ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

ANNEX 5

JOHN VON NEUMANN FACULTY OF INFORMATICS

THESIS

OE-NIK	Student's name:	**CICIL MUKALEL KURIAN**
2022-23	Student's registration number:	**T/007239/FI12904/N**

Óbudai University
John von Neumann Faculty of Informatics
Institute for Cyber-physical Systems

THESIS WORK SPECIFICATION

Name of the student:	**Cicil Mukalel Kurian**
Identification number of the student:	T/007239/FI12904/N
Neptun's code:	[redacted]

Title of the thesis:

Security breaches within ethical hacking and network penetration testing
Biztonsági problémák az etikus hackelés és penetrációs tesztelés során

Internal supervisor	Ernő Rigó
External supervisor:	
Deadline:	15 December 2022
Subjects of the final exam:	Computer Architectures Cloud service technologies and IT security specialization

The task: In the digital era ethical hacking and penetration testing have significant roles to ensure the safety of the users. The change in technology made it clear that data is asset and cyber attacks are more dangerous. As ethical hacking is a modern profession the code of ethics and policies are not scripted, moreover we engineers need to close the loops to provide more secure environment for testing and to find the data/security breaches while doing the penetration testing. This paper deals with the analysis of code of ethics, the existing breaches and put forward a technical solution also proposing a unified testing mandate. Citing the future this paper will also discuss about how national securities can be affected if the penetration testing goes illicit.

The thesis must contain:

- Detailed analysis of the existing code of ethics.
- Using Kali Linux checking how effective is network penetration done using the existing protocols, and how much it is deviating from the existing policies.
- Graphs showing the deviation, rising data breaches and how a modified policies can increase trust and efficiency.
- Major threats of existing pen-testing and how to overcome (especially recovery time)
- More secure environment can be made, furthermore how the security breaches can affect the national securities.

Place of stamp

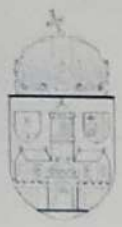
.....
Dr. Rita Fleiner
head of institute

This specification is valid until: **15 December 2024**
(According to OE TVSz 55.§)

I consider the thesis suitable for submission:

.....
external supervisor

.....
internal supervisor



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

NEUMANN JÁNOS
FACULTY OF INFORMATICS

Annex 7

STUDENT'S DECLARATION

I the undersigned student hereby declare that this degree project / thesis is the result of my own work; I have disclosed the references and tools used in an identifiable manner. The results indicated in my degree project / thesis completed may be used by the university and the institution announcing the task for their own purposes free of charge.

Dated: Budapest, 15th December 2022

student's signature

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Neumann János Faculty of Informatics

CONSULTATION LOG

Student's name: Cicil Mukalel Kurian

Neptun code: [redacted]

Specialty: [redacted]

Degree project / thesis title in Hungarian: **Biztonsági problémák az etikus hackelés és penetrációs tesztelés során**

Degree project / thesis title in English: **Security breaches within ethical hacking and network penetration testing**

Institutional supervisor: Ernő Rigó

External supervisor:

Please write the data in printed capital letters!

Occ.	Date	Content	Signature
1.	19/9/2022	Discussion about code of ethics of cerficatons	
2.	10/10/2022	Analysis of performance of testing tools	
3.	14/11/2022	Discussion about implementation and testing of ai based model for Vulnerability testing	
4.	12/12/2022	Finalizing the validations	

The Consultation log is required to be countersigned by either supervisor on a total of 4 occasions of consultation.

Dated: Budapest, 12.12.2022

institutional supervisor

ÓBUDAI EGYETEM
OBUDA UNIVERSITY

Neumann János Faculty of Informatics

CONSULTATION LOG

Student's name: Cicil Mukalel Kurian

Neptun code: [redacted]

Specialty:
[redacted]

Degree project / thesis title in Hungarian: **Biztonsági problémák az etikus hackelés és penetrációs tesztelés során**

Degree project / thesis title in English: **Security breaches within ethical hacking and network penetration testing**

Institutional supervisor: Ernő Rigó

External supervisor:

Please write the data in printed capital letters!

Occ.	Date	Content	Signature
1.	21/02/2022	Discussion about the thesis subject and an initial idea	
2.	10/03/2022	Description about the works and the further details	
3.	20/03/2022	Finalized research work tasks and how to move forward with the work	
4.	27/04/2022	Submission of the first half of research	

The Consultation log is required to be countersigned by either supervisor on a total of 4 occasions of consultation.

Dated: Budapest, 11.05.2022

institutional supervisor

CONTENTS

TABLE OF FIGURES

TABLE OF ABBREVIATIONS

ABBREVIATIONS	DEFENITION
ISACA	Information Sharing and Analysis Centre
CREST	Certificateless Registry for Electronic Share Transfer
CWE	Common Weakness Enumeration
AI	Artificial Intelligence
HTML	Hypertext Markup Language
PHP	Hypertext Pre-Processor
SQL	Structured Query Language
DNS	Domain Name System
IP	Internet Protocol
CEH	Certified Ethical Hacker
ISC	International Information System Security Certification Consortium
CVE	The Common Vulnerabilities and Exposures
XSS	Cross-Site Scripting
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
VLAN	Virtual Local Area Networks
OS	Operating System
TCP	Transmission Control Protocol
CVSS	Common Vulnerability Scoring System
CCA	Cross Cloud Automated
CESA	Cyberspace Electronic Security Act
GUI	Graphical User Interface
UDP	User Datagram Protocol
SSL	Secure Sockets Layer
AWS	Amazon Web Services
AE	Ai And Expert
DAC	Digital Automation Cloud
TLS	Transport Layer Security
HTTPS	Hypertext Transfer Protocol Secure
API	Application Programming Interface
VM	Virtual Machine
RSA	Rivest-Shamir-Adleman Algorithm

ABSTRACT

This research work deals with the fundamentals of penetration testing and a code of ethics. Exponential growth in the data generated made it one of the most valuable sources in the technological era. Safeguarding data became one of the priorities of every organization. The paper talks about issues during vulnerability assessments, which is a key part of ethical hacking and penetration testing. It also contains an analysis of the existing tools used for testing . Experts' opinions about the code of ethics and analyzing the testing mandate led them to a model for reducing the issues while performing vulnerability assessments.

INTRODUCTION

Accelerating technological advancement enabled by the Internet facilitates innovation. Since the advent of the era where data was the more valuable item, the threat posed by cybercriminals in the Internet age has also expanded significantly. There must be more technologically robust answers to the problem of protecting people's data considering the rise in cybercrime and data breaches. When most people hear the word "hacker," they immediately picture someone who is trying to steal money or sensitive information. They are technical professionals who know how to exploit loopholes and vulnerabilities in the system, how to get unauthorized access to a website or application and extract the data they want, and how to conduct covert surveillance of the system without the users' knowledge. Such actions can be detrimental to companies as they undermine confidence, undermine secrecy, and disrupt operations. To combat this issue, businesses have begun to employ "ethical hackers," sometimes known as "white hat hackers," who can uncover and fix security flaws within the company's systems.

This paper is divided into six chapters. The initial chapters deal with the basic concepts of hacking , penetration testing, and ethical hacking. The third chapter is a study report that explains in detail the code of ethics and the testing mandates followed by certified testers. The next two chapters deal with the cost important section of this research work the readers will be able to find the recent vulnerabilities , mitigation methods, and comparison of the proposed model with existing automated tools . Chapter five deals with the implementation, working, and results generated by the model .

CHAPTER 1

1.1 Hacking

Hacking refers to the act of breaking into a computer system without the authorization of the system's administrator. Hackers, in other words, are experts who have the technical ability to detect the flaw in an existing system and, by exploiting the disadvantage, obtain access to the interface, backend, or database of the system[1]. Depending on the circumstances or the context in which it was carried out, hacking could be considered either lawful or criminal. A sort of hacking that is considered ethical is one in which the process of breaking into a system is carried out in accordance with a predetermined set of guidelines. There are also hackers that are able to get into a system without wreaking havoc on the system's design or causing damage to the customers, and they then inform the owner of the product that there is a security flaw in the system. However, even with a catchy tagline, it is against the law to crack software without first obtaining permission from the program's creator. "A hacker with morals uses the same tools and methods as a hacker, but instead of damaging the target system or stealing information, they analyze the security flaws in the target system, notify the system's administrators, and provide suggestions for fixing those flaws. vulnerabilities they uncovered and advise how to correct the flaws" [2].

Ethical hacking is a method for conducting a system security audit. It involves doing a series of tests in order to identify the system's weaknesses; nevertheless, this method can never guarantee a system to be completely safe[3]. The corporation should correct the flaw, which is primarily the responsibility of the developer, in light of the reports that were produced by the ethical hacker, which reveal that an intruder may hack the system at a specific moment in time and with a certain degree of ability. In a broad sense, ethical hacking may be seen as a kind of risk assessment for potential vulnerabilities in a system. The first step in the security cycle is the preparation phase, often known as planning. After an organization determines that it is necessary to formulate a policy, the policy environment will be developed, and a management committee will be established to oversee the operation of the system. There must be an intrusion detection facility that notifies the proprietor of the system whenever a potential danger is discovered inside the infrastructure of the system. After it has been located, a security evaluation will be carried out in order to determine the severity of the attack and the means by which it may be thwarted. Following a series of hypothetical situations, the organization or the institution will become aware

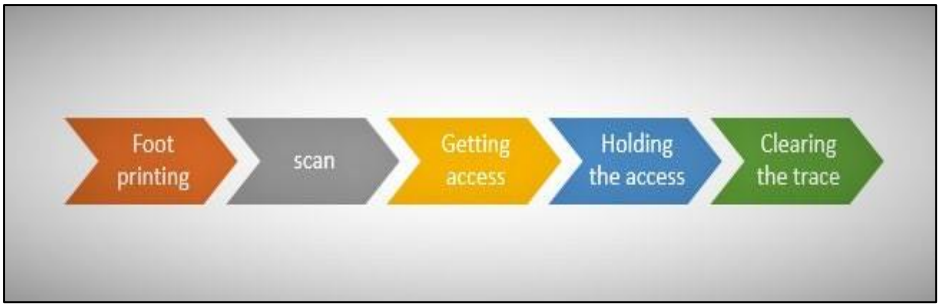
of the need to develop a security policy with which their further actions are compatible[4]. The Figure *1* below demonstrates the security life cycle.

Figure 1:security life cycle[5]

1.1.1 Phases of Hacking

Even though the process sounds easy it's not that much easily hackable any type of hacking consists of five basic steps they are Reconnaissance , scanning, gaining access, maintaining access and clearing tracks[6]. The order of execution is not important but in most cases the collection of information is the basic step, because having the clear understanding about the system we need to attack is the priority for this the attacker can use direct or indirect means of approach which can be used for foot printing. In the next phase of scanning ports are scanned to find the open and closed ones , vulnerabilities of the system are being scanned using the tool , for example in Kali Linux lots of fault detection tools are available, the next thing to do is to map the network , by using tools like N-map or something similar[7]. After these two phases the attacker would have gained access to the system or application but that not all now the attacker should get the admin rights to make changes or access confidential data, for this to happen the attacker installs an external software or a program which gives him rights , not only that, but major changes can also happen in the systems network. Once the attack is successful and the attacker got the control of the system its equally important to maintain the accrued access and that the next important phase, but if the attacker's motto is done the maintain access is not necessary. Closely related to this step and one of the most important one is to erase the finger prints[8], as an all-time famous English saying getting caught will only point you as a criminal, so erasing all the tracks that can lead the way towards you should be deleted including the files and folders , the changes made by

the attackers so that it's tough to understand that an attack was done on



the system. The *Figure 2* below can make the process understand bit more easily.

Figure 2:Different stages of hacking

Briefing a bit more the art of hacking not only need skill but also needs the knowledge of some basic tools , but it is always good for a hacker to have prominent knowledge of programming and scripting languages such as java, python , Linux , bash ,html and so on . As per my experience no tools are cent percent accurate so using more tools can mitigate the problem more accurately and reduce the faults thus increase the accuracy of the scanning or any identification process . Moreover, there are many tools available than the open-source tools , in my opinion I always prefer open-source tools though I can enjoy the full freedom or access[9]. Most of the tool's working is widely available on the internet so as per the advanced technology the basic tool knowledge might not help us for escalating a test and get into the systems privilege admin to gain access and make changes.

1.1.2 Hacking Techniques

Based on the way or the process of execution hacking can be categorized majorly into six types starting from the Dictionary attacks: where the hackers targets on the password of the client using a thus followed reference dictionary and run through it , though to the reason most users use some words from the dictionary. There are couple of important dictionary-based attack one is the pattern-based attack where the attacker uses a more advanced version of dictionary which contain different password references whereas the second one is more damage creating one where the hacker tries to steal the users card details by different number sequence[10].

The second type of attack is the web-based attack: knowing the importance of web application the attacker initiates malware attacks to steal the information from the users. The attack is conducted in a way that it first attacked the website which can allow that attacker to access the client

machine once they start using it or even the attacker implements malware into the system which can reduce the working of the machine. In the inspect element using the right-click method, you can temporarily change the source code of a web page. To see the source code, right-click the page and select analyze element. The source code for programming languages such as html, PHP, python, pearl, and others can then be inspected. With a basic understanding of these scripting languages, one can make changes to the original code as needed. You can copy the source code of the webpage using the inspect element and use it to create your own website with minor changes. Short-term customers are the most prevalent victims of this strategy[11]. Another form of web- based attack is SQL injection where the attacker uses malicious quarries to get into the database and try to corrupt the database or helps to leak the data. A cross-site scripting attack occurs when a web software obtains dangerous data from a user. The online application frequently receives data in the form of a hyperlink containing potentially hazardous content. The hacker can inject script into the web page that is visited by others by inserting script into the HTML output generated by the web application[12].

The next major technique is the Phishing Attacks : these are the oldest form of cyber-attacks which at least most of us have tried this attack can extend its hands to many more cyber-attacks. In simple terms its liking putting the hook for the target to share its information , mostly massive emails are being send mainly with a login link which direct the user to a fake website or a cloned one, once they enter the credentials in it the attacker can get it in the backend[13].

Other important type of attack is the wireless hacking: these days we find a lot of open wireless networks, but we never thought of whether these network are safe or not because for most of us the primary need is to access internet and the hackers are well sure of that , they can use something named Meta exploit which is a tool in kali Linux which have source codes to control the devices connected to the same network using this they can easily access you device and loot lots of personal information’s[14] .

The Dictionary Harvest: approach may be used by an attacker to find the valid email addresses linked with a mail server. These addresses are then gathered and sold to anyone wishing to send spam or steal personal information. To negotiate thee-mail account, an attacker must know the user's id. Using a Directory Harvest attack, he may obtain a list of email addresses. Email addresses are usually separatefrom e-mail-ids[15].The way of attack initiation is very similar to the phishing; they use brute force to generate massive mail sender in a go the other method is by selective attack in that case if an id is not found we will get a no response notification.

The last major kind of attack is by using the keylogger they are background-runningsoftware programs that gather each keystroke and track the user's activity. These keystrokes are then logged in the system, and data is transferred to the hacker. Thehacker then analyses the data to get access to the network by determining the password or other relevant data. The attacker may either force or convince the targetto run the Keylogger software. The target might be duped if the Keylogger softwareis attached to a seemingly legitimate email. There are hardware loggers which can be attached to the hardware of the system and monitor the notes, Acoustic loggers uses the micro sound generated by the keypads to track the notes , there arewireless and software loggers which can track the system through an internal far- end way[16].

1.2 Types of Hackers

Hackers may be classified as "black hats," "white hats," or "grey hats" based on their motivations. Black hat hackers are actual cybercriminals since they break into networks, steal information, and use it for harmful purposes. Ethical hackers are people who breach computer systems with no malicious intent. These individuals are often referred to as "white hackers." Gray hackers are individuals who operate on the fence and are able to use the information they collect in either an ethical or unethical way, depending on the circumstances. Those that fall between white hats and black hats are known as "grey hackers" because they may use the information they acquire for good or harm. Now that we've covered the fundamentals, let's go further into each of the three kinds of hats[17].

1.2.1 Black Hat Hackers

These are the types of hackers who attack the system and stealthe data use it for illegal activities[18]. They are the real threats because data now a days are the most powerful weapon. They basically crack the systems architecture to enter it and can spread malwares to bring down the whole system partially or fullyshut it down.

1.2.2 Gray Hat Hackers

Out form the other two these are the types of hackers who resamples the nature of both ethical and non-ethical hackers, they basically crack the system and derive the information’s but using it for a any illegal activities depends

on because in most cases the hacker informs the company that there is a security fault in the system , so that the company can solve it and also can pay himin return for his work, something similar to bug bounty[19]. In other words, they do security breaches but without doing much harm to the system. As per my understanding grey hackers are someone who do network testing or hacking as a part of their hobby.

1.2.3 White Hat Hackers

Ethical hackers are hired by clients to evaluate their own IT security. They follow a set of official requirements to protect the client's assets, particularly commercial assets. Penetration testers are white hat hackers who perform more than just vulnerability testing. To stress and examine their client's IT security, they may utilize a range of approaches, including social engineering skills.An ethical hacker will attempt to imitate a black hat while remaining out of the company's bad books[20]. He'll scour the physical and virtual worlds for vulnerabilitiesthat hostile hackers could exploit. The ethical hacker is instructed by the target, or more particularly, the target's representative, to stress and test the target's IT security. If the black hat is acting as a mercenary, the objective of a black hat attackis always a third party, not the black hat himself[21].

