SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Vulnerability Assessment Studies of Existing Knowledge-Based Authentication Systems: A Systematic Review

Temitayo Caroline Adeniran[1*], Rasheed G. Jimoh[2], Emmanuel U. Abah[3], Nasir Faruk[4], Emmanuel Alozie[5], Agbotiname Lucky Imoize[6]
[1,3]Department of Telecommunication Science, University of Ilorin, Nigeria
[2]Department of Computer Science, University of Ilorin, Nigeria
[4,5]Department of Information Technology, Sule Lamido University, Kafin Hausa, Nigeria
[6]Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Lagos, Nigeria.
adeniran.tc@unilorin.edu.ng[1*], jimoh_rasheed@unilorin.edu.ng[2], eaustin890@yahoo.com[3], faruk.n@slu.edu.ng[4], alozie.m@slu.edu.ng[5], aimoize@unilag.edu.ng[6]

[*]Correspondence Author

Abstract

In the past decades, web applications have become a vital part of our daily activities, while hacking activities have also increased speedily. Both individuals and organizations are collectively facing significant challenges in securing their applications due to the high rise of threats and attacks. It is now expedient for developers to look for loopholes in authentication techniques to prevent attackers from exploring such weaknesses. Vulnerability Assessment and Penetration Testing have proven to help organizations determine if their security measures are working perfectly. Hence, it is crucial to ascertain these flaws and install security patches to mitigate them. This study illustrates a non-intrusive systematic overview of various knowledge-based authentication techniques, vulnerabilities, and attacks. It also describes the significance of data, the effects of vulnerabilities over data, and the tools used in detection and prevention. Finally, it focuses on creating awareness and the importance of adopting up-to-date security measures to stay protected from attacks.

Keywords: *Knowledge-based authentication techniques; Vulnerability assessment; Penetration testing; Security; Attacks.*

1. Introduction

Nowadays, security has filtered into the top of the list of business goals of most organizations, and protecting any system's security of all services rendered to a user is equally an important task (Al-Khurafi & Al-Ahmad, 2016; Alaidaros & Albeedh, 2022; Katsini et al., 2016; Shah & Shah, 2013). Studies have also shown that there is no single definition of security. It means privacy in healthcare systems, secrecy in the military systems, availability in e-commerce/marketing systems, and integrity in the banking systems (Alomar et al., 2017). However, in the context of Information security, the security goals can be classified into confidentiality, integrity, and availability. On the other hand, their corresponding attacks are Interception, an attack on confidentiality, modification and fabrication, attacks on integrity, and denial of service, an attack on availability (Alozie et al., 2023; Lu et al., 2021; Sikiru et al., 2023; Wakili et al., 2023). Moreover, this study focuses not on the overall security of information but on authentication, which is considered a subset of security. Authentication, according to Aravindhan & Karthiga (2013), Park (2018), Plata & Calpito (2020), Rice (2012), and Xiaoyuan et al., (2005), is defined as the act of confidence establishment in the truth of identities of users claimed by an information system.

Knowledge-Based Authentication (KBA), also known as cognitive passwords, was referred to as a form of user authentication that requires verification of identity claimed by matching one or more pieces of information, otherwise known as factoids, presented by the user or claimant against the information sources associated with the claimant (Basharzad & Fazeli, 2017; Parmar et al., 2022). These pieces of information, also known as factoids, can be static, such as "mother's maiden name," or dynamic, such as bank withdrawals and mortgage payments, personal,

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

such as “your favorite movie or team,” or non-personal, such as “your phone number and job title”. KBA was thought to be ideal for lessening the existing over-reliance on traditional passwords in the search for the best and easiest ways to ensure the authentication of entities that would provide a significant level of security (Alphanumeric). Despite the fact that the complexity and security issues provided by conventional passwords have caused consumers to worry about their privacy, these issues include identity theft, the loss of sensitive data, and others (Alhakami & Alhrbi, 2020; Mohan & Harun, 2022; Sani et al., 2022; Wu et al., 2021). Some studies have shown that KBA use may be not only secure but also easy and effective. This is a result of humans doing better with knowledge information rather than memorizing some set of passwords (Erdem & Sandikkaya, 2018; Ghiyamipour, 2021; Khurana et al., 2022; Sadikan et al., 2019).

