The ISC Int'l Journal of Information Security

ISeCure

November 2021, Volume 13, Number 3 (pp. 11–19)

http://www.isecure-journal.org

Selected Paper at the ICCMIT'21 in Athens, Greece

Broken Authentication and Session Management Vulnerabilities **

Hanan Aljoaey [1], Khawla Almutawa [1,*], Ruyuf Alabdali [1], and Dina M. Ibrahim [1,2]

[1] *Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia.*

[2] *Computers and Control Engineering Deptartment, Faculty of Engineering, Tanta University, Tanta, Egypt.*

ARTICLE INFO.

Keywords:
Broken Authentication, Session Management, Credential Stuffing, Password Spraying

Type: Research Article

doi: 10.22042/ISECURE.2021.0.0.0

dor: 20.1001.1.20082045.2021.13.3.2.9

ABSTRACT

Web application protection is today's most important battleground between the victim, intruder, and web service resource. User authentication tends to be critical when a legitimate user of the web application abruptly ends the contact while the session is still active, and an unauthorized user chooses the same session to gain access to the device. For many corporations, risk detection is still a problem. In other cases, it is a usual way of operating that provides the requisite protection to keep the product free of weaknesses. Using various types of software to identify different security vulnerabilities assists both developers and organizations in securely launching applications, saving time and money. Different combinations of tools have been seen to enhance protection in recent years, but it has not been possible to combine the types of tools available on the market until the writing of this report. This paper aims to clarify vulnerabilities in broken authentication and session management. It is worth noting that if the creator practices the preventive techniques outlined in this article, the chances of exploitation being discussed are reduced. This paper revealed that the most powerful ways to exploit the Broken Authentication and Session Management vulnerabilities of the web application in those domains are Session Misconfiguration assault and Cracking/ Guessing Weak Passwords. Correspondingly included techniques to defend authentication and the most important is using a robust encryption system, setting password rules, and securing the session ID.

1 Introduction

Broken authentication attacks have been responsible for many of the most serious data breaches in recent history, prompting security experts to express concern about this underappreciated threat. Broken authentication is a catch-all term that refers to a variety of security flaws that hackers might exploit

* Corresponding author.

**The ICCMIT'21 program committee effort is highly acknowledged for reviewing this paper.

Email addresses: 421200385@qu.edu.sa, 421216228@qu.edu.sa, 421216233@qu.edu.sa, d.hussein@qu.edu.sa, dina.mahmoud@f-eng.tanta.edu.eg

ISSN: 2008-2045

to impersonate legitimate online users. Broken authentication is a term that relates to flaws in two elements of authentication: session management and the management of credentials. Both are classified as broken authentication because they allow attackers to impersonate a user by using either hijacked session IDs or stolen login credentials to accomplish their goals. Attackers have also discovered that using someone else's credentials to get access to restricted networks is the most effective method of gaining access. Following the release of Verizon's 2020 Data Breach Report [1], phishing and the use of stolen credentials have emerged as the most common means of conducting a data breach. Broken authentication is not detected by automatic scanners, and thus necessitates the painstaking manual work of understanding how the authentication schema ensures the user's identity, session, and authentication when using automatic scanners. Once a decision has been made, manual tests are carried out to see whether any authentication attacks are possible.

Attacks of this type aim to overtake accounts that offer the attacker the same rights as the user. In addition, when attackers can compromise passwords, keys or session tokens, user account information, and other details to impersonate another user's identity. The authentication is broken when hackers try to discover the other person's password, key, and other details that can affect his life by stealing his identity, work, and safety.