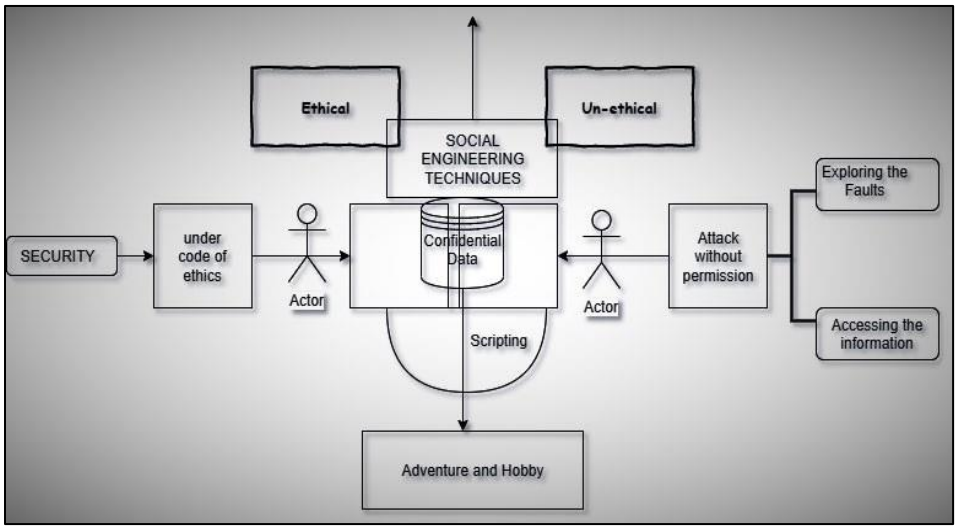


Figure 3.Un/Ethical hacking

According to the Figure *3*, the difference between ethical and unethical hacking is rather minimal. The statistics in the figure support this conclusion. This conclusion is supported by the fact that the gap between the two groups is not all that significant. The graphic demonstrates that the difference is most likely to be discovered near the origin of the problem. Even if every possible

negative event occurs, the hacker will still have access to the classified data. White hats are evaluated according to market-determined criteria. The most challenging aspect of this circumstance is that the customer lacks the technical ability to determine if the ethical hacker's activities violate the given criteria[22]. This is because the average consumer lacks the technical expertise required to make such a determination. Given the circumstances, this is the most urgent issue that arises. Given the frequency of data theft, it is impossible to attribute the whole issue to foreign hackers. Numerous data breaches have happened, necessitating this. If a similar incident were to occur with data pertaining to national security or other sensitive sectors, the threat may be significantly exacerbated[23].

Consumers must now place their faith in an ethical hacker since there is no alternative. Given that the customer has put their faith in the ethical hacker, the hacker owes them a token of appreciation. The willingness of an ethical hacker to abide by a client's regulations is a condition for working with that client. If hackers are to behave themselves ethically, something is required. Any violation of these standards has the potential to harm the ethical hacker's reputation, which might limit their ability to progress in their chosen area[24]. Now, unlike in the past, many firms will only let a white hat Do penetration testing after the contract has been signed.

This is due to the fact that signing the contract is analogous to obtaining permission for a hacker to join the system and search for security flaws. This is because accepting the conditions of the contract is equivalent to giving a hacker permission to test the system's vulnerabilities. This is because agreeing to the contract's terms is equivalent to granting a hacker permission to access the system and search for security holes. This is the case because accepting the conditions of the contract is equivalent to allowing a hacker to join the system and search for weaknesses in the safeguards it provides. Due to the fact that accepting the terms of the contract is equivalent to allowing a hacker to join the system and search for security flaws, this is the effect[25]. Due to this reality, we are now in a dilemma. The reason for this is that the concerned organization has authorized the hacker to continue testing. Consequently, we are in this dilemma. As stated in the conclusion, the organization must provide the ethical hacker with administrative competence. Since there is a possibility that the corporation may provide these powers, the consequences if anything goes wrong will be far more severe. As part of the theoretical foundation for delegating administrative functions to an ethical hacker, this topic was brought up for discussion. This may require a drastic redesign of the system's or network's architecture, which might expose sensitive data to the public. If anything does go wrong, the repercussions will be far more severe than they are now. Should take the

written concern of the client before conducting a test forsecuring.

Let’s take an example of the code of conduct that ensure the hacking ethical[26]:All the employees’ privacy should be ensured. Should not cross the limits set by the client. use of proper scientific and ethical means of testing and all the process shouldbe documented. letting the client know the results without keeping any loopholes for furtherentries. informing the manufactures and developers about the detected problems removing all the trace is equally important. On many occasions the need for these set of rules are important because it can in a way added more protection and can standardize the working principle of an ethical hacker[27].

CHAPTER 2

2.1 Penetration Testing

The purpose of penetration testing is to determine the vulnerability of a system by performing a certain number of tests or checks to understand the core adaptability of the network[28].It is in other words the mode of ethical hack testing which allows the tester to understand the system loops and can report to the administrator about the discovered vulnerabilities. The main reason why these tests are done is to ensure that no malicious exterior attackers can harm the system if there is an existing codeleak which can make you confidential files to get exposed and make your system more vulnerable.

Taking in account for the steps for pen testing the tester should have a concrete knowledge of the system and how is it organized then after scanning the system the tester should be able to figure out the loopholes in the framework , it is important to know these because a hacker can also use the same techniques tomitigate into the system. Mainly we have five testing strategies depending on the reason of execution[29].

Companies that do penetration testing have a responsibility to build ethical decision-making frameworks with great care. In reality, establishing an ethics board with the ability to select the course of action in all morally questionable scenarios, ranging from whether to accept an ambiguous engagement to operational questions during testing, seems to be beneficial. This could be related to why this situation has arisen. This may be the result of an ethics board's capacity to determine the optimal course of action in each and every instance when ethical issues are presented.

The firm's workers should choose the members of the Ethics Board; the company's management should have no influence on the issue. Rather, the firm's employees should make the selection. Executives are permitted to serve on the Ethics Board, although they should make up a negligible portion

of the board's overall membership. The company should be legally obligated to conduct its activities in accordance with the Ethics Board's recommendations[30]. .

2.1.1External Testing

These kinds of attack take place outside the framework of the system[31] , for example I am trying to access the system through internet by collecting information about the client or company and the attacking the target mostly in these kind of testing publicly available information like DNS, websites company mails are used by the tester to try getting into the companies system and if they are able to access it that means that there are major security issues and then they forward the test results to the developer for further rectification.

2.1.2 Internal Testing

This kind of testing is done to protect the company form the insiders attack , let's say an employee of the company is trying to corrupt the system from inside in this case how a company can tackle these situations and to what extend the system can me destroyed by these attacks because from inside it will be more easy to attack the system interface, mostly in these type of test the pen tester tryto create or suggest for a different level of user access or a credential requirement to perform certain activities[32] . It can be a bit tough on the side for anemployee or a client to use credentials for each execution but considering the security its more important. The Figure 4 shows the internal and external testing setup.

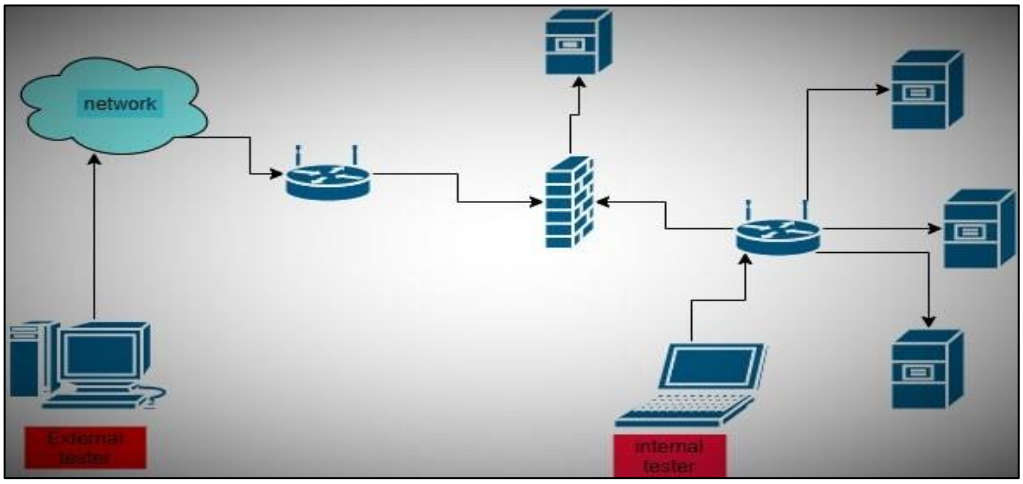


Figure 4.Inside and outside testing mandate

2.1.3 Targeted Testing

In these types of testing all the parties know that the testing will be done so the company , tester and the people will be prepared prior , these are some kinds of test that can save a lot of time as well a capital. They have a specific target and will have all details of the system ; about the type of test, they should perform[33]. In my opinion these kinds of test are more safe thinking from a client side because the transparency of the test in evident before its done.

2.2 Classification Based Strategy

Like hats of hacker's penetration testing can also be divided into white , black, andGray depending on the requirement of the organization , we will try to go throughit briefly to gain some good knowledge in penetration testing environment.

2.2.1 Black Box Penetration Test

In this kind of testing environment, the tester doesn'thave an actual knowledge about the test or test system , but they are very clear aboutthe expected results[34]. Here the tester doesn't really bother about the inside details ofthe system , they try to gather external information about the target from the outside.In most cases they are not sure how the system gave that output. Moreover, what structure the network have doesn't really bother, the merited technical knowledge is not in demand for these tests.

2.2.2 White Box Penetration Test

This type of testing is known by the name open boxtest though to the reason that before the execution of the test the tester receives all the information's related to the system including source code , detail about thesystem , the accessible IP address and so on , these are more structural tests and can detect cent percent of network failures. These structured test runs through allthe codes and can determine path and data flows[35]. With the help of these kinds oftesting design errors that occurs due to the issues or deviation from the logical codeand the flow programs can be mitigated.

2.2.3 Gray Box Penetration Tester

These kinds of tester have very limited knowledge about the internal

code of structure of the network[36]. In my opinion I prefer this kindof testing though to the reason that it provides equivalent respect to the tester and the developer because the tester do not have access to the source code so that provide a moral integrity for the developer and can avoid conflicts within the company. In simple terms we can relate this testing to the activity of an unauthorized system user trying to gain access so the need of knowing inside network structure is not necessary[37].

2.3 UN-Ethical Hacking

Is this hacking consistent with at least one ethical value? Certainly not, because such hacking might violate other fundamental ethical principles. Indeed, to be called ethical, hacking must adhere to at least the most fundamental ethical norms at stake while staying adequately balanced[38].

As a result, not every unethical hacking is unethical. So, what precisely constitutes ethical hacking? In most cases if a hacker respects the guidelines set by the stakeholders it can be termed as ethical hacking or in general, he follows the code of ethics[39].

In this point every time I had a question that how a hacker can follow the stakes codes if the code is unethical so more than the value its mostly about choice so bringing a standardized mandate is convenient in that way the integrity and principles can be maintained. We have couple of points to be pick from here the stage is set in a way that if a hacker follows the rules set by the client side whether those rules deeds are un-ethical or ethical it doesn't matter to the ethical hacker if he follows it blindly he is termed as ethical hacker and if an external hacker tries to hack the same client knowing that their cause is bad and did attack for safeguarding will be coined as black hat, these are very contradictory situations.

According to this logic, unethical hacking is hacking that does not follow all ethical norms and moral standards in that setting. It makes sense to regard any illegal hacking to be ethical if it is done for a good cause. We frequently meet conflicting ethical values[40].

Not all ethical ideals can be upheld at the same time; they must be prioritized in terms of purposes and behaviors in each environment. As a result, reducing a generic concept of ethical hacking to unethical hacking would result in a meaningless definition. We've discussed the overall (morally) best solution. In the case of a value conflict, it should be acknowledged that in practice, a pluralist community may not agree on a single means of balancing and resolving value trade-offs. Consider creating trust in cybersecurity against attaining justice as one example of balancing conflict[41]. Both views may be at odds when a white hat hacker discovers

evidence of unethical behavior or suspected indicators of unlawful action by a firm during pen testing[42].

These are more towards the moral side than the technical side because as a hacker who follows the ethics set by the company cannot reveal that there are some sort ofillegal activities happening in the company not only that indirectly I would turn in to his duty to protect those sensitive data, and not let any sort of investigation dragtowards it. We can assume that firms will be hesitant to hire penetration testers unless they trust them to protect their own interests in any case. We may also arguethat broad and protected trust in the services of white hat hackers is crucial to ensuregood levels of cybersecurity for overall social structure. In most cases I felt that ethical hacking is a choice between technical and social justice[43].

Companies who donot have a completely clean record in terms of ethical and legal behavior should be included or not. This contradicts another fundamental value: the purpose of seeking immediate justice and shielding potential victims of a crime or unethical conduct. Therefore, the need for a committee is really demanding as I said in the paper previously about internal committee a supervised external committee can beformed. In that instance, a penetration tester can always expose highly unethical behavior or criminal clues to the public, or at the very least threaten to do so, to offer the organization an incentive to report the problem to an external body.

As it was mentioned earlier, an 'ethical' hacker may be forced to choose between retaining trust in oneself and white hat hackers in general on the one hand and accomplishing justice or other ethical ideals in the short term on the other. It shouldbe noted that depending on the legal environment in which the white hat hacker operates, the choice between trustworthiness and other ethical ideals may be resolved differently[44].

Assume the hacker works in a jurisdiction where the law mandates the white hacker to violate a confidentiality if significant offenses are proven. An attacker would decrease trust by giving hints or even proof of illicit behavior on the part of corporations. However, this is not the same as acting unethically: depending on the extent of the unethical behavior revealed, the act including a breach of trust is the most ethical (ethically ideal), or even the only ethical (morally required). Nothing guarantees that the (most or only) moral path ofaction will always be the legal course of action. It should be highlighted that whendeciding between these two legal systems, society or its elected representatives must choose between several, equally legitimate societal values. The dilemma necessitates striking a balance between increasing motivation to rely on white hat hackers on the one hand and identifying certain serious crimes in the short term onthe other. Civilizations may decide where the utilitarian optimum rests, while othersocieties may pass

legislation based on non-utilitarian concerns.

For example, eventhough confidentiality protection is the most utilitarian option, serious discourse ofa case in which a white hat hacker has a legal responsibility to conceal heinous behavior may convince public opinion against it[45]. Highlights that ethical decision cannot be restricted to the a priori assumption that business-oriented standards should take priority, and that the term "ethical" should not be restricted to a specific understanding of professional ethics.

Competence in a hacker, on the other hand, is knowledge. It is unethical and makesno explicit ethical assertions. In terms of ethics, the hacker's tools are also unimportant. This is not to suggest that hacking tools are morally ambiguous[46]. Indeed, the development or lack thereof of some hacking tools, such as weaponizedzero-days, raises fundamental ethical concerns on a societal level: on the one hand, weaponized zero-days allow country to establish cyber-weapons to stop potential opponents; on the other hand, unpatched gaps can endanger large-scale IT-systemsif discovered by or made available to black hats[47]. The WannaCry global ransomwareattack in May 2017, which knocked down UK hospitals and other networks, emphasizes the significance of a weaponized zero-day falling into criminal hands[48]. Finally, the evaluator must only analyze the hacker's values and mannerof activity in terms of ethics. It is important to note that the reviewer can either be the hacker or some other entity[49].

CHAPTER 3

3.1 Code of Ethics Analysis

A risk management decision framework is designed to help you identify which security risks are worth taking and which ones aren't. The ethical hacking community has a set of agreed-upon values that guide their decision-making process. These values are based on the work of Mario Bunge, a philosopher who specialized in pragmatic ethics[50].

Mario Bunge's pragmatic value theory explores the idea that technologists have a unique opportunity to take personal responsibility for the moral implications of their work[51]. He suggests that "technical rules" are needed to create an ethical system that can be used by philosophers[52]. Broadly, techno ethics is the study of the social and moral implications of technology. It is an interdisciplinary field that looks at the impact of technology on society and individuals[53]. For Mario, technology and ethics are two sides of the same coin. He believes that ethics can be a part of technology, and that ethical issues should be investigated just as thoroughly

as any technical problem[54].

Mario Bunge's practical theory of value emphasizes the utilitarian nature of technology[55]. Applying this theory to achieve ethical (fair and more efficient) use of technology can help derive an ethical code of conduct for engineers. Below are Bunge's three ethical guidelines for responsible engineering[56]:

- Technology should aim to benefit society, not just special interest groups.
- Technology should be used in a way that is fair to all people affected by it.
- Technology should be used in a way that is efficient and causes the least harm possible.

Bunge's technical approach is focused on balancing goals with potential side effects, with a focus on efficiency and fairness. This approach is used to help the technology (its usefulness within social systems) through the application of this approach. Quality and value can be evaluated[57].

A practical decision-making framework is essential for this methodology to work. Technological ethics assessment must integrate multiple disciplines and perspectives, as well as involve multiple stakeholders in data collection and analysis to understand the implications, ethics, uses and value of emerging technologies in society and make ethical decisions[58].

