-Effy/dp/1423901789

Pamarthi, K. (2020). Artificial intelligence and machine learning techniques to control SQL injection attacks. Journal of Scientific and Engineering Research, 7(5), 101–108.

Parasram, S. V. N., Samm, A., Boodoo, D., Johansen, G., & Others. (2018). Kali Linux 2018: Assuring security

Park, J. S. (2017). U.S. Patent No. 9,769,177. Washington, DC: U.S. Patent and Trademark Office.

Pfleeger, C. P., & Pfleeger, S. L. (2006). Security in computing. Prentice Hall.

Rapid7. (2018). Drupal CKEditor Module XSS Vulnerability. Retrieved from https://blog.rapid7.com/2018/03/28/drupal-ckeditor-module-xss-vulnerability/

Rapid7. (2018, March 8). Apache Struts 2: CVE-2017-5638. Rapid7 Blog. https://blog.rapid7.com/2018/03/08/apache-struts-2-cve-2017-5638/

Rapid7. (2021). Metasploit: Penetration testing software. Retrieved from https://www.metasploit.com/

Sage. (2021). Sage X3 Security. https://www.sage.com/en-us/products/sage-x3/security/

Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). The protection of information in computer systems. IEEE Computer, 29(2), 38-47.

SAP. (2021). SAP S/4HANA Security. https://www.sap.com/products/s4hana-erp/security.html

Steube, J. (2016). Hashcat: Advanced password recovery. Retrieved from https://hashcat.net/hashcat/

SYSPRO. (2021). SYSPRO Security. https://www.syspro.com/security/

Tabassum, M., Mohanan, S., & Sharma, T. (2021). Ethical Hacking and Penetrate Testing using Kali and Metasploit Framework. International Journal of Innovation in Computational Science and Engineering, 2(1), 09-22.

Tracy, M., Jansen, W., & McLarnon, M. (2002). Guidelines on Securing Public Web Servers Web Servers. NIST Special Publication, 800, 44.

U.S. Government Accountability Office. (2018). Data protection: Actions taken by Equifax and federal agencies in response to the 2017 breach (GAO-18-559). Retrieved from https://www.gao.gov/assets/gao-18-559.pdf

Van Hauser, M., & Kühn, D. (2021). Hydra: A parallelized login cracker. Retrieved from https://github.com/vanhauser-thc/thc-hydra

Wang, P., & Johnson, C. (2024). The impacts of generative artificial intelligence (AI) in knowledge discovery and generation for cyber defense. Issues in Information Systems, 25(1), 215–229.

Weidman, G. (2014). Penetration testing: a hands-on introduction to hacking. No starch press.

Zhuravchak, D., Opanovych, M., et al. (2024). Design of an integrated defense-in-depth system with an artificial intelligence assistant to counter malware. Eastern-European Journal of Enterprise Technologies, 9(3), 45–60.