The number one factor behind successful broken authentication attacks is bad web application design [2]. Weaknesses in session management can only be understood by understanding how online authentication and browsing typically work. For each visit, the web application issues a session ID to the user. To allow the application to communicate with the user and respond to requests, this ID is necessary. Protocol designers may use the proposed principles to come up with schemes that are both user-friendly and safe. Several broken instances of authentication attack are:

(1) Session Hijacking: Checked session IDs can hijack user identities. Suppose a user forgets to log off from a public device. In that case, the same session ID that was previously generated for the original user can be used by some other person to continue the session.
(2) Session ID URL: The Session ID appears in the website URL and can be used by any person who accesses the URL through a wired or wireless network to impersonate the identity of the user.
(3) Credential Stuffing: Hackers access a database containing unencrypted user passwords and use techniques to assess if the passwords are correct and usable and protocols that protect against such attempts must be provided for a stable web application.
(4) Password Spraying: Password spraying refers to the use by hackers attempting to access protected accounts of the most common and poor passwords, such as 'password' or '123456.' Consequently, to prevent such attacks, minimum password specifications have been added.
(5) Phishing Attacks: To get users to reveal their login credentials, hackers give users links to a website that resembles the original web application. However, with due caution and by checking the web application in use, phishing attacks can be easily avoided [3].

2 Related Work on Broken Authentication

The rest of this paper is arranged as follows: In Section 1, introduction and background on broken authentication are given; Section 2 describes the literature review; Section 3 explains methods for detecting broken authentication; Section 4 shows how to prevent it, and finally, Section 5 gives the conclusion.

The system of Mishra *et al.* [4], the system of Wu *et al.* [5], and the system of Moon *et al.* [6] were all tested, and it was discovered that, although all of them were equipped with structured security evidence, the system of Mishra *et al.* fails to maintain user privacy and is vulnerable to a new type of insider attack, as illustrated in [7]. The system developed by Wu *et al.*, on the other hand, is incapable of avoiding a de-synchronization attack. The cryptanalysis of these three systems, together with their previous experience creating and analyzing protocols, has led them to discover two concepts for designing more secure user authentication systems based on smart cards for the future. For protocol designers, the stated principles will be useful in developing schemes that provide desirable user-friendliness and protection. It is utilized by the system of Mishra *et al.* [4], the scheme of Wu *et al.* [5], and the scheme of Moon *et al.* [6] to authenticate users via implicit key authentication. There is no way for the servers in any of these three systems to know whether the session key has been completely obtained by the user, which indicates that this implicit key authentication is devoid of key validation. In the absence of an explicit means to verify the validity of user-keyed identification and password, subsequent systems are subject to smart card failure attacks, i.e. when there is no/insecure smart card verification or when the protocol is used in conjunction with interactive verification. As described in [8], the authors provide a technique for identifying authentication and session management

security vulnerabilities that have been compromised. The algorithm suggested in this paper performs a scanning procedure for files to be used in blogs and mobile applications. The purpose of this study is to uncover the constraints of authentication and session management that have gone wrong. The suggested method would aid companies and developers in identifying and resolving vulnerabilities while also improving overall protection. Their scanner tool is reliant on learning the source code of the application, which is dependent on ASP.NET files and the code behind files (C sharp and C#). They sought to test their algorithm on a few other online platforms. They are, however, unable to obtain a functioning website. As a result, they must do offline verification of the information. The IIS server was used to host a web application that was developed specifically for this purpose. In a similar vein, the authors of [9] emphasize the importance of security and how it is necessary. Then they get into authentication and how difficult it is for users to do certain tasks. In their paper, they discuss two approaches to password authentication: 1- Mishra *et al.* and 2- Wu *et al.* Both plans are filed to safeguard users from being targeted by attackers. After that, they offer two fundamental ideas for the future. One principle is:

(1) Better password change principle. It suggests a new standard set of ten qualities and nine security requirements.
(2) candid key authentication is meant to match key authentication policy. These principles help with authentication security.

The work presented in [10] proposed an improved detecting model that can detect two attacks that is cross-site request forgery attack and broken authentication and session management attack. They also developed a packet tracker module to trace the request for any malicious script that the attacker can craft. While in [11], they focus on broken authentication and session management vulnerability. Also, they show techniques to enhance the security of web services and they proposed a model architecture for broken authentication and Session Management that is shown in Figure 1.