3.2 Security values of present

After understanding how the base was set, we will now examine the current available code of ethics or mandate for professionals. We will be looking at the Global Information Assurance Certification Code of Ethics, EC-Council Code of Ethics, ISACA Code of Professional Ethics, The International Information System Security Certification Consortium, and CREST Code of Conduct[59].

3.2.1 Global Information Assurance Certification Code of Ethics

The GIAC Code of Ethics is made up of four sections[60]: Respect for self, respect for accreditation, respect for clients and respect for public. Officials who adhere to this code of ethics are expected to "be responsible, act in the best interest of the public and ensure ethical and lawful conduct." In order to maintain confidentiality, competency and accurate representation of skills and certifications, it is important to avoid conflicts of interest.

All customers have the right to expect privacy when it comes to their protected information. To ensure this, the customer signs a confidentiality

agreement. Data protection is one of the most important ethical obligations in information security for this reason. For example, one way to keep customer information separate is by using the Biba integrity model[61]. Another common industry rule is to limit administrative privileges within departments and limit access to specific computers whose activity can be monitored - like in the Bell-LaPadula model[62]. GIAC offers many high-level security certifications, such as digital forensics, security management, and penetration testing[63]. These certifications are highly regarded in the security field and can help you further your career.

3.2.1 EC-Council Code of Ethics

The Certified Ethical Hacker (CEH) certification offered by the EC-Council is recognized by the US Department of Defense[64].
The followed mandate is including confidentiality of the gained information that means this information cannot be sold to a third party. All the finding will be granted to organizer. Proper evaluation of the target should be done, and enough knowledge should be necessary. Use of legal software. The transactions should be transparent. If doing an internal testing only use the organization resource with their knowledge and approval. Managing conflict of interest. Producing a well-structured report. Keeping self-respect. Do testing within legal limits. Always bound to law[65].

3.2.2 ISACA Code Of Professional Ethics

ISACA was founded in 1969 with a focus on IT governance. There are over 140,000 members worldwide. Information security and cybersecurity professionals can be trained and certified by ISACA[66].Control, audit, safety, and risk management are of the main concern it also provides appropriate standards and procedures for the efficient governance and management of enterprise information systems and technology, duties to professional standards with objectivity, sensitivity, and professional diligence. Serve the stakeholders lawfully, maintain high moral and ethical standards, and do no harm to our profession or association. Maintain the privacy and confidentiality of information collected during our activities, except as required by law. This information will not be disclosed to inappropriate persons or used for personal gain. Agree to maintain competence in your respective field and to engage only in activities which can reasonably be expected to have the necessary skills, knowledge, and expertise. Notify stakeholders of the results of work. Disclose all material information they know

In short, a code of conduct consists of the norms and methods, a reasonable level of effort, legitimate leadership and secrecy, and the skills upheld to move things forward based on knowledge[67].

3.2.3 The International Information System Security Certification Code Of Ethics

Over 125,000 information security professionals are members of this international non-profit organization. They have majorly two code of conduct which is further having sub-classifications. Code of ethics preamble and canons, the preamble states that we must uphold and make visible the highest ethical standards for the safety, welfare and common good of our society. We are also committed to our customers and each other. Canons works on four major guidelines those are Protection of society, public welfare, infrastructure, and necessary public trust. Honesty, justice, responsibility, and lawful behavior are important. A service provider with care and competence. Protect and promote occupations[68].

3.2.4 CREST Code of Conduct

The UK is home to the non-profit organization CREST. There are dynamic sections in Europe, Middle East, Africa, India , Americas, Asia, Australia, and New Zealand. CREST's goal is to "ensure that organizations and their security personnel have a certain level of competence and qualification when performing security tasks such as penetration testing, threat intelligence, and incident response[59], [69]. The trusted advisor role includes providing advice and services in the field of technical information security. This entails responsibility and accountability to others. This activity is results-oriented and focused on achieving positive results that benefit and protect our clients. Principles that guide universal ethical and good business conduct. They talk about practices that our certified member companies and individuals should follow. The CREST code of conduct mainly comprises of Advancing Best Practices, Expert Advocacy, CREST Assignments, Regulations, Competencies, Client Benefits, Sanctions, Ethics, Responsible Reporting[70]

3.3 Study Report

3.3.1 Code Of Conduction

With the assistance of security specialists and client companies, research was conducted to assess testers' understanding of rules of conduct and ethics, as well as their credibility. For this study, seven security specialists were individually interviewed: three accredited testers, one security staff head of a multinational corporation, one student pursuing a doctorate in cyber security and networking, and one certified ISACA professional professor. The purpose of this study was to obtain an understanding of how different stakeholders within the testing community perceive the importance of ethics and credibility when conducting security testing. The results of this study will help inform future research in this area. To ensure that the responses received from the poll participants were reliable, an online questionnaire was used. The poll was sent to those who work in the fields of security and testing. In total, there were approximately 60 responses to the online poll. The challenges were designed to assess fundamental comprehension and to simplify the job. In the first phase, two alternatives were provided to determine if the respondent was a client or a security expert; more than sixty percent were security experts. Let's suppose that 88% of the experts were qualified testers, auditors, or hackers.

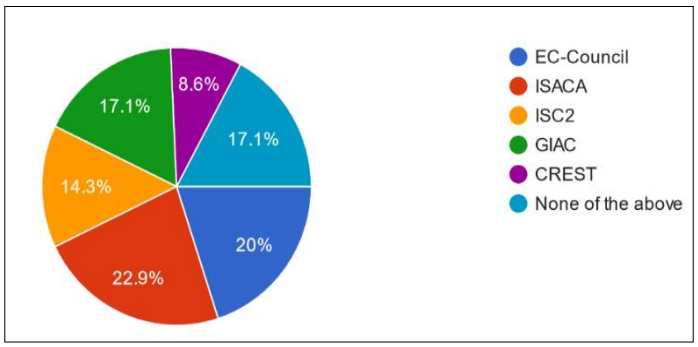


Figure 5:Responders Certification

The majority of the remaining 12 percent have received workplace training; relatively few do it voluntarily. 22.9% of qualified professionals are ISACA-certified, 20% are EC-Council-certified (Figure 5), and, curiously, 17% have been taught and certified by their respective firms. The findings of the study showed that a majority of respondents (45%) believed that following the code of ethics was more important than adhering to the law. Furthermore, a vast majority of respondents (54%) always observed the law during very rigorous testing. These findings astonished me, as nearly 42% of

licensed professionals have not followed or have only partly emphasized the code of ethics. However, 54% of respondents found that tests that adhere to the rules are vital, and roughly 45% said that security measures are more relevant than adhering to legal testing processes (Figure *6*). The ethical hacker posed a question to me: if a system is tested and found to have significant flaws that are then fixed, who is liable if a black hat hacker subsequently breaches the system via an untested secret file? They also asked what ethical standards should be observed if a sensitive file that could potentially damage society is discovered during testing.

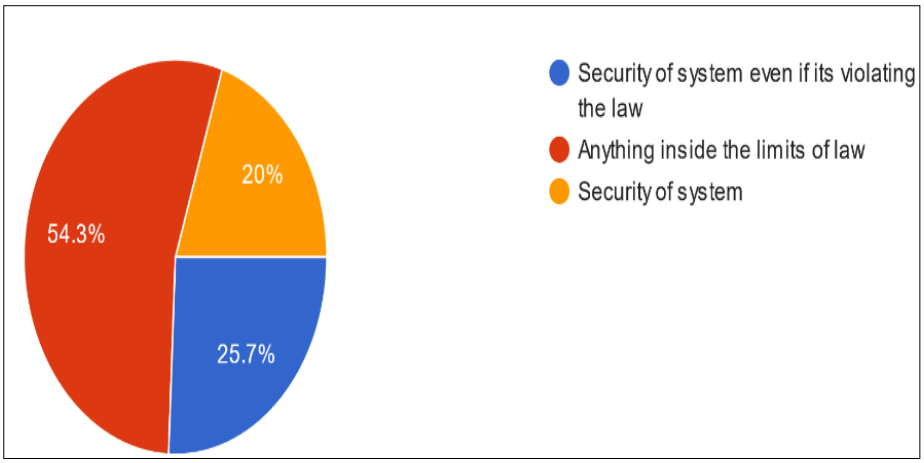


Figure 6:Priority while testing

For every good business it's always important to have review and opinion from both employee and employer[71] for this and to give more dimension to the research, we have 40% of our responses came from client side. They were asked about their way of selecting a tester, what agreement they follow and is there a trust issue. A majority of the client; 39% of them prefer certified testers whereas 34% said they can be trained by the company with their own certification. On the other hand, 26% really don't consider whether the tester is certified or not, but the security should be ensured, another reason for this is the pay scale where the company can get an uncertified yet skillful employee in a more cost-efficient way this even caused confusion in the certified professionals on what if the things go wrong, the business cost can be much more than expectation. When it comes to the agreements (Figure *7*) 56% of them follow companies' regulations for testing whereas 26% of the clients don't have them own code of conduction rules so makes use of general rules set by the certification authorities.

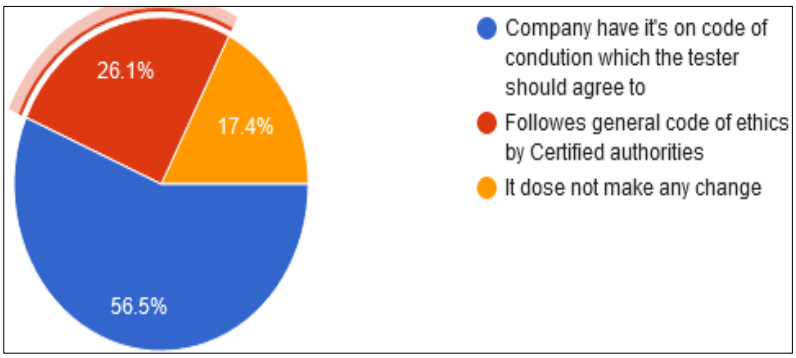


Figure 7:followed agreements

There was a section that was made to check the trust of the client on their testing professionals (Figure *8*), it was very interesting that 52% of the clients opted that's it's not the matter of trust but there is always a way of monitoring the testers work in the company. It's always said that trust is an issue which proves that 21% of the clients don't have blind trust in their employee, 26% have complete trust on their tester even without any monitoring. During the survey the most interesting section for me was the 52% of the other monitoring methods, there were three main ways of doing that; Within a subnet, keep track of all network activity. This, however, is of little value if the pen tester generates a lot of noise. If testing takes place over several days, it also becomes highly expensive. Destroy machinery from the space (in system aspect; to avoid backdoor root access). Although a skilled hacker can switch to a different machine or another server on the same subnet as the one, they have access to, this keeps the machine secure while it is in their line of sight. Give access to systems where the testing workspace is configured as a virtual desktop for ethical hackers. Checking shell history to see what that computer is doing[72].

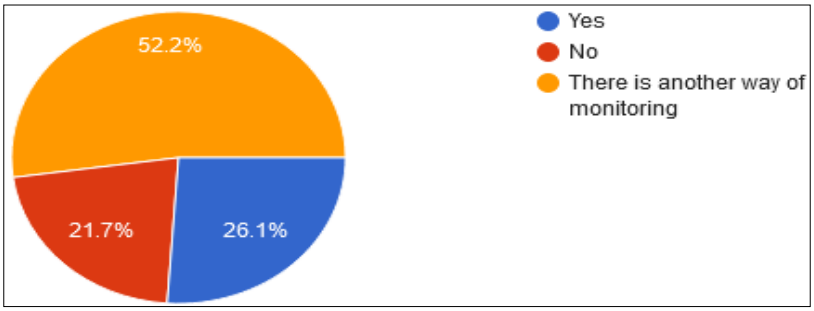


Figure 8:Need for monitoring

From the above survey it was clear that don't give your pen tester access to your production facilities until it's necessary. Instead, construct a

hypothetical system with the same configuration but no databases, sensitive data, etc. If the pen tester works at a manufacturing facility, you will need to accept that as fact. If it's necessary to let an unreliable pen tester work on production systems, create a backup of your website on a different server so that any server changes may be compared to the backup.

This might not be achievable if the entire network needs to be tested or if a test system that is exactly like the production system cannot be put up. In that scenario, you will need to have faith in them. A skilled pen tester will typically be dependable. They can't afford to ruin your computer and risk tarnishing their reputation in a field where repairing it is so challenging. You and the pen tester will both sign a contract that specifies what they are permitted to perform, which attacks are covered by it, and other information[73]. This will allow you to prove negligence if they violate the guidelines, such as when they stress test your production server against your preferences and end up essentially denying service.

To draw a conclusion for this section of study I should say that there are many codes of ethics of protect the customer from the data breaches, but the main problem is that it is so scattered that picture is not yet completed. This clearly shows the need of a unified code which can be adopted. It is always recommended to have a certified professional, if possible, give them company training so that they can clearly understand the limits in which they should work. Making an agreement is the most important part, this can safeguard if any breaches happen. Having a way to monitor is the best option but there isn't a perfect answer to this. I believe we should combine the following three standards:

View what they are doing: This might be both rational and physical. a person monitoring them who has a similar level of competence. You can also ask for a thorough explanation of their job and contrast it with statistics gathered through monitoring. You can even instruct them to bring nothing and take nothing. Using companies own tools or laptop. On the other hand, we can offer specialized laptops with a variety of protocols. On the other side, pen testers frequently employ specialized, very inconvenient tools.

Using a virtualized environment: Use a test environment that is comparable to your real environment but separate from it and devoid of important information. Although it might seem like the ideal choice, the distinction between mock and production reflecting outcomes often renders this option completely worthless. Additionally, you may simply revert modifications made after the pen test is finished and restrict the scope of the pen tester to a predetermined setting.

Professional trust: You might eventually have to have faith in the pen testers you hired. working with those who are to work as a pen tester, you

may need a special permit in some locations. Considering pen testing a concern in and of itself.

3.3.2 Ethical Hacking Goals, Means, Values And Ethics

Based on the review from the experts the whole output of the testing can be classified as elements of administration and aspect of technology. One section suggested that improving information security and data protection is the main goal of ethical hacking or testing whereas the other group believes assessing the vulnerability is the main goal. The EC-Council Certified Professional's goal is to work with the wider community to continuously improve information security. Information assets were protected by ethical hacking, according to experts and university professors. For the rest it is to assure applications, information assets, including software and data, must be protected from unauthorized modification or damage, including potential threats to the confidentiality, integrity, or availability of information. In other words, ethical hacking involved identifying weaknesses and addressing them before attackers could take advantage of them. The constant enhancement of information security's performance is the strategic objective of information security management. Protection of information assets is a second objective. Participants' perspectives included socio-cultural, technical, and financial perspectives on the potential adverse consequences of using ethical hacking within an organization. The main concerns were social and cultural. Ethical hacking had a negative reputation in public relations as it put information at risk. Another technical issue was that ethical hacks, if done incorrectly, could damage information systems or erase data. Second, the system may continue to be exploited after the ethical hack has been performed, and the system can survive accidental damage during the ethical hack procedure. The potential theft of documents related to ethical hacking, such as reports and strategies, was also a major financial concern. So, upgrading the system was expensive. Ethical hacking can therefore be costly.

The information security risk assessment employed both open source and commercial resources. In their conversations with their authorities, experts favor open source. Commercial software providers are more likely to update security fixes that are more easily available than open-source vendors[74], and some application programs that are better suited to the firm are commercial rather than open source. Many respondents thought that proprietary software could be more appropriate than open source. Businesses occasionally prefer open-source software to commercial software due to financial reasons.

By valuating the risk assessment utility, we can evaluate the testing from

an administrative and technical point of view. The capacity to make decisions regarding the best security techniques and countermeasures is a way to measure the value of testing. When it comes to technical perspective the evaluation is based on prioritizing the risks[75]. Using ethical hacking as a pragmatic value system, we weigh the costs and benefits of countermeasures against the risk of attack and the associated damage costs to select the appropriate countermeasures[76]. To determine value, estimate the possibility of an attack and the resulting cost of stopping it. Most experts advised protecting other components after the Internet layer[77].

This is one of the most important and interesting area of opinion. We mostly discussed about the ethics behind the ethical hacking we already discussed is a quick note about the code of ethics of some security organization. As per the experts the ethics evaluation is based on the following the technical and legal procedure. In a more specific argument, it can be a systematic approach that help in understanding the assets, upcoming threats, and to evaluate the asset whether its worthy attacking it. The one who do security testing as a passion argued that it’s not about ethics because he considers it as a technical evaluation process. He added that it’s not about attacking the system it’s about how wisely the hacked data is being used. Two certified hackers and a client talked about the legal side of the assessments. From the client he said that there were always companies’ policies any security tester should sign, and a declaration should be made so that the hacker gives permission to use companies instruments and gives authority to supervise his testing duties. The certified professionals opted that a white hat must be able to break into information systems and patch security holes with permission from senior management. Tester can only "hack the system legitimately." For the certified professional It's crucial to have a contract in place with a company that permits access of confidential data before you begin testing. By doing this, attackers can evade legal action. A certified expert said that “Ethical hackers are always invited by the companies to make sure their system is secure and if anything goes wrong, the company should be blamed not the ethical hacker”[78].

