Ethical hacking: real evaluation model of brute force attacks in password cracking

Buthayna Al Sharaa, Saed Thuneibat
Department of Electrical Engineering, Al-Balqa Applied University, As-Salt, Jordan

Article Info

Article history:

Received Nov 4, 2023
Revised Nov 18, 2023
Accepted Dec 25, 2023

Keywords:

Brute force computation
Brute force modelling
Ethical hacking
Kali linux
Markov chains UML state diagram
Network simulator

ABSTRACT

Despite ongoing efforts to convince users of the value of password security and to enforce password creation standards on them, in many information systems the human factor still plays a role. In addition, not only do most users' password creation and management practices largely remain unchanged, but password cracking tools and more critically, computer hardware also continue to advance. In this paper we present a model in ethical hacking; the proposed model concentrated on brute force attacks for password cracking. The main novelty of our work is that it first presents a mathematical model that calculates the number of different password permutations of varying lengths. Then the brute force attack is modelled using the Markov chain model and a method is developed to formulate the conventional optimization problem, which is classified as a discrete nonlinear problem. The experiments' results demonstrate and validate the method's effectiveness and suitability.

Corresponding Author:

Buthayna Alsharaa
Department of Electrical Engineering, Al-Balqa Applied University
As-Salt, Jordan
Email: buthayna-alsharaa@bau.edu.jo

1. INTRODUCTION

Weaknesses in a system that an attacker could use to cause dangerous effects are known as vulnerabilities. When a system is vulnerable, a threat can appear in the form of a threat agent that uses a particular penetration technique to achieve unwanted outcomes [1]–[3]. There exists a genuine risk of significant financial harm to businesses. A total of 74.3% of losses are attributed to infections, unauthorised access, theft of laptop or mobile hardware, and theft of personal information, according to the 11th annual computer crime and security survey [2]. In fact, a study by McCue [1] found that 90% of security measures are directed at preventing external threats, even though 70% of fraud is committed by internal rather than external criminals. Employing ethical hackers can help businesses identify security weaknesses in their systems before the hackers do. The work of ethical hackers, who are licensed professionals who work in teams, is crucial in detecting system hazards and highlighting their shortcomings [4], [5]. Their objective is to strengthen their system and make it less vulnerable to threats, which will increase system security [6], [7].

Information security may be viewed as a type of computer audit according to article [8]. Regarding article [9], ethical hacking may be one of the best approaches to proactively fix the Internet's numerous security flaws. As they both aim to find problems, systems and hacking abilities might be compared to auditing abilities. Network analysts, penetration testers, and any other type of cybersecurity and analytics activity can use Kali Linux as a safe environment for ethical hacking. Penetration testing and advanced security audits are Kali Linux's main uses. The Kali software contains several hundred tools that are intended for a range of information security tasks, such as penetration testing, computer forensics, and reverse engineering [10]–[15]. A brute force attack is a hacking method that uses trial and error to crack passwords, login credentials, and encryption keys. It is a simple yet efficient method for gaining unwanted access to networks, enterprise systems, and user accounts. Hackers experiment with different usernames and passwords until they find the right login information. They usually use a computer to test a large number of combinations. [14]. Most scientific research papers, in recent times, contain a survey, comparison or review study of ethical hacking [14]–[16]. This paper examines the brute force cyberattacks in real manner. The six hacking steps for brute force attack are illustrated. The information acquired from each phase as well as the tools used to hack the phases are listed. The paper also includes the algorithm used to conduct real-time brute force attacks on an experimental login page.

In general, three major groups of current password strength estimation techniques can be distinguished, each of which takes a different approach to the same issue: attack-based, a heuristic-based, and probabilistic-based methods. The attack-based techniques estimate a password's resistance based on how long it takes a particular attack (or group of attempts) to crack it [17], [18]. The strength of the password increases with the amount of time it takes the assault to crack it. On the hand, the heuristic-based approaches concentrate on offering a heuristic-based assessment of password difficulty [19].

The limitations of the earlier approaches are addressed by probabilistic-based solutions. They employ techniques for calculating password strength based on statistical analyses of passwords [20], [21]. Many of these techniques are built on Markov models [19]. Markov models have shown to be quite helpful for password security as well as computer security in general. They may be used to precisely evaluate the strength of new passwords [22] and are a powerful tool for password cracking [23]. According to the research described in [24], which examined various probabilistic password models, Markov models are more effective at calculating password probabilities than probabilistic context-free grammars. In this work we present the mathematical password model. The presented model shows the most important factors that determine the strength of passwords and the practical experiments that have been conducted prove the validity of the presented model. Markov chain-based model has also been used to model probabilities in the brute force attacks. The conclusions of these studies showed that Markov processes can be used to model information systems.