Broken authentication is exploited by the authors in [12] to recover the password by asking a secret question, which is usually answered by the city's name at the time of its creation, and the site displays the password immediately without further verification, thereby circumventing the authentication process. A user's country can be determined using social engineering, and then the attacker can search for him using the dictionary method, and then get login credentials and session identifiers with the use of brute force tools.

The document [13] presents an evaluation and review of the vulnerability of Broken Authentication and Session Management, as well as the five types of exploitation that can occur as a result of it. Web vulnerabilities such as broken authentication are caused by misconfigured session management and are classified as critical. The completion of the authentication procedure will result in the establishment of a session, which will be engaged for data transmission between the server and a particular user. Figure 2 depicts the Broken Authentication issue by exploiting the issue of session mismanagement to illustrate the problem.

The issue of failed authentication and session management is one of the most significant roadblocks to ensuring the secrecy of a web-based application's data. It is important to note that if the developer follows the prevention measures suggested in this paper, the likelihood of abuse being addressed would be reduced significantly.

As demonstrated in the work provided in [14], training lightweight security protocols in resources associated with the Internet of Things applications is a difficult undertaking. The signature build-up over-voltage over-scaling is critical to the hardware lightweight authentication protocol's success (VOS). According to the researchers, machine learning (ML) is reliant on modeling attacks to compromise authentication. This group has gained hands-on expertise with voltage overscaling-based lightweight authentication (VOLtA) and how it interacts with machine learning. They get to the conclusion that (VOLtA) is not protected by ML. It is discussed in [15] by Daniel Huluka and Oliver Popov, who address numerous methods of securing online applications, which is considered one method among many for keeping the Internet safe from regular cyberattacks. Attempt each attempt, the attacker comes up with something new. As part of Root Cause Analysis, researchers attempt to get to the root of the problem (RCA). They did investigate the association between RCA and online applications that had significant issues. This results in a relationship between application security and vulnerability.

In [16], the authors cover the OWASP technique of risk assessment, as well as the probability factors that are derived from it. As depicted in Figure 3, they summarized the information by using the associated probabilities. In addition, we can see that the issue of faulty authentication came in second place for the year 2017.

The most recent vulnerability survey is included in [17], as well as the determination of vulnerabilities, the methodology used in the determination, and the