3.4 Key Findings

The above-mentioned code of ethics is structured in a way that the confidentiality of the user data is given major priority. The GIAC book of ethics is saying that when it comes to confidential information, all customers have reasonable privacy expectations. EC-council stays that Confidentiality of Information Obtained. This means that this information cannot be sold to third parties. All decisions are given to the organizers. ISACA asserts to

maintain the privacy and confidentiality of information collected during the testing activities, exceptions as required by law. This information will not be passed on to inappropriate people or used for personal gain. ISC code of canons clearly states the need to protect customer infrastructure. CREST's mandate is something that personally impressed me though they bring responsibility and accountability to others. There activity is result-oriented and focused on achieving positive results that benefit and protect the clients. From the above-mentioned code of ethics, we can say that five major ethical decision theories are presented in the form of ethical decision frameworks to assist in conducting comprehensive ethical analysis of ethical dilemmas in the real world. Utilitarianism, deontology (duty ethics), virtue ethics, rights ethics, pragmatic ethics.

In the EC-Council rule book they mention that the targets should be accurately assessed and well-informed before conducting of any test. ISACA mainly give priority to Securing, managing risks, auditing, and controlling. From my reading and close analysis of the code of ethics of different security organizations its very clear that any given institution is concerned about the data security, data privacy, data integrity and updated knowledge of the system or target. As the survey shows a vast majority of the professional prefer to stand in the legal side but a good percent of certified and most noncertified hackers gives importance to systems security whereas the data integrity is not given much priority this can create very dangerous trend in the society. It's always recommended to go with a certified professional even though companies need to invest more considering the end security which can earn a lot of reputation, the initial investment can turn worthy. Trust remains as an unsolved question as an interviewed client said, "No contract fully protects or mitigates the error, one mitigated error will open up other, so there is nothing about trust, but a continuous security teste is needed". Having an agreement is always important to safeguard both parties. The use of virtualized platforms is always recommended and creating a back door protection[79]. The use of organizations system, network and software is recommended so that nothing comes into the company, and nothing goes out of the company.

CHAPTER 4

4.1 VULNERABILITY DETECTION

A technique for locating security flaws in a system or system environment is vulnerability assessment[80]. Find exploitable flaws, their level of exploitability, and any potential harm they could do to a system with a specific purpose. Therefore, an essential component of company's organizational security procedures is effective vulnerability management.

In general network scanners scans network, search for vulnerability and valuate the vulnerability. Finding active hosts on your computer network, as well as the operating systems and services they operate, is the initial stage in scanning[81]. To properly confirm a vulnerability's presence, many tools try to ascertain whether the listed functionality really exploits it. Understanding the value of the vulnerabilities figured is an important task because it’s a way to evaluate how much that security fault can cost and what should be the priority given while solving the patches. This shows us how crucial the assessment testing is it not a matter of concern whether the scan was done using existing tools, manually or automated the result is more important, but the precision is still very less. Many occasions the testing tools gave false reports that and haven’t reported the existing issues this opened a grater issue of working towards non existing problems on the other side give proper time frame for the intruders to sneak and infect the system[82]. The work main point of concentration is towards the data breaches, not finding and evaluating a target is considerably a breach itself though it can open the scope for the hacker to bug the system. So, we will evaluate the efficiency of the assessment tools used by hackers to find out how much is the reliability.

The problem with ratings is that each combination of features is different, making it very difficult to compare items on a broad level. Furthermore, no quantitative data are provided regarding the accuracy of the assessment. For the study and comparison, we focused on tools that can detect bugs in software, configuration. We will analysis port scanning ability of the tools. We will note the passive and active scanning. We will keep an eye on the exploits that the tool can detect. Making credential checks as well as cross checking the entries in database with respect to vulnerabilities. We will evaluate the precision of the checks.

4.2 Recent Vulnerabilities

According to the recent reports (Figure 9), there are around 38,000 CVE data points spanning the past two years' worth of time[83]. Out-of-bounds

write and cross-site scripting, which is sometimes referred to as XSS[84], continue to be the two vulnerabilities that provide the most risk. Race conditions have moved up to position 22 from position 33, code injection has moved up to position 25 from position 28, and uncontrolled resource usage has moved up from position 27 to position 23. These are also the newly discovered classes of vulnerabilities that were added to the list for the year 2022. These reorganizations are among the most important ones that have taken place. Furthermore, both command injection and NULL pointer dereference climbed up the list a few positions.

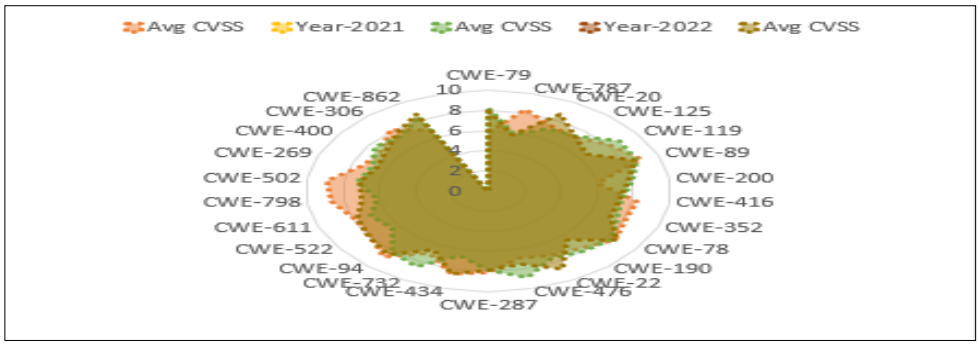


Figure 9:[82]–[84],Recent critical Vulnerabilities

Some of the vulnerabilities where in the past was moved out from the list CWE-200 is one among that , incorrect permissions ;CWE-732 fell to lowers scorings and was moved out from the most critical rated vulnerability[85].

4.2.1 CWE-787

The CVE critical vulnerabilities for the last 3 years its very evident that ‘out-of-bounds write’ CWE-787 is consider are one of the most critical vulnerabilities, for the last couple of years it leads the CWE average scores. In short what this vulnerability exploits the majority of the time, this may result in data corruption, a crash of the operating system, or the execution of malicious software[86]. It is possible for the script to alter a pointer calculation that uses an index or points to a memory location outside of the memory address that is located outside of the buffer's parameters[87]. When data is stored via the buffer, this might potentially take place. After then, each subsequent write operation will produce results that are either unforeseen or cannot be specified.

4.2.1.1 Detection And Mitigation

In many cases, to detect this vulnerability by using automated techniques for performing static analysis. With the help of a number of recently developed technologies, such as flow analysis and constraint-based methods, the number of false positives might potentially be cut down significantly. In

most cases, automated static analysis does not take into consideration environmental conditions before reporting on memory operations that are outside of their expected range[88]. As a consequence of this, users could have a difficult time deciding which cautions they should implement first. For example, an analysis tool can point up buffer overflows that are triggered by Using command line options with a program that isn't meant to run with intended to run with any other form of special privileges[88]. These overflows can occur when the program reads in too many arguments at once. This would be an illustration of a circumstance in which the tool would prove beneficial.

Make use of a coding language that either does not allow this vulnerability to occur or that provides features that make it easier to avoid triggering this weakness. If at all possible, choose a language that does not allow it to occur. Utilize a library or framework that has undergone extensive testing and either eliminates the possibility of this issue arising or provides components that make it simple to get around it if it does occur[88].Execute the program while creating Buffer overflow maybe reduced or completely avoided by the program by employing features or extensions that automatically offer a protective mechanism automatically provide a protection mechanism that lowers or eliminates buffer flow

This should be done in tandem with the execution of the program. Rather than using unbounded copy functions, you should use equivalent methods that accept length arguments. One possible alternative is to use strncpy rather than strcpy. You need to make sure that people can get to them if they aren't already there[88].

4.2.2 CWE-79

CWE-79 is the next most common and critical application vulnerability it basically arises due to the fluctuating input while generating web page. It occurs due the fact that it is possible for information that should not be trusted to be introduced into a program that is executed on the web via the usage of a web request. The web software builds a website on the fly, and it is possible that the website may include malicious material. The application does not perform any filtering that would remove data that might possibly contain browser-executable components prior to the creation of the pages. These browser-executable components include JavaScript, HTML tags, HTML attributes, mouse events, Flash, ActiveX, and a variety of others[89]. Because unreliable data was used to inject the malicious script into the web page that was created, which can be read by a web browser, the user runs the danger of being infected with malicious software. The malicious code will be

run by the target web interface inside the constraints of the targets server's main page since it was supplied by the web server as part of its web interface[89]. In this case, the web server is the aggressor. This is because the script was sent to the web server, where it was successfully received. This activity represents a flagrant violation of that policy since it is against the same-origin policy of a web browser to let scripts from one domain to access resources or execute code on another domain. A web browser must adhere to the same-origin policy[89].

After successfully injecting the malicious script, the attacker may carry out a number of damaging, system-harming operations. It is possible for private information to be sent from the victim's system to the attacker's computer. Cookies, which may include session-related information, are an illustration of this. This is one of the conceivable effects of an adversary's intrusion into a system[89]. An adversary may impersonate a victim in order to send harmful queries to a website. This is a very real possibility. It is feasible that this may do even greater damage to the website if the victim has the administrative skills required to manage the website at issue. For the purpose of deceiving users into divulging their passwords, phishing assaults may seem to be legitimate websites. This allows the attacker access to the victim's account on the website that was the target of the attack. The third scenario involves the script exploiting a security hole in the victim's webpage to "drive-by hack" the machine and gain control. Until the actual attack happens, the person who is about to be assaulted usually has no clue that they are the intended victim[90]. Even when consumers take precautions, attackers often use different methods of encoding techniques for the corrupted script of an attack. These methods, which may involve encoding the URL with Unicode or using Unicode in its entirety, are intended to make the request seem less suspicious[89].

4.2.2.1 Detection and mitigation

Implement automated techniques for static analysis that are primarily geared toward the discovery of flaws of this sort. When it comes to saved cross scripts, the indirection that is potential difficulties presented by the data repository to discover the problem. The XSS string must first be entered by the tester into the database next, they must locate the component in the program that permits broadcasting the XSS text to other users. who are using the software[91]. The activation of the XSS might take place in either of these two distinct procedures any time between minutes and days after the vulnerabilities were initially inserted into the database. This activation could take place at any moment[91].

Utilize a tested module or library that either does not let this vulnerability become exploitable or that includes components that make it easy to bypass any prospective concerns. Be sure you are aware of all the entry points that might be used by untrusted inputs. Include all external data sources, such as databases, parameters, cookies, network information, configuration files, DNS queries, response flags, URL sections, e-mail, directories, and file kinds, as well as all other external data sources. You may help prevent XSS attacks from being launched against the session cookie of the client by setting the value of the cookie to HTTP Only. XSS stands for "cross-site scripting."[92] Because of this property, potentially dangerous papers may be sidestepped. Client-side applications that make use of cookies should be prevented from accessing the session identifier for the client. client-side applications that make use of cookies Utilize a firewall that has the capacity to detect efforts being made to take advantage of this vulnerability[93].

4.2.3 CWE-89

The other critical vulnerability is related to database which arise due to the improper using of special elements in query language ; commonly known as SQL injection [94], CWE-89 which moved 3 places higher in the list to reach the 3rd most critical vulnerability[85]. In an era where the data is considered as one of the most important resource this vulnerability and why attackers target the database is not surprising. If inadequate efforts are made to remove or quote SQL syntax in those inputs, it is conceivable that the user-controllable inputs may be read as SQL rather than as ordinary user data as a consequence of the resultant SQL query[95]. This will occur if the SQL syntax is not removed. This could be used to change the query logic in order to avoid security checks or to introduce additional statements that have an impact on the back-end database. These statements could include the execution of system commands. Alternatively, this could be used to avoid having an impact on the front-end database. The user has the ability to choose either of these two options. SQL injection has become a ubiquitous issue, particularly for websites that are supported by databases. Because the vulnerability can be located and exploited with relatively little effort, it is highly likely that an attempt will be made to exploit it against any website or software package that has even a moderate number of users. This is because the vulnerability can be found and exploited with relative ease. This issue is brought on by the fact that SQL does not make a meaningful distinction between the control plane and the data plane in any of its implementations

4.2.3.1 Detection and Mitigation

If dynamic tools and methods are utilized, include fuzz testing, resilience testing, and operator error. fault injection, all of which interact with the program using enormous test suites that include a wide variety of inputs, it may be possible to identify this vulnerability. These tools and methods interact with the program in a way that simulates real-world conditions. All of these are examples of dynamic tools and approaches. Although it is conceivable that the functioning of the program may become slower, it should not become unstable, crash, or produce incorrect findings. Scanners for databases, scanners for web apps, and scanners for web services are three examples of the various sorts of detection methods that might be advantageous[96].

Using permanent layers such as Hibernate or Enterprise Java Beans, which, if used in the appropriate manner, may offer a high level of protection against SQL injection, is one strategy that may be utilized[97].

The use of structured methodologies that automatically maintain the separation between the data and the code Rather than depending on the developer to supply this functionality at every site where output is created, these systems could be able to automatically provide the appropriate encryption, verification, and quoting instead[96]. You have to do input validation using a technique referred to as "accept recognized products." This suggests that you should make use of a checklist of permitted contributions entirely meeting the criteria that have been defined as standards. Any information that is supplied but does not precisely conform to the criteria should either be discarded as it is or transformed into something that does adhere to the standards[96].

It is vital to make use of strict allow lists while creating database query statements. These allow lists limit the character set depending on the predicted value of the parameter in the request. This is carried out in a manner that is consistent with the SQL standard. Even though input validation might provide some additional defense in depth, the most effective way to prevent SQL injection is to make sure that output encoding[98], escaping, and quoting are carried out in the correct manner. This is the case regardless of whether input validation might provide some additional defense in depth. On the other hand, protecting against SQL injection could also be possible via the use of input validation[99]

4.3 Performance and Accuracy Analysis

For finding the efficiency of the tools we are using Open VAS, NeXpose, Nessus, Nmap, Manual testing and AI trained testing. Four virtual machines have been used where two VLAN have been used. Different OS was used namely window 7, Centos 9, Kali Linux. Many configuration mistakes have been done to check the efficiency. Different network services were used for making the analysis more reliable. The system architecture remained constant throughout the scan and remained the same for every scan. a firewall that's also set up to accept all traffic. Because all vulnerability signatures were enabled and set up to look for services on all TCP ports, the scanner was able to accomplish that. Additionally, all tests fall under ethical standers. Every indication of a vulnerability is regarded in the same way: as proof that vulnerabilities exist. Most relevant vulnerability is being used we used a list of 20 flaws. High risk flaws where prioritized. 10 flaws were chosen to check whether the tools are giving wrong outcomes. The percentage of misleading results from authenticated scans was calculated by comparing all inaccurate results of unauthenticated scans with the corresponding output of each authenticated scan.

4.3.1 Ways of Validation

To make sure the selected tools perform efficiently in a way that we can get an authentic result all the selected six testing tools are undergone a quality check. All the tools selected can perform its assessments in selected operating system. Technically the selected tools can figure out the software faults, detecting the open port, check for the errors in configuration, can perform authenticated scan. Most of the testing tools have a vast database of pre-existing vulnerabilities. The (Figure *10*) below dissipates the quality check results.

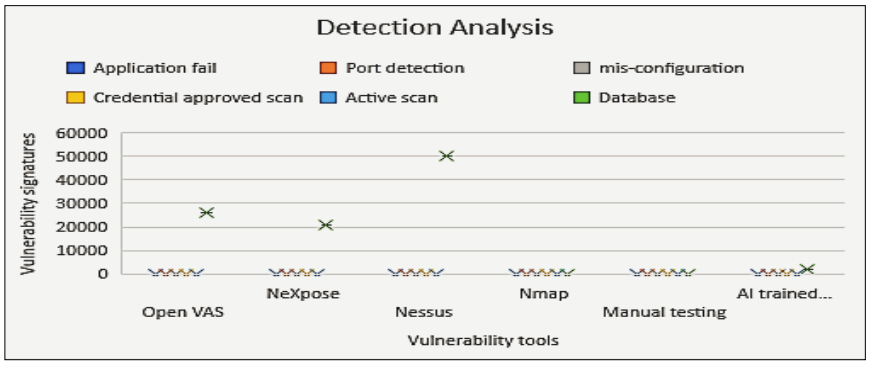


Figure 10:Selected Tool analysis

In finding the vulnerabilities most of the tools performed equally good, though to the reason the data was trained AI implied testing showed a more efficient results. The assessments have been done both as internal and external as expected when testing as an authenticated tester more vulnerabilities were detected.

4.3.2 Detection During External Scanning

The test results stood up to the mark. firstly, the eight targets were tested from the outside that means after gaining the IP address or hostname I tried to attack the systems from an external source. The reason was to make a comparison of tools while they perform the authenticated and unauthenticated tests. Started the assessment with manual testing though it was the most time consuming (Figure *11*).

The manual test was able to detect 12 critical and nearly 25 moderate error, mostly all tools or way on conduction showed a lot of minor errors. The reason for excluding this type was to increase the practicality in concentrating on more critical security errors that can harm the root files. Seeing the number of vulnerabilities signature, I expected Nessus will give the best results. But against the polls Open-VAS was able to detect 35 critical and moderate vulnerabilities whereas Nessus and NeXpose spotted 32 and 28 major faults respectively.