2. METHOD

This section is devoted to explaining research methodology. The six phases of Ethical hacking are first illustrated. Then, a mathematical model for the stochastic brute force process is made. Hacking has six phases which are: information gathering, scanning, gaining access, maintaining access, installing backdoors, and covering tracks [25]–[28]. Figure 1 is a UML state diagram that summarizes the hacking phases in general and tools used for each phase. The hacking process starts at the first state which is information gathering. Brute force attack goes through these states. State 1 and 2 are used to collect information about the victim to obtain the username and any information which may facilitate password guessing.

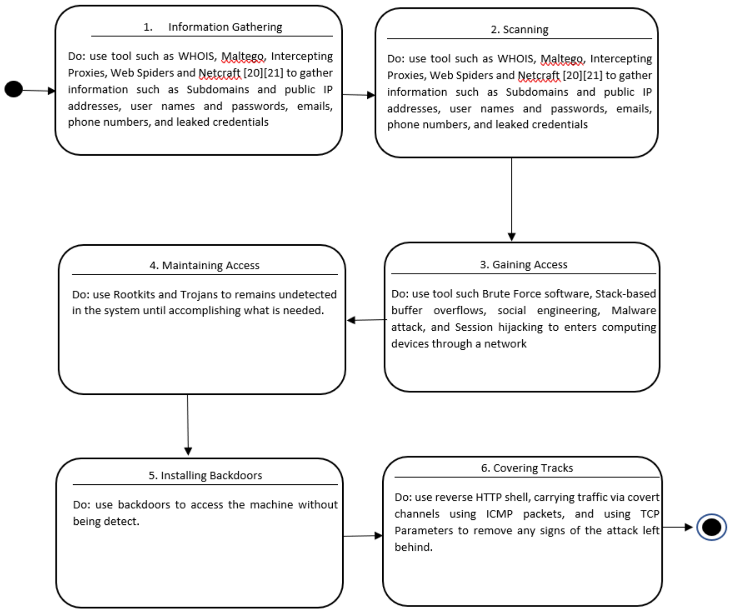
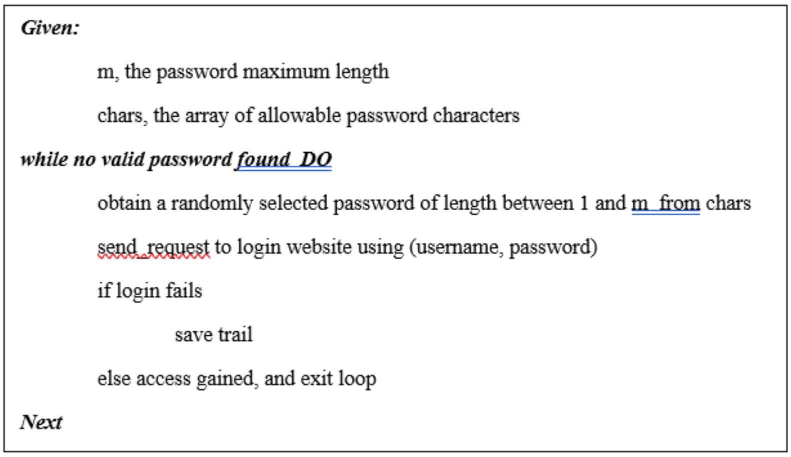

1. Information Gathering
Do: use tool such as WHOIS, Maltego, Intercepting Proxies, Web Spiders and Netcraft [20][21] to gather information such as Subdomains and public IP addresses, user names and passwords, emails, phone numbers, and leaked credentials
2. Scanning
Do: use tool such as WHOIS, Maltego, Intercepting Proxies, Web Spiders and Netcraft [20][21] to gather information such as Subdomains and public IP addresses, user names and passwords, emails, phone numbers, and leaked credentials
3. Gaining Access
Do: use tool such Brute Force software, Stack-based buffer overflows, social engineering, Malware attack, and Session hijacking to enters computing devices through a network
4. Maintaining Access
Do: use Rootkits and Trojans to remains undetected in the system until accomplishing what is needed.
5. Installing Backdoors
Do: use backdoors to access the machine without being detect.
6. Covering Tracks
Do: use reverse HTTP shell, carrying traffic via covert channels using ICMP packets, and using TCP Parameters to remove any signs of the attack left behind.