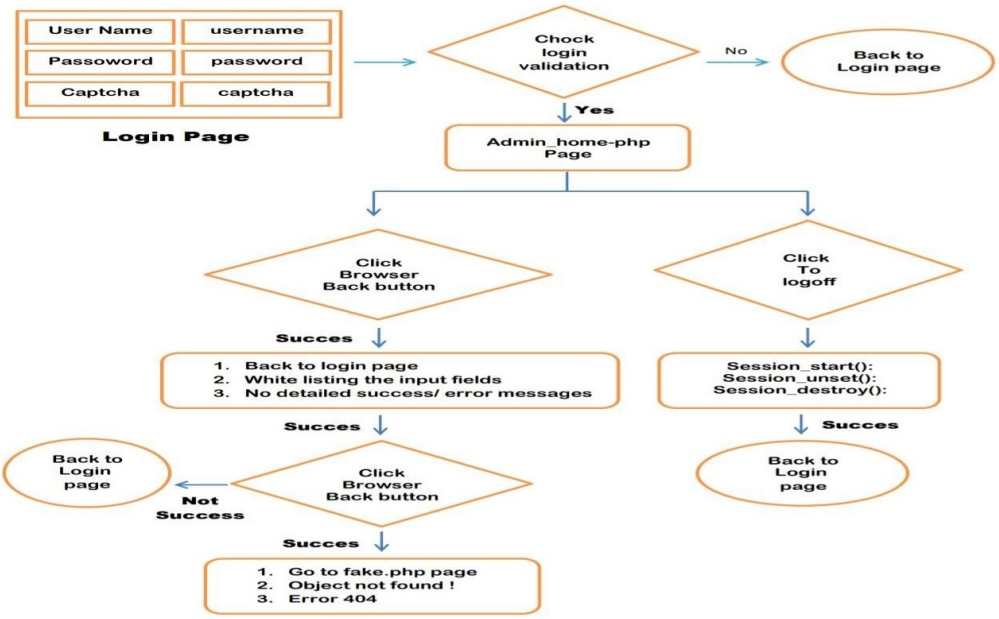

Figure 1. Steps of broken authentication and session management

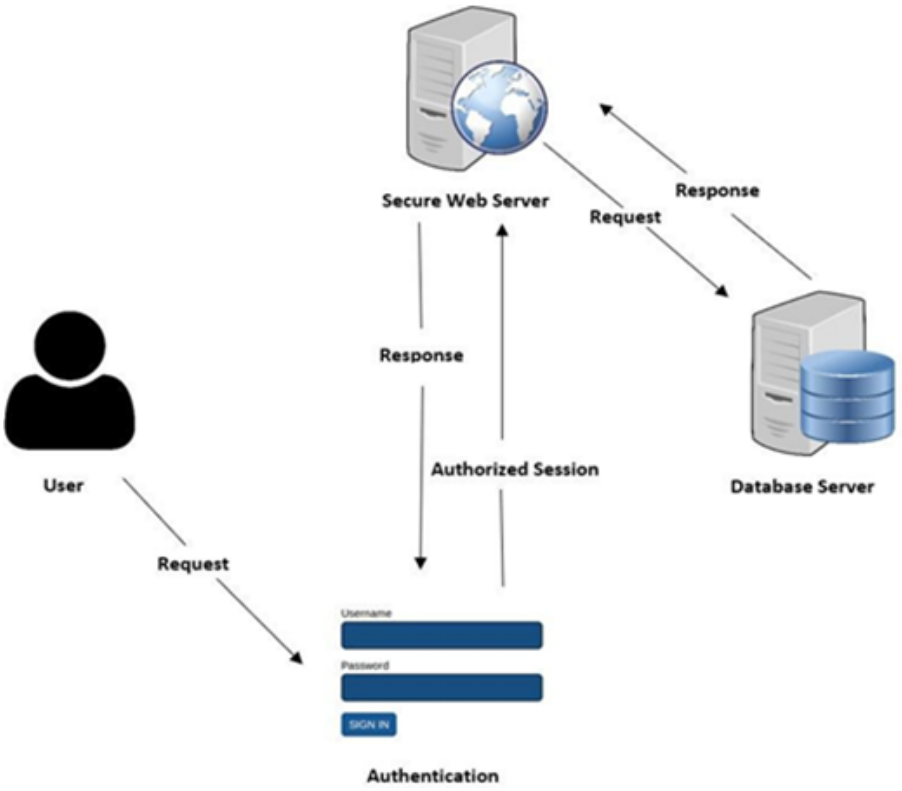

Figure 2. Authentication and session management process

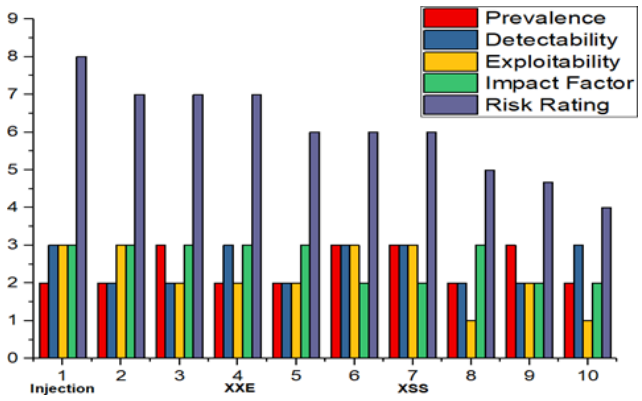

Figure 3. Decomposition analysis of the OWASP Top 10 (Horizontal Axis: 1. Injection Attack, 2. Broken Authentication, 3: Sensitive Data Exposure, 4. XML External Entities (XXE), 5. Broken Access Control, 6. Security Misconfiguration, 7. Cross-Site Script