The Nmap gave the least test results as expected though it was having the least port vulnerability validations. The most successful testing was done by the trained tool which calculated majority of the fault which equals to 37 in all together. The AI used tool was trained by a portion using a portion of data collected from the manual test drive. The graphical view of the test data can be found below.

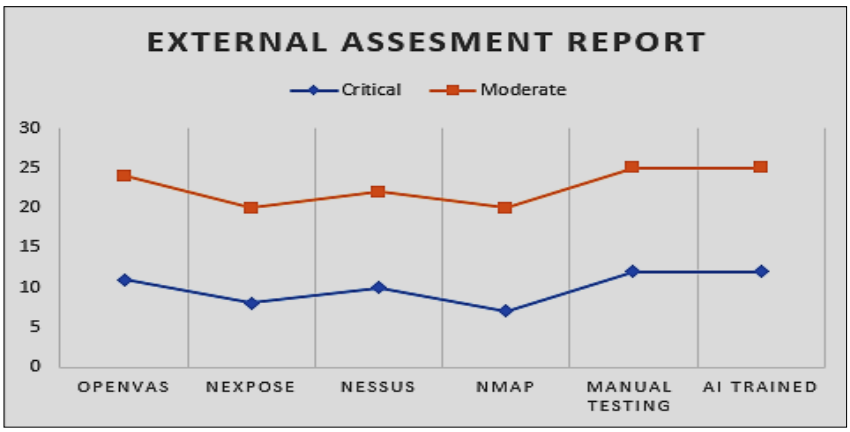


Figure 11:External testing results

4.3.2.1 Tools Performance: External Testing Phase

Efficiency and precision are one of the most important criteria for any testing tools especially when it comes to a security assessment[100]. AI trained tool which took its majority of data from manual testing was able to provide a precision of 82.2% , the test was able to detect 37 most critical vulnerabilities out of 45 configured errors, but when it comes to efficiency manual testing was only able to find 1 error in 5 minutes when calculating as a whole at the same time the trained tool was able to detect 21 vulnerabilities this makes a lot of performance gap, but from company perspective figuring out all vulnerabilities are prioritized, as they don't want to compromise on security[101], I believe in the coming era time take to troubleshoot an error can give rise to new complex errors. When it comes to already existing tools OpenVAS was able to detect nearly 3 vulnerabilities in a minute with a precision scale of 77.7% on the same line, NeXpose was only able to detect 62.2% of the vulnerabilities but the rate at which it was generating the result was notable, nearly 4 errors a minute but I think the tool should improve their database so that the precision can be up scaled. Far from every other tool Nmap was lagging to trigger out the errors. 60% was the precision when it took nearly 22 minutes to give the result (Figure *12*).

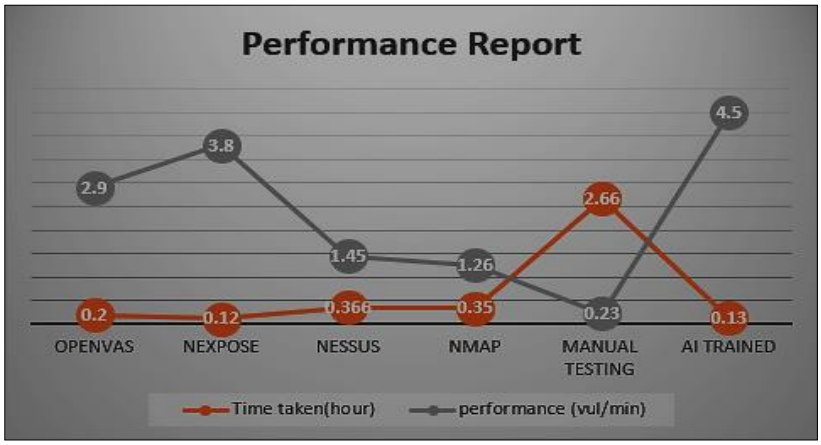


Figure 12:Efficiency during unauthenticated scan

In my point of observation AI trained tools performance was considerably better than any other tools but since getting the data for training and filtering out the anomalies out of the collected data is considerably tough, here it was comparatively easy though to the fact that the vulnerabilities was known before, and many recent common vulnerabilities were added in the database. The scenario will be entirely different and can take time in the initial phase of proper filtering and finding an efficient algorithm. Internal testing is considerably much easier though to the fact that the tester has the

authorized access to the internal system and network[102].

4.3.3 Findings From Internal Scanning

Now let's look at (Figure *13*)which shows the internal test results, where I expect the tools to find more vulnerabilities. The test was done in the same way but here I acted as an authorized tester, so I was logged in to the system as a user. This time the testing tools showed better results in case of finding vulnerabilities and performance. The trained tool showed better ability to find the vulnerabilities here too the trained tool was able to detect 310 major errors whereas the manual testing was able to detect a less of 280 vulnerabilities overall. Nessus was able to show 299 security errors, this was a bit unexpected because of the performance by the tool in the external assessment. This shows that a very updated database of vulnerabilities can offer a better result specially to reduce the false positive reports[103]. OpenVAS detected 214 critical system faults and NeXpose was able to find 238 vulnerabilities. In the internal assessments also Nmap detected 206 open port vulnerabilities, this is not only the lowest in case of total detection but when it came to critical vulnerabilities, they detected a very few where the system was configured with 110 actual critical vulnerabilities.

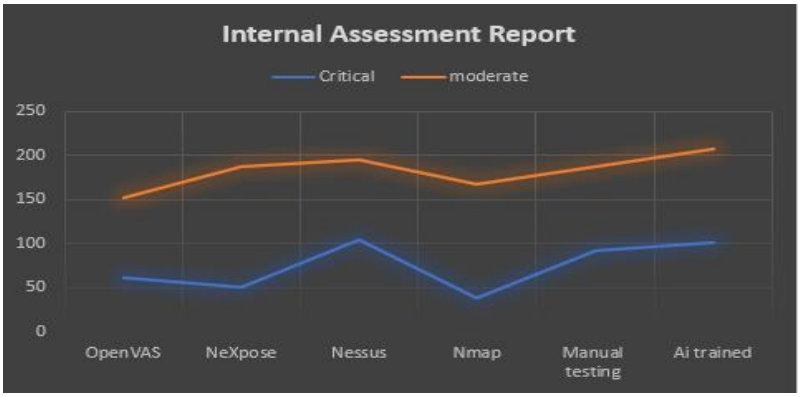


Figure 13:Authenticated scan results

4.3.3.1 Tools Performance: Internal Testing Phase

Looking from perspective of accuracy most of the tools performed in a decent way. The manual testing took a long of 240 minutes to find out errors of 8 servers/system. This was quite expected. Graphical representation given in fig shows that with 88.5% of the vulnerabilities detected by the AI trained tool with a detection speed of 10vul/min.

Manual testing was showing an efficiency of 1.16vul/min, considerably low than the other tools but the accuracy was high as expected which stood up to 80%.

The four selected automated tools did a decent assessment of the target. Nessus was able to spot out 85.4% of the major system vulnerabilities with an efficiency of nearly 7.56vul/min this was very close to Nmap which detected 6.43 vulnerabilities in a minute. Considering its 58.8% of accuracy other tools can be preferred moreover Nmap was only able to get access to uncontrolled system areas the reason is that it is a port scanner.

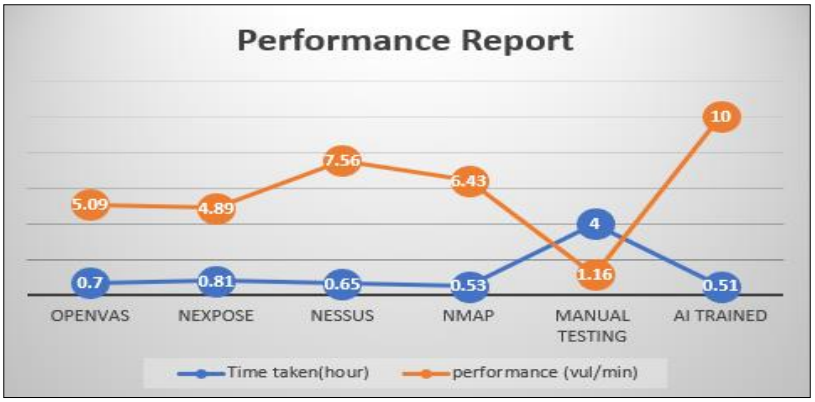


Figure 14:Internal testing tool performance

The performance of OpenVAS and NeXpose was very close, OpenVAS was able to detect 5.09vul/min and NeXpose was detecting 4.89 vulnerabilities at the same time. OpenVAS showed a precision of 61.1% compared to 68% of total detection by NeXpose but if we look that's not too different from each other (Figure *14*).

4.3.4 Comparison of test results

We can draw a result out of these, compared to external testing the tools were able to find more vulnerable faults that won't be the most interesting, but when it came to the performance even while the internal testing took more time compared to external testing the efficiency was high. AI trained tool showed an equal detection percentage while using the same data from manual test results, but when it came to internal testing even after using the same training from the input generated by the manual test case it gave better results than manual testing that too with a high efficiency. The most contradictory result was Nessus was providing better detection results than manual testing which is but deviating. That gave sight to conduct a false report analysis, will discuss a categorized detection report for Windows and CentOS

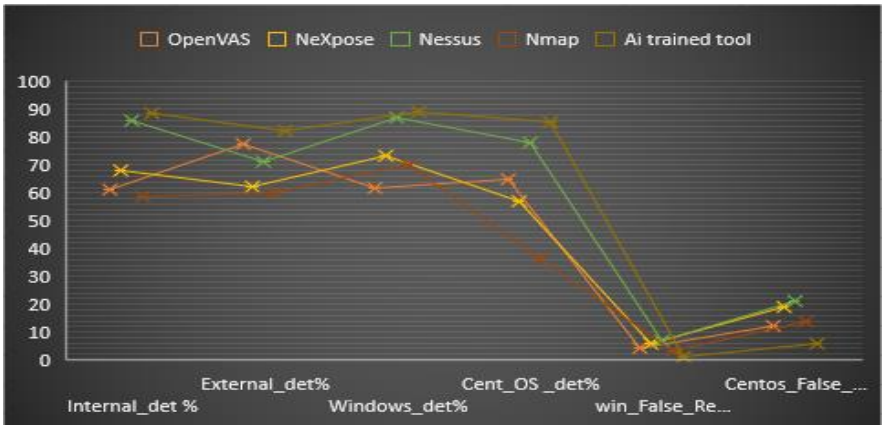


Figure 15:Detection and false positives

It appears that all the selected testing tools, namely Nessus, OpenVAS, Nmap, and NeXpose, were successful to some extent. To put it another way, roughly half of all network problems can be identified with these selected vulnerability detection tools. OpenVAS was able to detect a total of 69.4% of the vulnerabilities. NeXpose showed a trustable 65.0% detection rate. This shows the importance of scanning and how it can help lower a network's system vulnerabilities.

The results of the study show that vulnerability scanners have limitations in that they can identify only 65% of the weaknesses in a network or service. This precision problem is caused by many exploitable vulnerabilities, particularly when there are discrepancies in the findings of various testing technologies and when false negatives and false positives contribute to this issue[104]. Because of this, it's important to be careful and take steps to protect your system.

There are several ways to increase accuracy when screening servers, one of which is to provide privileges for the servers being tested[105]. Our testing tools were able to find 8 to 9 percent more vulnerabilities after a more thorough examination of the data. Nmap, for example, could detect 58.8% of internal issues; within the same host, it was able to identify 2% of additional vulnerabilities via external testing. Both internal and external evaluations of OpenVAS were about 15% different. This was a huge deviation (Figure *15*).

It is essential to use numerous tools and methodologies when evaluating servers for the most precise findings[106]. Our AI-trained tool, for example, was able to discover more vulnerabilities with greater efficiency. We found that, except for a few tools, most tests performed better during internal testing when we granted access to the tested servers. This allowed us to improve the performance of our tools.

However, the statistics may not be entirely accurate since we were only

able to use a limited number of measurements. If hackers use internal checking while administrators use external detection, network administrators have a significant advantage. But allowing unauthorized access with a simplified view of possible security holes would not only make it harder for network administrators to protect their infrastructure, but it would also make it less likely that an intrusion would work. Common Apache HTTP bugs made by third-party extensions like "ap get basic auth pw()" are often easier for scanners to find[107].

4.3.5 Tool Performance: In different Operating Systems

There is a difference in error rates between programs running on Centos and Windows, with more false alerts coming from Centos. However, this is not a significant difference. The codebase for Windows is larger after patches are applied, which may be the reason for these variances.

The analysis shows that automated vulnerability scanners may have difficulty detecting vulnerabilities in the Debian operating system due to the way that Debian developers often modify code while retaining the sequence number. This was found to be the case in 68% of instances. Given that Debian is a popular operating system, this presents a challenge for security researchers. It is possible that manufacturers of vulnerability scanners may be more likely to focus their future development efforts on Windows-based systems due to the larger potential market for these products[108]. Even though our testing tools were only able to identify 85 percent of CentOS vulnerabilities, they were able to identify approximately 5 percent of additional issues. This may be due to the fact that the majority of our testing tools are less likely to make mistakes when used on Centos.

4.3.5.1 Test environment

The network's design was created with a simulated machine. In a virtual environment, data loss and other performance issues may crop up. However, this vulnerability is more common in large virtual machines as opposed to smaller nodes, as was discovered during testing on a Windows system. The Nmap detection rate for vulnerabilities was 2.8 per minute. Analysis can only show how well a vulnerability scanner can detect problems in a certain network architecture.

4.3.5.2 Analysis: Limitations

This research only covers a small percentage of the vulnerabilities that are currently being studied. However, the system, applications, and threats of

the network infrastructure are of a different kind. Therefore, this research provides a useful indicator of the accuracy of the instrument.

4.3.5.3 Analysis: Performance difference

The findings from four different vulnerability scanners were analyzed. It was found that, on average, automated scanners were able to discover 73 percent of Windows vulnerabilities and 58.9 percent of Linux vulnerabilities. This means that the internal and external screening accuracy for Centos and Windows varies significantly. The automated tools used for reporting purposes yielded a large number of false positives in the absence of significant database changes. OpenVAS was found to be superior to other techniques, particularly in CentOS, due to its very low 12 percent false positive rate. Nessus was found to be the most dependable automated tool throughout an internal examination due to its high accuracy of 85% on the same server. However, precision declined to 71% during external testing. The high performance of this program was likely due to its database including a greater number of vulnerability signatures. The findings indicate that the AI-trained scanner was able to identify the majority of faults. However, the automated scan found just a tiny number of network vulnerabilities. Consequently, it is essential to properly understand the results of vulnerability scans and, if feasible, to supplement them with AI-supported verification.

This study was conducted on a limited number of systems, so further research is required to determine the effect of the findings on actual systems and servers. It would be interesting to investigate the factors that affected the accuracy of this research in further detail, given the 11% discrepancy between the Windows OS and the Cent OS. Even if reports show that the severity of vulnerabilities in a tool has changed, for example, if some vulnerabilities are rated as critical for the same problem while others are rated as less important, product managers need to be able to quickly evaluate the situation and make decisions.

The second objective is to offer clients the necessary knowledge to properly monitor and address vulnerabilities[109]. Consequently, it is crucial that scanners be seen as helpful and simple to use, although variances and misleading positive or negative findings might negatively impact this perception. When a company uses a Linux-based system, it can be very hard for the scanners to give an evaluation report because the rate of real threats is low and there are a lot of false alarms.

Furthermore, when a scanner affects the services of your network, it might be alarming. A prevalent issue, particularly when utilizing

unauthenticated scanning, is that it unintentionally launches a denial-of-service attack on a system[110]. This is more prevalent in applications running on Linux-based operating systems than on Windows-based operating systems, and it may cause a service termination in the real world.

Passive vulnerability scanning is an interesting area of study, but I was unable to explore it due to unanticipated obstacles. Companies that work with important infrastructure should give passive vulnerability scanning serious consideration. The ability to use passive and active scanning to evaluate the effects of routine operations is valuable.

4.3.6 Working towards a solution

There are only a few programs that offer specific analytic features and reporting capabilities, especially when it comes to performance. Some tools are able to deliver high performance while others are not, but the efficiency isn't awful either. Examining internal or external patterns of weakness can help improve scan findings or lessen the need for manual analysis. Obviously, these traits should also be considered and studied in depth. We are close to being able to deploy a cross-cloud security platform from which testers may choose the most recent vulnerabilities and assessments. This would be a substantial accomplishment and would make it much easier for testers to find the information they need.

Even if an endpoint rejects a payload and notices changes to parameters, a cross-cloud platform could still be used to take advantage of a vulnerability. Automated technology comes with a lot of risks, the most concerning being hackers gaining access to crucial systems or networks. If this happens, it could mean the end of your business as well as a lot of financial losses. To help reduce some of these risks, these solutions need to be set up by trained people who understand your needs.