Figure 1. The six phases of ethical hacking

Brute force attack starts at the gain access state. Brute force is a method of breaking into any websites or systems by using a program to try out different combinations of passwords. The user password combinations are repeatedly generated automatically by software until the correct combination is produced. Once access is gained the hacker has full access to all resources granted to the hacked user account. Figure 2 shows the proposed pseudo-code for the brute force attack algorithm [29]. From an analytical perspective, a brute force attack is a stochastic process that a Markov chain can explain. Therefore, the study's goals are to:

- Formally analyse the Markov chain paradigm that characterises brute force attacks.
- Perform a definition for system optimisation based on the mathematical paradigm provided.
- Apply the suggested method for operational model optimisation to the test for a real information system, then analyse the outcomes.



```
Given:
        m, the password maximum length
        chars, the array of allowable password characters
while no valid password found  DO
        obtain a randomly selected password of length between 1 and m  from chars
        send_request to login website using (username, password)
        if login fails
                save trail
        else access gained, and exit loop
Next
```

Figure 2. Pseudo-code for bruteforceattack algorithm

2.1. Mathematical model of the studied process

A brute force attack uses trial-and-error to determine login credentials. Hackers try every combination in the hopes of making an accurate approximation. Usually, an array that includes a list of allowed characters in a password is made. This array includes uppercase and lowercase letters, special characters, and digits. The total number of characters in this list is an integer number m. This list is used to create password permutations of lengths varying from 1 to n characters. The (1) calculates the number of different password permutations pore-water pressure (PWP) varying from 1 to n:

$$\text{PWP}= \sum_{i=1}^{n} {}^{m!}/_{(m-i)!} \tag{1}$$

In the equation above, i is the length of the password. The case when i is not known the attacker must generate password permutations with lengths ranging from 1 to a number n. Analytically, the brute force process can be described by a Markov chain. The sequence of trials produced belong to one of two states: accepted and unaccepted states. The output of any trial depends only on the initial state and not on any previous states. Figure 3 is the UML state diagram of the stochastic brute force process illustrated in Figure 2.

In this model p and q are called the transition probabilities. p is the probability that a valid password is created. q is the probability that an invalid password is created. Based on (1), the equations for p and q are given in (2) and (3) respectively. The (4) is a matrix of probabilities of transitions between states represented in Figure 3:

$$p=1/\sum_{i=1}^{n} {}^{m!}/_{(m-i)!} \tag{2}$$

$$q= 1\text{-}p \tag{3}$$

$$\text{W}= \begin{pmatrix} 0 & q \\ p & 0 \end{pmatrix} \tag{4}$$

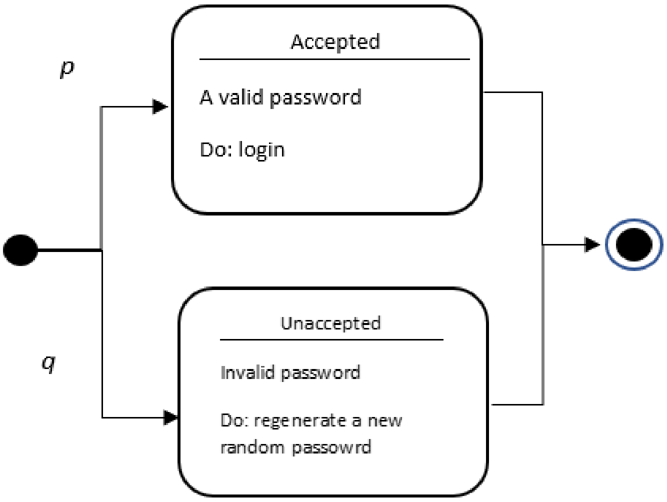


Figure 3. UML state diagram of the stochastic brute force process

3. RESULTS AND DISCUSSION

This section presents real-time results obtained from the attack on a login web page designed to carry out practical experiments for this research paper. A program written in python is used to generate random passwords of lengths varying from 1 to 5 characters. Each generated password is used to login into the webpage.