procedures used to assess the vulnerabilities to defend enterprises from cyber threats. Regularly, a great number of bugs are detected and fixed. According to a survey, there are at least 30-40 vulnerabilities that are exploited regularly. In web application security, the control of authentication and sessions is a critical section to consider. Organizations must identify and mitigate the greatest number of vulnerabilities and dangers to remain secure in their operations. Vulnerability Assessment and Penetration Testing must be performed on a daily basis in accordance with VAPT. VAPT results must be prioritized and explained with CVE numbers that can be derived from industry-standard references such as the National Vulnerability Database (NVD), the Common Vulnerability Scoring System (CVSS), and the Open Source Vulnerability Database (OSVDB), CVE information, and so on.

The writers of the paper [18] concentrate on the use of password generators for authentication to protect users. The ancient approaches did not work out well. They propose a new authentication mechanism to prevent hackers from launching another attack. Therefore, they have switched from phone to email authentication as a result of the possibility of phone loss or damage. Kyung-Ah Shim (Kyung-Ah Shim) (2017). As discussed in this article [19], it is necessary to validate each message from an automobile business sent over WAVE protocol. As a result, no one receives malicious messages or causes traffic systems to malfunction. The outcome of their investigation, security, and authentication protocol was a complete failure. All the previous studies have been summarized in Table 1.

3 Broken Authentication & Session Management Exploitation

Different forms of manipulation methods exist for Broken Authentication and Session Management. In the public and private sectors, the manual penetration testing approach [20] has been used to verify the vulnerability of web applications. The general Broken Authentication & Session Management exploitation technique is depicted in Figure 4.

(1) Attack on Session Misconfiguration
One of the most important factors in maintaining a stable authentication process for web applications is session length. When a device validates a user's credential, it generates a session for that user with a valid session ID for a fixed amount of time. If the user does not log off their account as instructed by the application's designer, the session will remain active for a set period.

(2) Exploiting Weak Passwords by Cracking/ Guessing
Some non-technical users retain their passwords in a generic form due to a lack of knowledge about password management. The attacker sends a user login connection to Hydra, which searches a predefined dataset for username and password.

(3) Exploiting the Authentication Problem
Conditional queries are used to verify usernames and passwords in web application authentication schemes. If these conditional requests become corrupted or are not properly managed, an attacker may easily gain access to the device.

(4) Decoding Inadequate Encryption
By exploiting the security vulnerabilities of exposing the session ID in the system's URL, for example, an intruder may steal the session ID against one consumer.

4 Defensive Techniques

One of the most challenging tasks for a developer is protecting credentials and session cookies. To protect web applications, use the following defensive actions

- Password complexity
  The Passwords should contain a combination of alphanumeric characters., e.g., letters, numbers, and punctuation marks.
- Session ID should not be visible
  Use the POST method or cookie instead of the GET method because it is a significant issue that shows variables that include a session ID in the path string of the browser [21].
- Use powerful encryption for transference
  Should use robust encryption algorithms to