Even though an automated technique was anticipated to cover the majority of security flaws in less time than a human approach, a tester would first focus on holes in business logic and subsequently attack vulnerabilities uncovered by automated tools. According to our investigation, the system could not identify logical problems, and even broken access was not recognized during external testing. But if we use the idea of a cross-cloud interface, we might be able to make a system that cuts down on the time and money needed to commit while giving us predictable result

4.3.6.1 Difficulties with False Report

The statistics reveal indisputably that the process of recognizing and filtering false positives is fraught with obstacles and provides a significant challenge. This is a really serious issue since the technique provides a formidable obstacle. This is very tough since it must be accomplished in a manner loaded with complexities. To accomplish it, one must adhere to a procedure that is laden with hardship. Spending additional time and effort on concerns that do not exist, as is the case in an industrial context, increases the strain and decreases the enjoyment of people who are not trained in security. Consequently, not only does the process of correcting the problem take longer, but hackers have more opportunities to exploit the system's vulnerability. This is quite likely to result in the company experiencing many difficulties in the future as a direct consequence of these issues. This has a high likelihood of occurring. Constant progress in all analogue and digital types of technology It is difficult to keep track of and regularly inspect them. They tend to become aggressive when they are not adequately managed to maintain control. It may be challenging to keep track of their whereabouts and conduct periodic inspections on a regular basis. Alternately, security officials are forced to check them individually, which takes a substantial amount of time and involves tremendous work from everyone.

In addition, the number of individuals who perform their professions from a remote place has grown, which increases the overall level of danger and decreases visibility. This is problematic because it makes it more difficult to keep track of who is doing what and where. The great majority of automated systems take extraordinary caution when determining whether to incorporate CVSS findings in their repository. This is likely because the CVSS was designed with the possibility of being excessively generic in mind. One of the primary factors contributing to the difficulty of issuing alerts is that businesses struggle to establish which types of vulnerabilities are most significant to their own organization. This is one of the primary reasons for the complexity of the challenge. Consequently, finding a solution to the notification problem would be far more difficult than anticipated. Since the CVSS score is just one component of a much wider framework, it does not immediately direct information security specialists to the most severe issue. The statistics indicate that distinguishing real positives from false positives is a challenging and time-consuming job. This is a serious issue that demands a complicated solution. Because of the required level of complexity, this is quite difficult. The technique that must be followed is somewhat challenging. Spending additional time and effort on vulnerabilities that do not exist may increase the workload and make it less

appealing for those who lack security expertise. Additionally, it is essential to appreciate the monetary worth of one's time in the present.

As the research progresses, I can see that one of the most troubling incidents while conducting vulnerability assessments was the triggering of false positives. So, while designing the platform for testing, we need to adopt some common approaches to tackle this issue. Let's have a closer look at how to tackle this issue.

4.3.6.2 Proposing Solution Using AI

In my opinion, one of the most important baselines is to properly monitor the target. The foundation of my cross-cloud AI model engine for risk assessments is the "expert system shell" concept. This is a way of limiting human input while conducting the assessment, but prioritizing can still be done by humans. The use of AI gave precision. An expert system programmer may explain only the code-line errors of a specific objective and correct them rather than finding ways to identify them in the future. The way may often be deduced by the expert shell system with the aid of other predefined rules.

The CCA engine is in charge of deciding whether or not a scan should happen when a vulnerability is found. When specific fixes are present, the logic for identifying bug fixes can become quite complex. As the scanner gathers more information and signatures, the CCA engine may adjust to the new intelligence. Centos only has a handful of security warnings in the recent release, including "CESA-2022:7008,6765" out of a total of roughly 3033.

During the unencrypted stage of a scan, a web server such as Apache or PHP may incorrectly represent itself as an older model. Nevertheless, if keys are provided and GUI-based data is available, the machine will consider the intelligence it knows about recent CentOS alerts. Apply source warning fixes and reject the less reliable signature. This prevents false-positive probes into the vulnerabilities that have been patched.

4.3.6.3 Step Towards Reducing False Positives

Updating the database with the most critical as well a recent signature is another way to reduce the false-positive. While we did the VA, Nessus was giving considerably better detection due to is high signature database. Prioritizing the signatures is an important aspect to make the machines understand which vulnerabilities should be tackled and what point of time. Using an analysis of risk, we found way to prioritize these signatures. To conduct a study on, among other relevant issues, the tactics used by hackers

and the accessibility of exploit kits. In addition to the advances in computation, all these criteria are incorporated in the Real Risk score, including the approach that focuses on time periods, the required level of knowledge to expose the weakness, and the comprehensive evaluation tags that you may apply based on your individual priorities.

This means that we will be able to protect against a lot more zero-day exploits since we will be able to find and use new flaws quickly. In addition to exploring a vast array of TCP and UDP services, the mission also includes the discovery of a secure sockets layer (SSL), a domain name system (DNS), and the hypertext transfer protocol (HTTP). Compared to the multitude of other scientific projects now underway, this one stands out as particularly remarkable. Find all of the web resources, including the ones you are familiar with and any new ones you may encounter, and arrange them in a single folder for easy access. This applies to both the ones you are acquainted with and others you may encounter in the future. By doing so, you will guarantee that you are acquainted with all that is accessible on their website. It is likely that you are unaware of the vast array of resources that may be acquired via mergers and acquisitions. This is a very acceptable inference to make. Given the complexity of the topic at hand, this is quite appropriate. This is one of the potential results.

The procedure is straightforward and requires no trespassing on private networks to be successful. This is due to the fact that the majority of the time spent analyzing the data is spent correcting inaccuracies. This is because the primary objective of the technique is to identify the data components that contain errors. In addition, there is not even the slightest chance of any associated danger in any way, shape, or form. Indeed, this is true in every circumstance. removes the remaining process steps from the cloud without severely impacting the server that is meant to serve as the activity's primary hub, if applicable. Through the process of classification, the ability to prioritize the repair of vital assets over less significant ones, such as those that produce merely false positives or minor issues, is gained. This ensures that vital assets are protected from potential threats. This guarantees that the crucial item in issue gets the necessary maintenance and repairs without delay. Therefore, key resources will be protected against future threats. In this manner, the afflicted vital asset will get the necessary maintenance and repairs in the shortest time feasible. As soon as possible, this will be completed. Therefore, vital resources will not be vulnerable to future threats. Consequently, you may be certain that the object in issue will always remain secure, regardless of its size or shape. As a direct result of this approach, valuable assets are preserved against any future threats that may develop.

4.3.7 Use of AI In Reducing Errors

Artificial intelligence always helps in rectifying human errors , analyze the targeted result, and provides the best solutions for the client. Following the real-time analysis of data by AI systems, or more particularly, machine learning, the system's vulnerabilities are graded according to the danger they represent. Following the examination of data in real-time by artificial intelligence algorithms, this rating is determined. The bulk of the time, artificial intelligence is the driving force behind systems, and these systems virtually always include capabilities for regulating risks and vulnerabilities inside the system. With these features, you can do a deep analysis of hundreds of different attack paths and threats, as well as a separate analysis of the risk associated with each attack path and threat.

When attempting to identify vulnerabilities using traditional techniques, a tester should err on the side of caution. This is because the technique often produces a substantial number of false alarms. This is because the procedure aims to identify areas of vulnerability. Security teams often use technologies powered by artificial intelligence to determine the likelihood that a discovered danger is real. This action is taken to safeguard against possible risks. The systems consider a range of characteristics, including the detection technique that triggered the process of warning consumers about the vulnerability. This is only one of the various aspects considered; there are many more. Utilize machine learning to analyze the user's behavior in search of anomalies that might indicate the presence of a previously unknown security problem[111]. Using technologies that use artificial intelligence may be very beneficial for determining which assets are crucial to the running of the organization and, consequently, need a greater degree of protection. This may be performed using technology based on artificial intelligence. The system may compare a huge number of assets against one another, calculate a mean value, and then inform the user of any assets whose values considerably differ from the mean. For a long time, security teams have yearned for the ability to handle vulnerabilities based on the circumstances in which they were identified. AI technologies now provide this power. By including a thorough knowledge of the setting in which an asset is held, these methods make it feasible to calculate a risk score that is more accurate. In other words, they enable the calculation of a risk score. For instance, it may seem that a potentially vulnerable resource has been completely detached from the network, but this may not be the reality. It is likely that sentiment analysis might be used to identify system vulnerability trends. AI algorithms are able to gather data from online cybersecurity discussion forums and

media websites, evaluate that data, and identify exploitable vulnerability patterns[112]. AI methods such as neural networks and natural language processing enable the recognition of both positive and negative emotions and the risk assessment of a document's content[113].

Utilizing the appropriate artificial intelligence tools may make it much simpler to discover, prioritize, and fix security risks[114]. This may also save time and decrease the quantity of necessary human labor by decreasing the amount of physical labor. As a direct result of this reality, it is possible to establish a robust security posture, especially when paired with a well-planned strategy. This enables the development of comprehensive vulnerability management.

CHAPTER 5

5.1 AI-Cloud base testing Architecture

From all the above research and analysis, we can say that the development of an assessment system using the technology of cloud computing and virtualization techniques using Artificial intelligence can increase the performance, reduce the risk and can assure a much secure digital working environment.

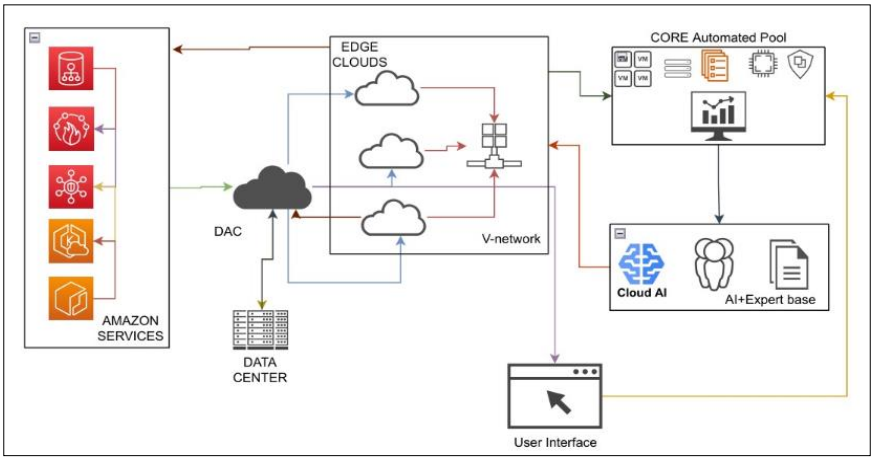


Figure 16:Base model Architecture

The AI integrated model have five main components (Figure *16*) we use a pool of AWS services that is pulled into a private Demand Automation cloud that is connected to our datacenter. This done for the safety of the user credentials and to keep a record of our analyzed data and vulnerabilities. This private cloud is further connecter on to the recent technology of edge clouds this is made possible of the fast-growing 5G telecom technology Edge computing is the practice of putting data processing and storage closer to the origins of the data[115]. This notion, known as "edge computing," exists

within the broader topic of distributed computing[116]. Consequently, the performance of the distributed computing model is enhanced until it reaches parity with the performance of the classic model of central processing units. The phrase "edge computing" may be seen as a reference to this concept. It is intended that this would decrease reaction times without reducing bandwidth. Why we used this concept will be explained further. Other major component of the model is the core automated pool which contains the vulnerability signatures, analytical algorithms a physical data storage unit.

Virtual firewall for filtering out the threats console is provided to make quick internal change to increase the efficiency and precision of the vulnerability detection. The core pool also consists of an initial report console.

The other most important part of the model is the AE-base which consists of live vulnerability assessment and will be cross checked by experts for logical faults especially in the code lines. A secondary report is made that is send to the DAC through edge cloud. Final check for the recent vulnerabilities from our live platform is done there and a report is sent to the user interface. Discussing some general but some key points before diving deep to the process.

5.1.1 Data collection

using network monitors to gather data from nearby as well as far-off networks, and then securely delivering that data to the central automated pool server. Edge clouds are often installed behind the client's firewall, where they can monitor the environment for any changes and securely communicate encrypted data to the core pool platform in order to be analyzed. In certain circles, edge clouds are also referred to as "edge computing." The key ideas listed below served as sources of inspiration for the creation of the collectors' designs: Only those with permission to administer the system are allowed to set up analyzers. There is no need to be concerned about this subject at all since very advanced encryption methods are being used to preserve all data that is being sent. Every piece of data that is stored on an edge cloud goes through its own individual process of monitoring and security before it is sent to the server. This happens before the data is delivered. The collector will be requested to supply both an activation key and a fingerprint at the first step of the process for setting everything up.

The activation key must be used by a user of the AE-based platform in order to establish a connection between the edge cloud and the data that belongs to the company. By using a handshake in combination with a known shared secret, the DAC platform is able to validate this information derived

from the edge cloud authentication key. This enables the platform to carry out the action in a practicable manner. Without initially establishing a TLS connection between them, the AE base and the DAC platform will be unable to interact with one another through HTTPS.

For instance, in order for certificates to be recognized by edge clouds, they must first have their verification certificates validated by an authenticator that is listed in the AWS cloud directory. As a direct consequence of the successful completion of the challenge-response handshake, the Edge cloud is now ready to take Certification commands sent from the DAC platform. Edge clouds will never be able to avoid connecting to the DAC platform because they are unable to get past the client firewalls that are responsible for that. This will prevent them from avoiding the connection.

The core automated pool is the one that is responsible for monitoring how close they are getting to attaining their target. This duty was given to them as a tag. The edge cloud is in charge of receiving real-time system information from your assets and then delivering it back to the AE base so that it can be subjected to further analysis. This is done so that DAC endpoint awareness and detection may be enabled. This data collection encompasses not only the essential identifying particulars of the item in question but also the relevant records and processes as well.

5.1.2 Authentication

The AE-based authentication system puts a substantial degree of dependence on client authentication that may be supplied by TLS as its primary form of defense. This is done to ensure the highest level of security possible. Incorporating a private SSL key that acts as a representation of the target source as part of the execution of the evaluation technique is one of the components of that approach that is included. The implementation of the assessment technique incorporates this component as a necessary aspect of the process. By combining the use of this key with the user authentication console, it is possible to successfully authenticate oneself and provide approval to the client. If you use the user authentication console, you will have the ability to do this task without any problems.

The authentication of the AE-base is performed not via the use of TLS client authentication but rather through the application of an API token. When it is feasible to do so, the client is able to establish a direct connection with the AE-base by interacting with the AE-base via a proxy, which is accomplished by using the user console. This connection is established when it is possible to do so. It is possible to carry out this phase in order to identify

the order of significance of the use cases and determine whether or not any configuration is required at this point. To do this, it is necessary to determine whether any setting change is needed.

Edge cloud environments and user console environments both make use of the automated pool in order to discover the systems that are being targeted and to establish a remote connection to those systems. It is the responsibility of the automated pool to run vulnerability scans of the networks, and after the scans have been completed, the automated pool will transmit its findings to the AE-base. Engines have the capability of being dispersed over several networks, which may include both public and corporate networks, as well as cloud providers. Even if it is located in close proximity to the target, the edge cloud may still be able to give strategic views of your network from the viewpoint of an attacker. A user may set the platform to carry out approved scans to seek for software programs and packages, as well as validate patches, by using the user console. Checking for holes in security in this manner is something that can be done.

The automated pool makes use of virtual machines so that it is possible to scan private web applications that are not available to the public through the internet. This makes it possible for the automated pool to scan private web applications. This is done so that the automated pool will function in the most effective manner possible. After four stages of screening and verification, the results are returned to the same user panel via automated pool VMs. These VMs create connections to the web apps that have been developed by the client. The automated pool console platform is unable to initiate a security scan because it is unable to get past the firewall that the client has installed. This is due to the fact that the scan needs consent from the customer.

5.1.3 Credential Management

It is not possible to decipher passwords by using the information that is stored in the cloud because the platform encrypts all of the credentials that it uses to collect data from the local environment of the target in such a way that it is impossible to decode the passwords. As a result of this encryption, it is impossible to decode the passwords. This is due to the fact that the cloud storage service makes it hard to read passwords. Unauthorized individuals will be unable to discover the account's passwords as a result of this security precaution. Following the generation of a one-of-a-kind combination of public and private keys in response to each request, that combination of keys is then dispersed among the different environments.

The private key is stored in a safe location on the physical device that the

automated pool makes use of, while the public key is uploaded to a storage service that is hosted in the cloud. This makes it possible for the public key to be accessible from any location on the globe. The contents of the private key are put into the V-encryptor, which is located inside the automated pool, before being written to the local disc. This ensures that the contents of the private key are secure. In order to guarantee the safety of the private key, this step must be taken. This ensures that unauthorized parties do not gain access to the information stored in the private key.

The private key can only be accessed by active, live vulnerability tracking that makes use of AI and Edge Cloud and maintains a rapid connection to the DAC. This is due to the fact that the private key can only be decrypted with the assistance of information that has been successfully obtained through successful communication with the DAC administrator. This is due to the fact that the private key can only be decrypted with the aid of information that has been successfully collected via successful contact with the DAC administrator. This is a result of the fact that the private key can only be decrypted with the aid of information that has been successfully gathered via successful interaction with the DAC administrator.

The reason for this limitation is as follows: DAC administrator When a credential is added to the DAC, the password for each edge cloud is encrypted using RSA Public Key Infrastructure and keys that are 4096 bits long. This happens whenever a credential is added to the DAC. Utilizing the public key that is connected to each VM in order to decrypt the encrypted password is the method that is used. Each of these stages, including the one you're currently in, will be completed immediately if the necessary credentials are provided. After that, a database structure that is specific to the client is applied to it before saving it to the database. This step occurs after the first step. After the credential has been successfully saved, the edge cloud will no longer have access to the cleartext version of the credential, and this access will be revoked after the credential has been securely stored.