3.1. Simulation results

Table 1 shows the time taken to figure out the correct password and login into the webpage. To run the brute force program faster, multithreading is used. Ten instances of the program are run at the same time. The number of threads chosen depends on the machine used.

Table 1. Time taken to figure out correct password

Number of characters	Numbers only	Lower case letters	Lower and upper case letters	Numbers, lower, and upper case letters	Numbers, lower, upper case letters and symbols
2	1.20406 sec	3.01219 sec	3.0014 min	1.001 min	17.3201 min
3	3.23332 sec	6.31009 sec	20.4107 min	43.67min	1.01 hour
4	4.61152 sec	3.01219 min	27.9104 min	1.6 hour	2.3 hour
5	7.62713 sec	15.8911 min	1 hour	4 hours	7.8 hour

The (1) shows the factors that affect the time required to obtain passwords and hack to a particular site. Figure 4 shows the relationship between the length of a password and the number of different permutations that result from that length for different character list lengths (m). Figure 5 shows how the number of password permutations increase by increasing the size of the list of allowed characters in a password m for different password lengths.

3.2. Discussion

Let's start the discussion with the statement shown in Table 1. Various factors affect the time values shown in the Table 1. One of these factors is the network properties. Also, the process of generating passwords is random and not sequential. So, the correct password can be obtained in the early stages without the need to find them all.

Next, we analyze the results in Figure 4 which explain when the different number of permutations increases as the password lengths increase. In case the attacker does not know the length of user password, he must try all possible alternatives ranging from one to n. It is expected that as the user choses longer passwords, it will become harder to figure out the correct password. The result here is affected by the length of the list holding the allowed symbols to be used in the password m. The results in Figure 5 uses the value 15 for m.

Figure 5 show the effect of m on the number of permutations. Here m is determined by the organization administrators. Organizations usually set security policies to choose passwords for their users. Some of these policies are:

- Passwords must contain at least 12 characters.
- Uppercase and lowercase letters, special characters, and digits should all be used in passwords.
- Passwords should be changed every 60 to 90 days.
- Reusing passwords should be prohibited.