Table 1. A summary of broken authentication previous studies

Ref	Purpose/Motivation	Methodology	Result	Limitation/ Future Direction
[3]	Three effective remote user authentication systems, i.e. the system of Mishra *et al.*, the system of Wu *et al.* and the system of Moon *et al.* was reviewed and show that considering all of them, they are fitted with a structured safety proof	The scheme is reviewed and analyzed.	The system of Mishra *et al.* also fails to retain user confidentiality and is subject to a new form of insider attack, while the system of Wu *et al.* does not withstand a de-synchronization attack.	-
[6]	The paper aims to discover the limitations of broken authentication and session management.	The proposed algorithm would assist organizations and developers in addressing vulnerabilities and enhancing overall protection. Depending on ASP.NET files and the code-behind files (C sharp C#), our scanner the the tool relies on learning the application's source code.	The algorithm can assist organizations and developers to address vulnerabilities and boost overall security.	On some online websites while checking their algorithm. They cannot get a live website at all.
[8]	Present two plans for authentication security for mobile.	-	Review two plans that did not accomplish security authentication and suggest two principals	Designed strong schemes
[9]	Detect the attack at the right time so that it does not bring the entire system down.	A packet tracker checks the input request to trace the request for any malicious script which can be crafted by the attacker.	They implemented the the proposed system in ASP.Net framework and the accuracy percentage equal 97.8%	Enhance the proposed architecture to detect these two attacks.
[12]	Discuss the top ten web security vulnerabilities and how to prevent the web application from three of the vulnerabilities.	The proposed individual prevention architecture for SQL Injection, Cross-Site Scripting, and Broken Authentication by implementing the proposed models.	By implementing the proposed models, they have prevented a real-life website from those external attacks.	-
[14]	Discusses an evaluation and review of the vulnerability of Broken Authentication and Session Management and its five kinds of exploitation.	Literature Review	Note that the probability of abuse addressed would be minimized once the developer follows the prevention strategies outlined in this paper.	Intended to work on other manipulation methods and explore other broken website authentication and session management flaws.
[15]	Discuss relation (VOLtA) with machine learning.	-	They discover that (VOLtA) is not protected with machine learning.	-
[16]	Get to the root of security and web application problems.	Try something different.	It did not succeed.	-
[17]	Calculating the rating risk for OWASP top 10 and Mapping the co-occurrence of high-profile vulnerability types from both OWASP Top 10 and CWE/SANS Top 25.	-	-	-
[18]	The latest vulnerability surveys.	Organizations need to recognize maximum vulnerabilities and mitigate risks in order to remain secure. To do that it must be performed evaluation vulnerability and testing penetration on a daily VAPT basis.	Vulnerability must be prioritized and explained with CVE numbers that can be used from industry standard references to render VAPT results are meaningful.	The vulnerability evaluation does not define the vectors of logical attacks.
[19]	Invite a new way of authentication.	-	Change the method way to get successful authentication.	-
[20]	Discuss the issue of car communication through WAVA.	-	Authentication protocol failed to achieve.	-

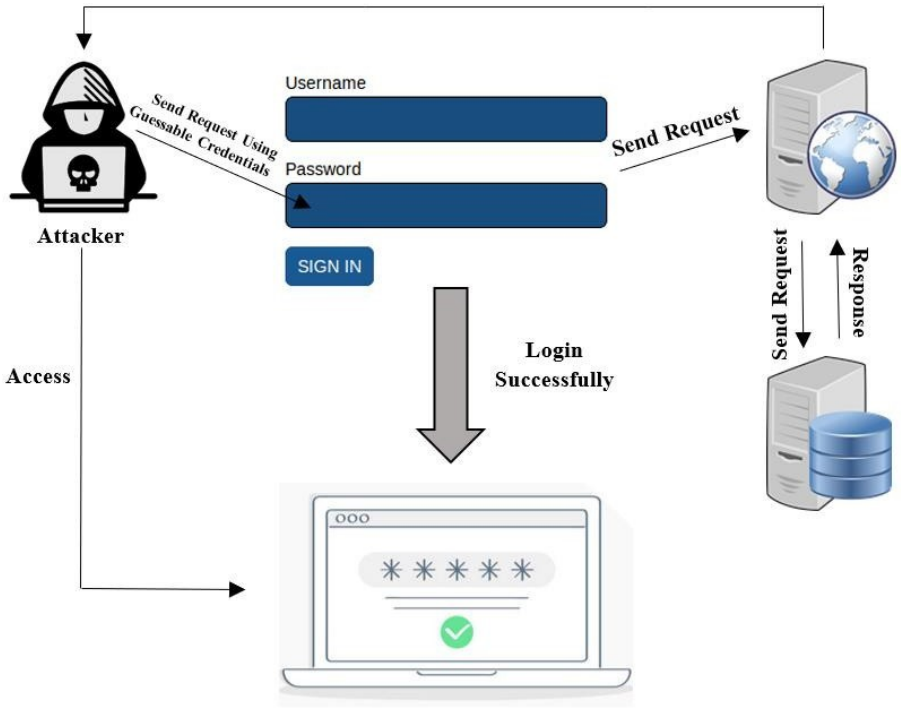

Figure 4. Exploitation of general broken authentication and session management [13]