When it comes time for the automated pool to make use of a credential, the encrypted password is retrieved from the DAC and then decoded within the collector using the assistance of the private key that is associated with the VM. Because the password is entered, used, and then removed from memory, it is never actually stored anywhere on the hard drive. This is due to the fact that the password is entered, used, and then removed. Because the client is the only person who is authorized to set up event sources on the VM in question, the VM is only allowed to seek access to the credentials that are necessary for those event source

5.2 Working principle

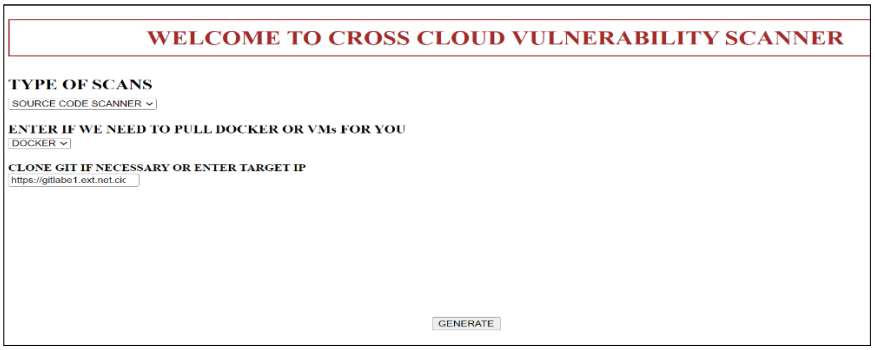


Figure 17:Sample user interface

The Client will be able to begin a scan using the user interface (Figure *17*), which is connected to an automated pool consisting of an automatic vulnerability scanner, an accompanying filter, and the backend admin panel where the database may be updated with vulnerability signatures. Client will have the opportunity to do so using the user interface that is linked to the automated pool. Moreover, a link exists between the user panel and an automated pool that is likewise connected to another automated pool. This relationship is made feasible by the automated nature of the user interface. Due to its connection to an edge cloud, this pool is able to update its database with the most recent information on active vulnerabilities and any false positives that have been triggered during the most recent time frame. In addition, this pool may monitor and log false positives that have been triggered.

This information will be compared using our analytical tool, and if any erroneous reports are discovered, our system will remedy the errors and send the findings to the AE-base. If any inaccurate reports are discovered, our system will resolve the issue by correcting the errors. If any erroneous reports are detected, our system will resolve the problem by rectifying those reports' mistakes. This procedure will continue until there are no more erroneous reports, at which time it will end. When a buyer seeks an acceptable answer to their demands, they often have three fundamental options to choose from. There are several options available, including network-based, host-based, and application-based scanners. When it comes to application-based computing, scanning is not a possibility during the development process if the software in question is performed on an external host or network, so we needed to clone the validated code. This is due to the fact that scanning requires software to have access to an internal host or

network. The only option accessible to the user is to get the source code separately from a git repository and then submit it to the platform, where it will be subject to a search (Figure *18*). This is the only accessible option to the user. There are no other alternatives available to the customer. An instantaneous exception request is sent to the AE-base at this point. At this stage in the operation, an authentication token is also generated if the client desires that only certain areas be scanned and configured, or if they want to place restrictions on the scan. This happens when the client wishes to limit the scope of the scan.

Taking this course of action makes sense given that the customer may choose which sections of the document they want scanned and altered according to their specifications. Once the evaluation of the request undertaken by the expert on the backend has been finished, the evaluation findings will be sent to the user console through the DAC. This will occur after the procedure is complete.

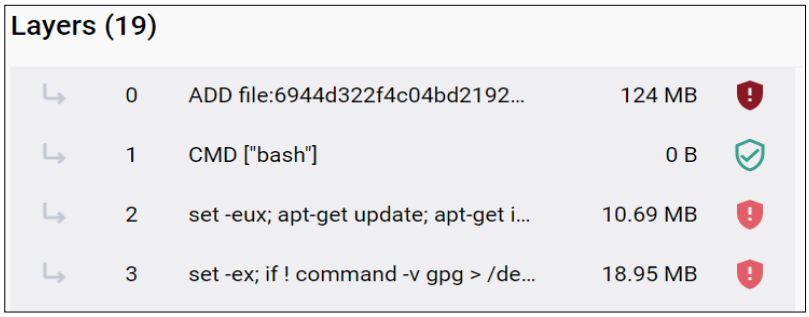


Figure 18:Cloning git

Whenever an exception is created, a duplicate will be stored in the DAC so that it may be handled at a later time. Since there is no other choice, using the authentication token is the only method available for altering the scan. There is just one technique to alter a scan if the client determines at any moment that it has to be revised. Upon completion of the upgrade, there will be two stages of verification between the Core and the DAC. The cause for this may be discovered in the fact that the following elements have contributed to this situation: This is crucial to one's security (Figure *19*) and safety for a variety of reasons, and as such, it should not be neglected.

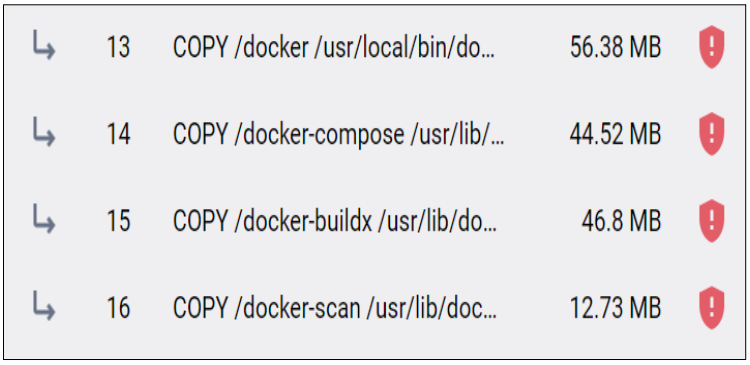


Figure 19:Second phase of scan started

5.2.1 Second phase of validations

Following the completion of the first phase of the automated verification procedure, the initial report is then sent to AE Base. At AE Base, the report of the vulnerability assessment is checked with the assistance of a platform that is operated by artificial intelligence that has been trained. This is done in order to ensure the accuracy of the report. After this stage, the findings of the first report are classified as either very effective, moderately effective, or not very successful, and it is anticipated that more than half of the erroneous reports will be removed (Figure *20*). Once that has happened, security auditor evaluates the report after it has been vetted. This stage is required since no validation will even come close to being finished until the results are cross-checked by a panel of experts. Until this step is taken, no validation will even come close to being finished.

	Package	Vulnerabilities
>	debian/vim 2:8.2.2434-3+deb11u1	3 C 102 H 21 M 1 L
>	debian/python3.9 3.9.2-1	3 C 6 H 3 M 0 L
>	debian/pcre2 10.36-2	2 C 0 H 0 M 0 L
>	debian/expat 2.2.10-2+deb11u3	1 C 1 H 1 M 0 L
>	debian/db5.3 5.3.28+dfsg1-0.8	1 C 0 H 0 M 0 L
>	debian/libtasn1-6 4.16.0-2	1 C 0 H 0 M 0 L

Figure 20:Detected vulnerability

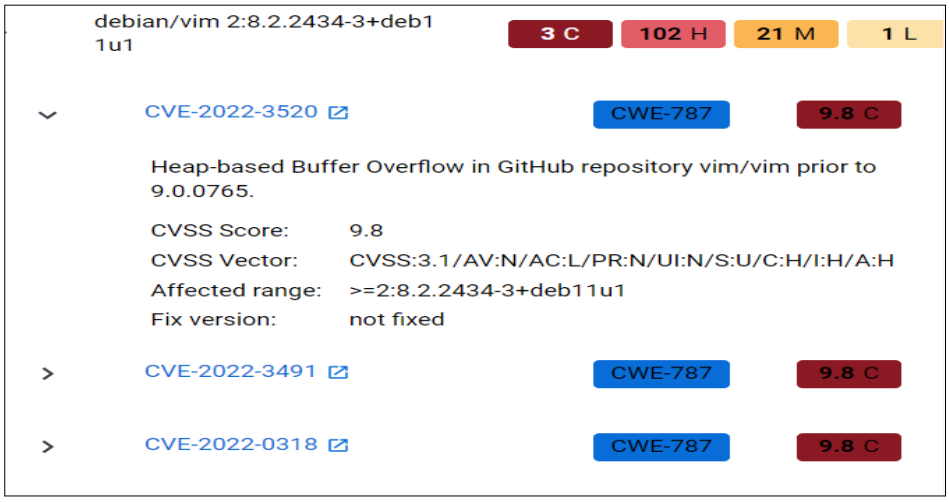


Figure 21:Trained model detecting CWE-787

During the second stage of the procedure, the primary emphasis is on the development of a vulnerability report that is far more relevant to circumstances that may arise in the real world. After that, the data is encrypted using a special procedure before it is sent to the edge clouds. Before the authentication token is delivered to the DAC for further processing, it is first authenticated on the edge cloud to ensure that it is the same as the encryption key. Only then is it sent to the DAC for further processing. After that, it is finally transferred to the DAC so that it may undergo additional processing. In a way that was not possible in the past, Amazon's services may now be used in an effortless and efficient manner after being incorporated into the digital automated cloud. As a result of the DAC's integration with Amazon's services, the firm is now able to deliver real-time cross-validation using the CWE-provided common vulnerability score. This became possible because of the integration. This integration has been completed in a form that makes it possible for Amazon to provide this service to its customers. After completion of the validation and creation of the final report, it will be possible to eliminate false positives and provide an accurate vulnerability assessment report (Figure *21*).

5.2.2 Conclusion of study Model

Following the conclusion of the entire study, this will be the case. While a copy of the report that has been downloaded from the cloud is supplied to the user interface, a second copy is stored in the data center for use in further analytical procedures. This provides for more adaptability. This allows for a higher degree of flexibility and efficiency. The client must pass over the

authentication key at the conclusion of the procedure so that the DAC may check its validity. This occurs at the very conclusion of the procedure.

Since the onset, we have emphasized that no system is entirely fault-tolerant, and our effort is a component of a wider push to increase cyber security in conjunction with the implementation of the 5G network and the edge cloud concept. As a result of the emergence of new technologies, it is now of the utmost importance to take steps to strengthen the security of user data. This technique has its own limits and downsides, particularly when it comes to the identification and removal of false positives from the assessment report. The frequency of false alarms is reduced, but more training is required to obtain the intended result of a 100 percent accuracy rate and almost no false alarms.

CHAPTER 6

6.1 CONCLUSION

The research mainly focused on the data breaches within the security practices . Studies showed that vulnerability detection is one of the main processes during the act of penetration testing . Majority have a contradictory understanding that penetration testing and ethical hacking are same. Technically penetration testing is a process included in ethical hacking. Focusing on this point, any issues that anchor during penetration testing can affect the performance of ethical hacking. Every issue in technology can be troubleshooted if we can find the root cause of the problem. The number of data breaches happening is showing a huge spike as times moves forward. Most of the tools we selected for the purpose of vulnerability assessment to analyze the efficiency came up with moderate results and with some false positives this can be a serious issue.

Analyzing code of ethics and to what extend the rules are followed was the next part of research. For these responses from experts specially someone with certification or expertise was selected. This survey gave some very important findings. Comparison of code of ethics of different security organizations was performed.

Mentioning of the most critical Vulnerabilities is an add on to the paper. Three most detected vulnerability in recent years have been stated. The ways of detection and rectification have been mentioned.

Deriving a cloud security model was done this was compared with the automated testing tools to check the efficiency and performance of the environment. Mostly the comparison showed a greater efficiency and performance of the trained tool

Finally, the success of every trained tool lies in the quality of the dataset

selected. The usage of recent vulnerabilities has been done. The efficiency of the model will show deviated results due to the reason of its less database. Even considering this fact the performance of the model in detecting very critical exploits will be near to perfect. Data categorization show be done to increase the performance when it comes to a different environment

6.2 FUTURE DEVELOPMENT POSSIBILITY

The developed model was tested only in specific test systems. The results can be deviating when it comes to real world. So, after the whole project has been performance efficiency should be evaluated. The code of ethics has been described but working towards a unified mandate is possible with reduced efforts. Put the start towards near to zero false positive vulnerability testing for this virtual cloud platform was developed the back end is supported by a collaboration of testers and AI trained tools . The training database was taken from CWE , this can make it possible to find the very critical errors, but I suppose, with a wider database the efficiency and performance can be leveled. The cloud platform is running in company private platform , making it accessible for normal user should be studied and it will be very useful tool for the future

REFERENCE

[1] A. Nordeen, *Learn Hacking in 24 Hours*. Guru99, 2020.

[2] S. Patil, A. Jangra, M. Bhale, A. Raina, and P. Kulkarni, "Ethical hacking: The need for cyber security," presented at the 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), 2017, pp. 1602–1606.

[3] K. Graves, *CEH certified ethical hacker study guide*. John Wiley & Sons, 2010.

[4] T. Grance, J. Hash, and M. Stevens, *Security considerations in the information system development life cycle*. US Department of Commerce, Technology Administration, National Institute of ..., 2004.

[5] "Figure 1: The Security Life Cycle," *ResearchGate*. https://www.researchgate.net/figure/The-Security-Life-Cycle_fig1_262726769 (accessed Dec. 10, 2022).

[6] M. Ashraf, A. Zahra, M. Asif, M. B. Ahmad, and S. Zafar, "Ethical Hacking Methodologies: A Comparative Analysis," presented at the 2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC), 2021, pp. 1–5.

[7] G. Kaur and N. Kaur, "Penetration Testing--Reconnaissance with NMAP Tool.," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 3, 2017.

[8] M. Tabassum, O. Muscat, T. Sharma, and S. Mohanan, "Ethical Hacking and Penetrate Testing using Kali and Metasploit Framework," 2021.

[9] J.-H. Hoepman and B. Jacobs, "Increased security through open source," *Communications of the ACM*, vol. 50, no. 1, pp. 79–83, 2007.

[10] J. Jose, T. T. Tomy, V. Karunakaran, Anjali Krishna V, A. Varkey, and Nisha C.A., "Securing passwords from dictionary attack with character-tree," in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, India, Mar. 2016, pp. 2301–2307. doi: 10.1109/WiSPNET.2016.7566553.

[11] S. Sharma, P. Zavarsky, and S. Butakov, "Machine Learning based Intrusion Detection System for Web-Based Attacks," in *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, Baltimore, MD, USA, May 2020, pp. 227–230. doi: 10.1109/BigDataSecurity-HPSC-IDS49724.2020.00048.