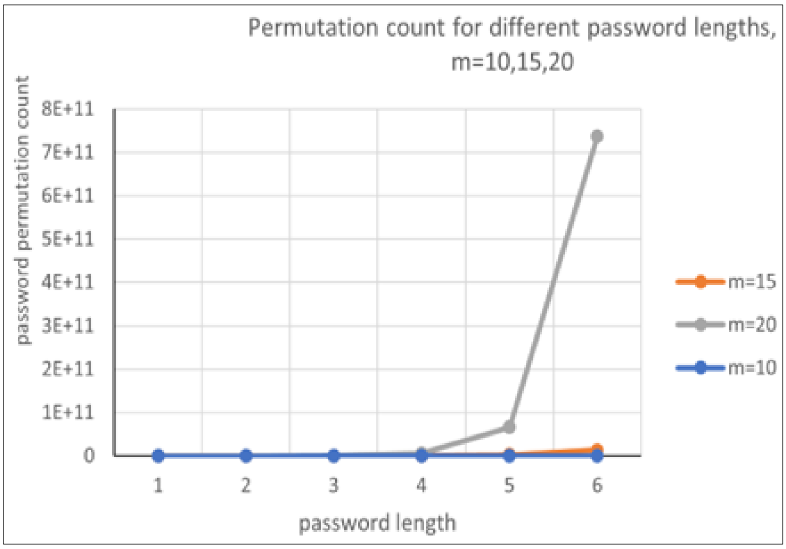


Figure 4. Permutation counts for different password lengths, m=10, 15, 20

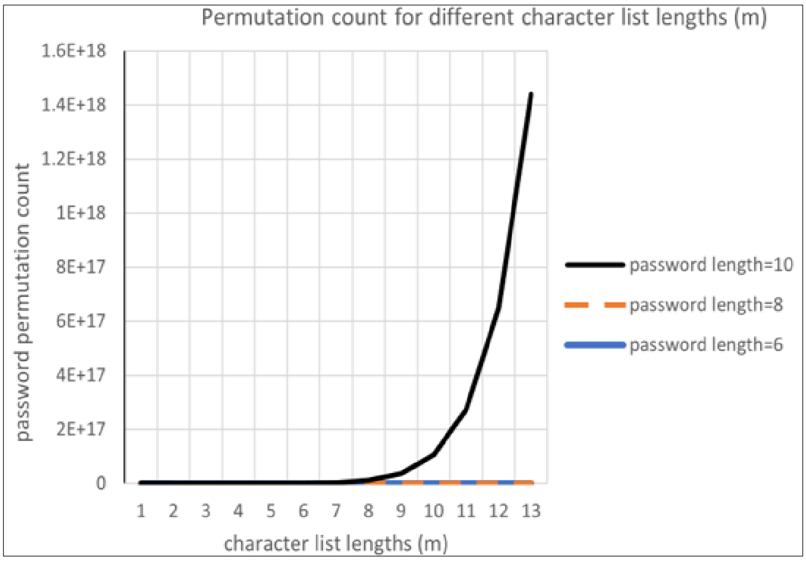


Figure 5. Permutation counts for different character list lengths (m), and fixed password length=6, 8, 10

4. CONCLUSION

The entire world is heading toward technological advancement, and as real-world activities get more and more digitized, the risk of security increases. In order to get beyond security measures and reach the intended data target, brute force assaults are employed. Although it might appear that this is just useful for hackers, several security firms use brute force attacks to assess their clients' systems. An automated assault, whether or not it is online, is dangerous anytime it targets a system because it won't take long for it to succeed. The article gives a Markov chain-based mathematical model of the brute force attack. The study proposed a mathematical programable model for modelling this type of attack because there isn't a mechanism that can be used universally for doing so. Experiment findings demonstrated the method's suitability and effectiveness. From the results it appears that the primary difference between strong and weak passwords is length. The use of graphical passwords or biometry, and multifactor authentication could be the solution.

REFERENCES

[1] M. Jouini, L. B. A. Rabai, and A. B. Aissa, "Classification of security threats in information systems," *Procedia Computer Science*, vol. 32, pp. 489–496, 2014, doi: 10.1016/j.procs.2014.05.452.

[2] M. Alhabeeb, A. Almuhaideb, P. D. Le, and B. Srinivasan, "Information security threats classification pyramid," in *2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops*, IEEE, 2010, pp. 208–213. doi: 10.1109/WAINA.2010.39.

[3] S. Geric and H. Zejko, "Information system security threats classifications," *Journal of information and organizational sciences*, vol. 31, no. 1, pp. 51–61, 2007.

[4] B. AlSharaa, S. Thuneibat, R. Masadeh, and M. Alqaisi, "Selected advanced themes in ethical hacking and penetration testing," *Computer Science and Information Technologies*, vol. 4, no. 1, pp. 69–75, Mar. 2023, doi: 10.11591/csit.v4i1.p69-75.

[5] X. Zhang, A. Tsang, W. T. Yue, and M. Chau, "The classification of hackers by knowledge exchange behaviors," *Information Systems Frontiers*, vol. 17, no. 6, pp. 1239–1251, Dec. 2015, doi: 10.1007/s10796-015-9567-0.

[6] R. Banda, J. Phiri, M. Nyirenda, and M. M. Kabemba, "Technological paradox of hackers begetting hackers: a case of ethical and unethical hackers and their subtle tools," *Zambia ICT Journal*, vol. 3, no. 1, pp. 40–51, Mar. 2019, doi: 10.33260/zictjournal.v3i1.74.

[7] B. Sahare, A. Naik, and S. Khandey, "Study of ethical hacking," *International Journal of Computer Science Trends and Technology*, vol. 2, no. 6, pp. 6–10, 2014.

[8] S. Patil, A. Jangra, M. Bhale, A. Raina, and P. Kulkarni, "Ethical hacking: the need for cyber security," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, IEEE, Sep. 2017, pp. 1602–1606. doi: 10.1109/ICPCSI.2017.8391982.

[9] R. Hartley, D. Medlin, and Z. Houlik, "Ethical hacking: educating future cybersecurity professionals," in *Proceedings of the EDSIG Conference*, 2017, pp. 1–10.

[10] S. Sinha, *Beginning Ethical Hacking with Kali Linux*. Berkeley, CA: Apress, 2018.

[11] P. Cisar and R. Pinter, "Some ethical hacking possibilities in Kali Linux environment," *Journal of Applied Technical and Educational Sciences (jATES)*, vol. 9, pp. 129–149, 2019.

[12] M. Bishop, "About penetration testing," *IEEE Security & Privacy Magazine*, vol. 5, no. 6, pp. 84–87, Nov. 2007, doi: 10.1109/MSP.2007.159.