protect the confidentiality of customer information transmitted over the internet, especially highly sensitive information like customer's login credentials [21].

- Session IDs should expire

  Logout should be forced after a short while when the user is inactive; this will reduce the chances of an attacker coming into an active session. Also, avoid constant login sessions [21].
- Username/Password Enumeration

  Be sure to display only generic error messages without specifying which part of the authentication data was wrong. e.g., instead of the "Invalid username" or "Invalid password" message, show the "Invalid username and/or password" message.

5 Conclusion

For many companies, risk detection is still a problem. In other cases, it is a usual way of operating that provides the requisite protection to keep the product free of weaknesses. Using various types of software to identify different security vulnerabilities assists both developers and organizations in securely launching applications, saving time and money. Different combinations of tools have been seen to enhance protection in recent years, but it has not been possible to combine the types of tools available on the market until the writing of this report. This paper revealed that the most powerful ways to exploit the Broken Authentication and Session Management vulnerabilities of the web application in those domains are the Session Misconfiguration assault and Cracking/Guessing Weak Password [7]. Also included techniques to defend authentication, and the most important, are using a robust encryption system, setting password rules, and securing the session ID.

References

[1] Auth. What is broken authentication, 2021. https://auth0.com/blog/what-is-broken-authentication/.Accessed 18 February 2022.

[2] Broken. Broken authentication, 2021. https://www.davosnetworks.com/broken-authenticatio. Accessed 18 February 2022.

[3] Cyber. Cyber security, 2021. https://www.jigsawacademy.com/blogs/cyber-security. Accessed 18 February 2022.

[4] Dheerendra Mishra, Ashok Kumar Das, Ankita Chaturvedi, and Sourav Mukhopadhyay. A secure password-based authentication and key agreement scheme using smart cards. *Journal of Information Security and Applications*, 23:28–43, 2015.

[5] Fan Wu, Lili Xu, Saru Kumari, Xiong Li, and Abdulhameed Alelaiwi. A new authenticated key agreement scheme based on smart cards providing user anonymity with formal proof. *Security and Communication Networks*, 8(18):3847–3863, 2015.

[6] Nguyen Manh Thang. Improving efficiency of web application firewall to detect code injection

attacks with random forest method and analysis attributes http request. *Programming and Computer Software*, 46(5):351–361, 2020.

[7] Wenting Li, Yaosheng Shen, and Ping Wang. Breaking three remote user authentication systems for mobile devices. *Journal of Signal Processing Systems*, 90(8):1179–1190, 2018.

[8] Rupal R Sharma and Ravi K Sheth. Discover broken authentication and session management vulnerabilities in asp .net web application. *Programming and Computer Software*, 3(1):290—-293, 2017.

[9] Wenting Li, Qianchen Gu, Yiming Zhao, and Ping Wang. Breaking two remote user authentication systems for mobile devices. In *2017 ieee 3rd international conference on big data security on cloud (bigdatasecurity), ieee international conference on high performance and smart computing (hpsc), and ieee international conference on intelligent data and security (ids)*, pages 37–42. IEEE, 2017.

[10] Virginia Mary Nadar, Madhumita Chatterjee, and Leena Jacob. A defensive approach for csrf and broken authentication and session management attack. In *Ambient Communications and Computer Systems*, pages 577–588. Springer, 2018.

[11] Md Fazlul Haque, Mohammad Badrul Alam Miah, and Fuyad Al Masud. Enhancement of web security against external attack. *European Scientific Journal, ESJ*, 13(15):228, 2017.