[12] G. E. Rodríguez, J. G. Torres, P. Flores, and D. E. Benavides, "Cross-site scripting (XSS) attacks and

mitigation: A survey," *Computer Networks*, vol. 166, p. 106960, Jan. 2020, doi: 10.1016/j.comnet.2019.106960.
[13] I. Vayansky and S. Kumar, "Phishing–challenges and solutions," *Computer Fraud & Security*, vol. 2018, no. 1, pp. 15–20, 2018.
[14] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of service attacks in wireless networks: The case of jammers," *IEEE Communications surveys & tutorials*, vol. 13, no. 2, pp. 245–257, 2010.
[15] S. Khanna and H. Chaudhry, "Anatomy of compromising email accounts," presented at the 2012 IEEE International Conference on Information and Automation, 2012, pp. 640–645.
[16] M. Wazid, R. Sharma, A. Katal, R. Goudar, P. Bhakuni, and A. Tyagi, "Implementation and embellishment of prevention of Keylogger spyware attacks," presented at the International symposium on security in computing and communication, 2013, pp. 262–271.
[17] F. Gandhi, D. Pansaniya, and S. Naik, "Ethical Hacking: Types of Hackers, Cyber Attacks and Security," *International Research Journal of Innovations in Engineering and Technology*, vol. 6, no. 1, p. 28, 2022.
[18] K. H. Kwon and J. Shakarian, "Black-hat hackers' crisis information processing in the darknet: A case study of cyber underground market shutdowns," in *Networks, Hacking, and Media–CITA MS@ 30: Now and Then and Tomorrow*, Emerald Publishing Limited, 2018.
[19] M. A. Shlyakhtunov, "White-Grey-Black Hat Hackers Role in World and Russian Domestic and Foreign Cyber Strategies," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 8, 2021.
[20] L. Radoilska and E. Ceva, "Ethical Theory and Moral Practice at 24," *Ethical Theory and Moral Practice*, vol. 24, no. 1, pp. 1–3, 2021.
[21] D.-O. Jaquet-Chiffelle and M. Loi, "Ethical and unethical hacking," in *The ethics of cybersecurity*, 21st ed., Springer, Cham, 2020, pp. 179–204.
[22] J.-P. A. Yaacoub, H. N. Noura, O. Salman, and A. Chehab, "A Survey on Ethical Hacking: Issues and Challenges," *arXiv preprint arXiv:2103.15072*, 2021.
[23] Y. Li and Q. Liu, "A comprehensive review study of cyber-attacks and cyber security; Emerging trends and recent developments," *Energy Reports*, vol. 7, pp. 8176–8186, 2021.
[24] Z. Terneva, M. Nenova, V. Terneva, I. Vladimirov, and D. Nikolova, "Cyberattack types-methods and technics for protection of communication resources," presented at the

2022 57th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST), 2022, pp. 1–4.
[25] R. Shimonski, *CEH v9: Certified Ethical Hacker Version 9 Study Guide*. John Wiley & Sons, 2016.
[26] D.-O. Jaquet-Chiffelle and M. Loi, "Ethical and unethical hacking," in *The ethics of cybersecurity*, Springer, Cham, 2020, pp. 17–20.
[27] S. Nicholson, "How ethical hacking can protect organisations from a greater threat," *Computer Fraud & Security*, vol. 2019, no. 5, pp. 15–19, 2019.
[28] P. Engebretson, *The basics of hacking and penetration testing: ethical hacking and penetration testing made easy*. Elsevier, 2013.
[29] P. Vats, M. Mandot, and A. Gosain, "A comprehensive literature review of penetration testing & its applications," presented at the 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), 2020, pp. 674–680.
[30] A. Kanunje, S. Mahale, N. Khare, P. Daund, and A. Deshmukh, "Penetration Testing and Vulnerability Assessment".
[31] C. N. Shivayogimath, "An overview of network penetration testing," *International Journal of Research in Engineering and Technology*, vol. 3, no. 07, p. 5, 2014.
[32] I. Yaqoob, S. A. Hussain, S. Mamoon, N. Naseer, J. Akram, and A. ur Rehman, "Penetration testing and vulnerability assessment," *Journal of Network Communications and Emerging Technologies (JNCET) www. jncet. org*, vol. 7, no. 8, 2017.
[33] S. Shah and B. M. Mehtre, "An overview of vulnerability assessment and penetration testing techniques," *Journal of Computer Virology and Hacking Techniques*, vol. 11, no. 1, pp. 27–49, 2015.
[34] S. Nidhra and J. Dondeti, "Black box and white box testing techniques-a literature review," *International Journal of Embedded Systems and Applications (IJESA)*, vol. 2, no. 2, pp. 29–50, 2012.
[35] S. Nidhra and J. Dondeti, "Black box and white box testing techniques-a literature review," *International Journal of Embedded Systems and Applications (IJESA)*, vol. 2, no. 2, pp. 19–30, 2012.
[36] S. Acharya and V. Pandya, "Bridge between black Box and white Box–gray Box testing technique," *International Journal of Electronics and Computer Science Engineering*, vol. 2, no. 1, pp. 175–185, 2012.
[37] D. Zissis and D. Lekkas, "Addressing cloud computing

security issues," *Future Generation computer systems*, vol. 28, no. 3, pp. 583–592, 2012.
[38] D.-O. Jaquet-Chiffelle and M. Loi, "Ethical and unethical hacking," in *The ethics of cybersecurity*, Springer, Cham, 2020, pp. 100–110.
[39] G. Morgan and B. Gordijn, "A care-based stakeholder approach to ethics of cybersecurity in business," *The ethics of cybersecurity*, p. 119, 2020.
[40] T. Jordan and P. Taylor, "A sociology of hackers," *The sociological review*, vol. 46, no. 4, pp. 757–780, 1998.
[41] P. Formosa, M. Wilson, and D. Richards, "A principlist framework for cybersecurity ethics," *Computers & Security*, vol. 109, p. 102382, 2021.
[42] R. Hartley, D. Medlin, and Z. Houlik, "Ethical hacking: Educating future cybersecurity professionals," presented at the Proceedings of the EDSIG Conference ISSN, 2017, vol. 2473, p. 3857.
[43] K. Milberry, "Hacking for social justice: The politics of prefigurative technology," in *(Re) Inventing the Internet*, Brill, 2012, pp. 109–130.
[44] B. Dupont, A.-M. Côté, C. Savine, and D. Décary-Hétu, "The ecology of trust among hackers," *Global Crime*, vol. 17, no. 2, pp. 129–151, 2016.
[45] L. Floridi, "Information ethics: On the philosophical foundation of computer ethics," *Ethics and information technology*, vol. 1, no. 1, pp. 33–52, 1999.
[46] S. A. Saleem, "Ethical hacking as a risk management technique," presented at the Proceedings of the 3rd annual conference on Information security curriculum development, 2006, pp. 201–203.
[47] N. Bates, "Comparing Cyber Weapons to Traditional Weapons Through the Lens of Business Strategy Frameworks," *Egham, United Kingdom: Royal Holloway, University of London*, 2020.
[48] K. Arapi, "The healthcare industry: Evolving cyber threats and risks," 2018.
[49] M. Siponen, "A pragmatic evaluation of the theory of information ethics," *Ethics and Information Technology*, vol. 6, no. 4, pp. 279–290, 2004.
[50] L. C. Botterill and A. Fenna, *Interrogating public policy theory: A political values perspective*. Edward Elgar Publishing, 2019.
[51] M. Bunge, "The philosophical richness of technology," presented at the PSA: Proceedings of the biennial meeting of the philosophy of science association, 1976, vol. 1976, no. 2, pp. 153–172.
[52] R. Hirschheim, H. K. Klein, and K. Lyytinen, *Information*

systems development and data modeling: conceptual and philosophical foundations, vol. 9. Cambridge University Press, 1995.
[53] R. Luppicini, *Ethical impact of technological advancements and applications in society*. IGI Global, 2012.
[54] C. Mitcham, *Thinking through technology: The path between engineering and philosophy*. University of Chicago Press, 2022.
[55] M. Bunge and R. Boudon, *The sociology-philosophy connection*. Routledge, 2017.
[56] M. Bunge, "The philosophical richness of technology," presented at the PSA: Proceedings of the biennial meeting of the philosophy of science association, 1976, vol. 1976, no. 2, pp. 143–161.
[57] M. Franssen, G.-J. Lokhorst, and I. Van de Poel, "Philosophy of technology," 2009.
[58] B. C. Stahl, G. Eden, and M. Jirotka, "Responsible research and innovation in information and communication technology: Identifying and engaging with the ethical implications of ICTs," *Responsible innovation: Managing the responsible emergence of science and innovation in society*, pp. 199–218, 2013.
[59] B. Abu-Shaqra and PhD, "Ethics of ethical hackers - Professional ethics," May 23, 2022. https://supervisorbullying.com/ethics-of-ethical-hackers/ (accessed Dec. 12, 2022).
[60] T. Georg, B. Oliver, and L. Gregory, "Issues of implied trust in ethical hacking," *The ORBIT Journal*, vol. 2, no. 1, pp. 1–19, 2018.
[61] T. Jaeger, R. Sailer, and U. Shankar, "PRIMA: policy-reduced integrity measurement architecture," presented at the Proceedings of the eleventh ACM symposium on Access control models and technologies, 2006, pp. 19–28.
[62] R. S. Sandhu, "Lattice-based access control models," *Computer*, vol. 26, no. 11, pp. 9–19, 1993.
[63] E.-C. Davri *et al.*, "Cyber Security Certification Programmes," presented at the 2021 IEEE International Conference on Cyber Security and Resilience (CSR), 2021, pp. 428–435.
[64] T. Georg, B. Oliver, and L. Gregory, "Issues of implied trust in ethical hacking," *The ORBIT Journal*, vol. 2, no. 1, pp. 1–19, 2018.
[65] G. A. Thomas, "On the offensive: is 'hacking back'ethical?," presented at the Higher Degree by Research Symposium, 2017.
[66] T. Georg, B. Oliver, and L. Gregory, "Issues of implied trust in ethical hacking," *The ORBIT Journal*, vol. 2, no. 1,

pp. 1–19, 2018.
[67] T. J. Bell III, “The social psychology of IT security auditing from the auditee’s vantage point: Avoiding cognitive dissonance,” *ISACA Journal*, vol. 3, no. 1, pp. 1–4, 2010.
[68] B. Oliver and L. Gregory, “Issues of implied trust in ethical hacking,” *The ORBIT Journal*, vol. 2, no. 1, pp. 1–19, 2018.
[69] L. G.Low, “Issues of implied trust in ethical hacking,” *The ORBIT Journal*, vol. 2, no. 1, pp. 1–19, 2018.
[70] S. Faily, J. McAlaney, and C. Iacob, “Ethical dilemmas and dimensions in penetration testing.,” presented at the HAISA, 2015, pp. 233–242.
[71] K. McCleary, R. Z. Goetzel, E. C. Roemer, J. Berko, K. Kent, and H. De La Torre, “Employer and employee opinions about workplace health promotion (wellness) programs: results of the 2015 Harris Poll Nielsen Survey,” *Journal of occupational and environmental medicine*, vol. 59, no. 3, pp. 256–263, 2017.
[72] M. T. Simpson, K. Backman, and J. Corley, *Hands-on ethical hacking and network defense*. Cengage Learning, 2010.
[73] A. Whitaker and D. P. Newman, *Penetration Testing and Network Defense: Penetration Testing _1*. Cisco Press, 2005.
[74] C. Payne, “On the security of open source software,” *Information systems journal*, vol. 12, no. 1, pp. 61–78, 2002.
[75] S. Fenz, J. Heurix, T. Neubauer, and F. Pechstein, “Current challenges in information security risk management,” *Information Management & Computer Security*, 2014.
[76] B. Abu-Shaqra, “Technoethics and Organizing: Exploring Ethical Hacking within a Canadian University,” 2015.
[77] J. Cannady, “Next generation intrusion detection: Autonomous reinforcement learning of network attacks,” presented at the Proceedings of the 23rd national information systems security conference, 2000, pp. 1–12.
[78] A. Y. Ding, G. L. De Jesus, and M. Janssen, “Ethical hacking for boosting IoT vulnerability management: A first look into bug bounty programs and responsible disclosure,” presented at the Proceedings of the Eighth International Conference on Telecommunications and Remote Sensing, 2019, pp. 49–55.
[79] B. D. Payne, M. Carbone, M. Sharif, and W. Lee, “Lares: An architecture for secure active monitoring using virtualization,” presented at the 2008 IEEE Symposium on Security and Privacy (sp 2008), 2008, pp. 233–247.
[80] V. B. Livshits and M. S. Lam, “Finding Security

Vulnerabilities in Java Applications with Static Analysis.," presented at the USENIX security symposium, 2005, vol. 14, pp. 18–18.
[81] J. Gadge and A. A. Patil, "Port scan detection," presented at the 2008 16th IEEE international conference on networks, 2008, pp. 1–6.
[82] G. Li *et al.*, "Exposing Vulnerable Paths: Enhance Static Analysis with Lightweight Symbolic Execution," presented at the 2021 28th Asia-Pacific Software Engineering Conference (APSEC), 2021, pp. 441–451.
[83] W. Zheng, J. Cheng, X. Wu, R. Sun, X. Wang, and X. Sun, "Domain knowledge-based security bug reports prediction," *Knowledge-Based Systems*, vol. 241, p. 108293, 2022.
[84] "CWE - 2022 CWE Top 25 Most Dangerous Software Weaknesses." https://cwe.mitre.org/top25/archive/2022/2022_cwe_top25.html (accessed Dec. 06, 2022).
[85] "CWE - 2022 CWE Top 25 Most Dangerous Software Weaknesses." https://cwe.mitre.org/top25/archive/2022/2022_cwe_top25.html#remapping_task (accessed Dec. 06, 2022).
[86] T. Riom, "A Software Vulnerabilities Odysseus: Analysis, Detection, and Mitigation," 2022.
[87] A. E. Michael *et al.*, "MSWasm: Soundly Enforcing Memory-Safe Execution of Unsafe Code," *arXiv preprint arXiv:2208.13583*, 2022.
[88] "CWE - CWE-787: Out-of-bounds Write (4.9)." https://cwe.mitre.org/data/definitions/787.html (accessed Dec. 06, 2022).
[89] "CWE - CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') (4.9)." https://cwe.mitre.org/data/definitions/79.html (accessed Dec. 06, 2022).
[90] "Vulnerability in HTML design: the script tag | Uploadcare Blog," *Uploadcare Inc.*, Dec. 30, 2020. https://uploadcare.com/blog/vulnerability-in-html-design/ (accessed Dec. 13, 2022).
[91] F. M. M. Mokbal, D. Wang, and X. Wang, "Detect Cross-Site Scripting Attacks Using Average Word Embedding and Support Vector Machine," *International Journal of Network Security*, vol. 4, no. 1, pp. 20–28, 2022.
[92] S. J. Weamie, "Cross-Site Scripting Attacks and Defensive Techniques: A Comprehensive Survey," *International Journal of Communications, Network and System Sciences*, vol. 15, no. 8, pp. 126–148, 2022.
[93] D. Korać, B. Damjanović, D. Simić, and K.-K. R. Choo, "A hybrid XSS attack (HYXSSA) based on fusion approach:

Challenges, threats and implications in cybersecurity," *Journal of King Saud University-Computer and Information Sciences*, 2022.
[94] B. Garn, J. Zivanovic, M. Leithner, and D. E. Simos, "Combinatorial methods for dynamic gray-box SQL injection testing," *Software Testing, Verification and Reliability*, vol. 32, no. 6, p. e1826, 2022.
[95] H. Maurel, "Deep learning applied on Web security," 2022.
[96] "CWE - CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') (4.9)." https://cwe.mitre.org/data/definitions/89.html (accessed Dec. 06, 2022).
[97] R. Firment, "Monitoring Support for Manta Flow Agent in Cloud-Based Architecture," 2022.
[98] S. Hiranwal, A. Sharma, and A. Saini, "SQL Injection Attack Prevention System".
[99] A. Jesudoss, T. M. Mercy, A. Christy, M. Maheswari, M. Selvi, and V. Ulagamuthalvi, "Analysis and implementation of SQL injection attack and countermeasures using SQL injection prevention techniques," *International Journal of Engineering Systems Modelling and Simulation*, vol. 13, no. 4, pp. 262–267, 2022.
[100] Z. Song, J. Ji, R. Zhang, and L. Cao, "Development of a test equipment for rating front to rear-end collisions based on C-NCAP-2018," *International journal of crashworthiness*, vol. 27, no. 2, pp. 522–532, 2022.
[101] A. Takanen, J. D. Demott, C. Miller, and A. Kettunen, *Fuzzing for software security testing and quality assurance*. Artech House, 2018.
[102] J. Wack, M. Tracy, and M. Souppaya, "Guideline on network security testing," *Nist special publication*, vol. 800, no. 42, pp. 13–14, 2003.
[103] Y. Zheng *et al.*, "D2A: a dataset built for AI-based vulnerability detection methods using differential analysis," presented at the 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), 2021, pp. 111–120.
[104] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," presented at the 2010 IEEE symposium on security and privacy, 2010, pp. 305–316.
[105] J. Pahdye and S. Floyd, "On inferring TCP behavior," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 287–298, 2001.
[106] P. Prabhakar *et al.*, "Cyber Security of Smart Metering Infrastructure Using Median Absolute Deviation

Methodology," *Security and Communication Networks*, vol. 2022, 2022.
[107] B. Chauhan, "💬1 - 3 Most Critical Apache Vulnerabilities Found - Astra Security Blog," Feb. 28, 2018. https://www.getastra.com/blog/911/top-3-most-critical-apache-vulnerabilities-found/ (accessed Dec. 09, 2022).
[108] R. Mazzolin and A. Madni, "IEEE Invited Paper-An Overview of Cyber Security Considerations and Vulnerabilities in Critical Infrastructure Systems and Potential Automated Mitigation-A Review".
[109] B. Farahani, F. Firouzi, and M. Luecking, "The convergence of IoT and distributed ledger technologies (DLT): Opportunities, challenges, and solutions," *Journal of Network and Computer Applications*, vol. 177, p. 102936, 2021.
[110] J.-P. A. Yaacoub, H. N. Noura, O. Salman, and A. Chehab, "Robotics cyber security: Vulnerabilities, attacks, countermeasures, and recommendations," *International Journal of Information Security*, vol. 21, no. 1, pp. 115–158, 2022.
[111] D. Dai and S. Boroomand, "A review of artificial intelligence to enhance the security of big data systems: state-of-art, methodologies, applications, and challenges," *Archives of Computational Methods in Engineering*, pp. 1–19, 2021.
[112] J. Yin, M. Tang, J. Cao, M. You, and H. Wang, "Cybersecurity Applications in Software: Data-Driven Software Vulnerability Assessment and Management," in *Emerging Trends in Cybersecurity Applications*, Springer, 2023, pp. 371–389.
[113] L. Wu, N. A. Dodoo, T. J. Wen, and L. Ke, "Understanding Twitter conversations about artificial intelligence in advertising based on natural language processing," *International Journal of Advertising*, vol. 41, no. 4, pp. 685–702, 2022.
[114] J. E. T. Akinsola *et al.*, "Application of Artificial Intelligence in user interfaces design for cyber security threat modeling," in *Software Usability*, IntechOpen, 2021.
[115] P. Madduru, "Artificial intelligence as a service in distributed multi access edge computing on 5g extracting data using IOT and including AR/VR for real-time reporting," *INFORMATION TECHNOLOGY IN INDUSTRY*, vol. 9, no. 1, pp. 912–931, 2021.
[116] L. Kong *et al.*, "Edge-Computing-Driven Internet of Things: A Survey," *ACM Computing Surveys (CSUR)*, 2022.