[13] D. Bhatt, "Modern day penetration testing distribution open source platform-Kali Linux-study paper," *International Journal of Scientific and Technology Research*, vol. 7, no. 4, pp. 233–237, 2018.

[14] V. Grover, "An efficient brute force attack handling techniques for server virtualization," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3564447.

[15] O. Valea and C. Oprisa, "Towards pentesting automation using the metasploit framework," in *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, IEEE, Sep. 2020, pp. 171–178. doi: 10.1109/ICCP51029.2020.9266234.

[16] International Council of E-Commerce Consultants, "Ethical hacking and countermeasures," 2010, Accessed: Dec. 20, 2022. [Online]. Available: https://pdfroom.com/books/ethical-hacking-and-countermeasures-attack-phases/3kZdoDb1gM8.

[17] P. G. Kelley *et al.*, "Guess again (and again and again): measuring password strength by simulating password-cracking algorithms," in *2012 IEEE Symposium on Security and Privacy*, IEEE, May 2012, pp. 523–537. doi: 10.1109/SP.2012.38.

[18] J. Bonneau, "The science of guessing: analyzing an anonymized corpus of 70 million passwords," in *2012 IEEE Symposium on Security and Privacy*, IEEE, May 2012, pp. 538–552. doi: 10.1109/SP.2012.49.

[19] R. Shay *et al.*, "Encountering stronger password requirements: user attitudes and behaviors," *ACM International Conference Proceeding Series*, 2010, doi: 10.1145/1837110.1837113.

[20] J. Bonneau, "Statistical metrics for individual password strength," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7622 LNCS, pp. 76–86, 2012, doi: 10.1007/978-3-642-35694-0_10.

[21] J. Ma, W. Yang, M. Luo, and N. Li, "A study of probabilistic password models," in *2014 IEEE Symposium on Security and Privacy*, IEEE, May 2014, pp. 689–704. doi: 10.1109/SP.2014.50.

[22] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989, doi: 10.1109/5.18626.

[23] A. Narayanan and V. Shmatikov, "Fast dictionary attacks on passwords using time-space tradeoff," in *Proceedings of the 12th ACM conference on Computer and communications security*, New York, NY, USA: ACM, Nov. 2005, pp. 364–372. doi: 10.1145/1102120.1102168.

[24] C. Castelluccia, M. Dürmuth, and D. Perito, "Adaptive password-strength meters from markov models," *Ndss-Symposium.Org*, vol. 1, no. 1, pp. 1–14, 2021.

[25] EC-Council, "Ethical hacking and countermeasures: attack phases, volume 1," in *Cengage Learning*, 2nd ed., vol. 1, 2009, p. 352.

[26] M. C. Ghanem and T. M. Chen, "Reinforcement learning for efficient network penetration testing," *Information*, vol. 11, no. 1, p. 6, Dec. 2019, doi: 10.3390/info11010006.

[27] M. I. Tayag and M. E. A. D. V. Capuno, "Compromising systems: implementing hacking phases," *SSRN Electronic Journal*, 2019, doi: 10.2139/ssrn.3391093.

[28] A. Gupta and A. Anand, "Ethical hacking and hacking attacks," *International Journal Of Engineering And Computer Science (IJEECS)*, Apr. 2017, doi: 10.18535/ijecs/v6i4.42.

[29] D. Chudasama and R. Bhavsar, "Technical methods of information gathering," *Journal of Web Engineering and Technology*, vol. 8, no. 3, pp. 1–5, 2022.

BIOGRAPHIES OF AUTHORS

Buthayna Al Sharaa Received her master's degree in computer engineering from Jordan university for science and technology, Jordan in 2006. Now she is a lecturer at Al-Balqa Applied University-Al-huson University College, Jordan. Her research interests include data and network security, Artificial Intelligence, and algorithms, Cloud computing, Routing protocols, Embedded Systems. She can be contacted at email: buthayna-alsharaa@bau.edu.jo.

Dr. Saed Thuneibat received his B.Sc. in Automatic telecommunication Engineering from Novosibirsk State University in 1994. He received his M.Sc. and Ph.D. in telecommunication/networks engineering from the same university, Russia 2005. Currently, he is an associate professor at the Department of Electrical Engineering at Al-Balqa` Applied University, Jordan. His research interests are fiber optics, digital communication systems, and networking. He can be contacted at email: saed1970@bau.edu.jo.