[12] Chanchala Joshi and Umesh Kumar Singh. Performance evaluation of web application security scanners for more effective defense. *International Journal of Scientific and Research Publications (IJSRP)*, 6(6):660–667, 2016.

[13] Md Maruf Hassan, Shamima Sultana Nipa, Marjan Akter, Rafita Haque, Fabiha Nawar Deepa, Mostafijur Rahman, Md Asif Siddiqui, Md Hasan Sharif, et al. Broken authentication and session management vulnerability: a case study of web application. *International Journal of Simulation Systems, Science & Technology*, 19(2):6–1, 2018.

[14] Jiliang Zhang and Haihan Su. Machine learning attack and defense on voltage over-scaling-based lightweight authentication. *arXiv preprint arXiv:1807.07737*, 2:50–55, 2018.

[15] Daniel Huluka and Oliver Popov. Root cause analysis of session management and broken authentication vulnerabilities. In *World Congress on Internet Security (WorldCIS-2012)*, pages 82–86. IEEE, 2012.

[16] Jinfeng Li. Vulnerabilities mapping based on owasp-sans: a survey for static application security testing (sast). *Annals of Emerging Technologies in Computing (AETiC), Print ISSN*, pages 2516–0281, 2020.

[17] Kyriakos Kritikos, Kostas Magoutis, Manos Papoutsakis, and Sotiris Ioannidis. A survey on vulnerability assessment tools and databases for cloud-based web applications. *Array*, 3:100011, 2019.

[18] Detchasit Pansa and Thawatchai Chomsiri. Integrating the dynamic password authentication with possession factor and captcha. In *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 530–535. IEEE, 2018.

[19] Mark A Runco. Comments and corrections: Chance and intentionality in creative performance. *Creativity Research Journal*, 19(4):395–398, 2007.

[20] Subir Biswas and Jelena Mišić. A cross-layer approach to privacy-preserving authentication in wave-enabled vanets. *IEEE Transactions on Vehicular Technology*, 62(5):2182–2192, 2013.

[21] Luke Murphey. Secure session management: Preventing security voids in web applications. *The SANS Institute*, 29, 2005.

Hanan Aljoaey is a graduated student from Computer Science department since 2012, Umm Al-qura University, Saudi Arabia. She is currently a Master thesis student in cybersecurity program at Department of Information Technology, College of Computer, Qassim University, Saudi Arabia.

Khawla Almutawa is a graduated student from Computer Science department since 2020, College of Computer, Qassim University, Buraydah 51941, Saudi Arabia. She is currently a Master thesis student in cybersecurity program at Department of Information Technology, College of Computer, Qassim University, Saudi Arabia.

Ruyuf Alabdali is a graduated student from Information Technology department since 2016, Lawrence Technology University, Saudi Arabia. She is currently a Master thesis student in cybersecurity program at Department of Information Technology, College of Computer, Qassim University, Saudi Arabia.

Dina M. Ibrahim Assistant Professor at Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia from September 2015 till now. In addition, Dina works as an Assistant Professor in the Computers and Control Engineering Department, Faculty of Engineering, Tanta University, Egypt. She was born in the United Arab Emirates, and her B.Sc., M.Sc., and Ph.D. degrees have taken from the Computers and Control Engineering Department, Faculty of Engineering, Tanta University in 2002, 2008, and 2014, respectively. Dina works as a Consultant Engineer, then a Database administrator, and finally acts as a Vice Manager on Management Information Systems (MIS) Project, Tanta University, Egypt, from 2008 until 2014. Her research interests include networking, wireless communications, machine learning, security, and the Internet of Things. She is serving as a reviewer in Wireless Network (WINE) the Journal of Mobile Communication, Computation, and Information since 2015, and recently in the International Journal of Supply and Operations Management (IJSOM). Dina has also acted as a Co-Chair of the International Technical Committee for the Middle East Region of the ICCMIT conference since 2020.