KBAs come in two varieties: static and dynamic. Banks, commercial service providers, and email providers frequently employ static KBA, sometimes known as "shared secret questions," as a kind of identity proof for customers or as a fallback authentication whenever an authorized user forgets his or her password (Albayram & Khan, 2016; Javed et al., 2014; Parmar et al., 2022; Rabkin, 2008). Nonetheless, a company or individual employing static KBA must gather the data, which consists of questions and their responses, to be communicated between the claimer and the verifier. Sadly, it is the most widely used and least effective KBA type that almost everybody is familiar with. It employs standby old questions like (what is your mother's maiden name?) Instead, "Where were you born? Thus, it is said to be the type of KBA that a fraudster can break the easiest (Park, 2018). On the other hand, dynamic KBA, also known as "out of wallet" questions, is a high level of authentication that makes use of knowledge questions and answers for verification but does not require prior contact. The questions are said to be generated on the fly, and the answers are only known to authorized users, making it much more difficult for a fraudster to guess them correctly. The response to this query must take into account the circumstances. Moreover, responses are produced from the data records according to the specified human identity. For instance, getting data from third-party systems or public documents and not necessarily storing the answers (Just & Aspinall, 2009; Parmar et al., 2022; Sadikan et al., 2019).

KBA has several benefits and many systems, including Facebook and Bank of America, employ this approach to verify that users are valid, especially the dynamic KBA, however, it is not without challenges or vulnerabilities (Aljohani et al., 2016; Parmar et al., 2022). Previous research efforts have proposed and evaluated an enhanced dynamic KBA for social media (Adeniran et al., 2023). This paper aims to assess the vulnerabilities in KBA systems by conducting a non-intrusive systematic overview of various knowledge-based authentication techniques, vulnerabilities, and attacks. It also describes the significance of data, the effects of vulnerabilities over data, and the tools used in detection and prevention. Finally, it focuses on creating awareness and the importance of adopting up-to-date security measures to stay protected from attacks.

The remaining parts of this paper are organized as follows: Section 2 dwells on the knowledge-based authentication methods. Section 3 presents the review methodology. Section 4 gives the Vulnerability Assessment and Penetration Testing (VAPT). The key findings are briefed in Section 5 and the conclusion to the review is given in Section 6.

2. Knowledge-Based Authentication (KBA)

KBA is known as one of the easiest and simplest techniques for gaining additional confidence in a user’s identity. This view is supported by (Alhakami & Alhrbi, 2020; Just & Aspinall, 2009) that KBA is referred to be an Information-based Identity Authentication. It is a form of an authentication technique consisting of textual and graphical passwords tied to something the claimer knows, such as recall-based passwords such as text-based or alphanumeric passwords, One Time Passwords (OTPs). Also, as shown in Figures 1 and 2, KBA can be recognition-based or cognitive examples are color-based, image-based, and challenge-response protocols (Akula & Devisetty, 2004; B. Patel et al., 2021; Rajesh et al., 2015; Saranya et al., 2017; Shah & Shah, 2013; Verma, 2012). In the sequel, and staying within the scope of this study, the focus is thereby on the challenge-response KBA. An interactive question-and-answer procedure supports this kind of knowledge-based authentication technique. The KBA system is rumored to run password security protocols that, if a user forgets their passwords, encourage them

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

to answer at least one personal question. By the use of scoring models and algorithms, the technique analyzes the information already available about a user with the information provided to identify that person (Lu et al., 2021; Mell et al., 2006). Also, the study showed that KBA is typically utilized as a part of Multifactor Authentication (MFA) and for self-retrieval services, keeping in mind that authentication might be a single factor or multifactor (Alsaiari et al., 2016; Alsaleem & Alshoshan, 2021; O, 2015; Zangooei et al., 2012). However, two types of KBA use challenge-response questions and answers; they are static KBA and dynamic KBA (Basharzad & Fazeli, 2017; Parmar et al., 2022).

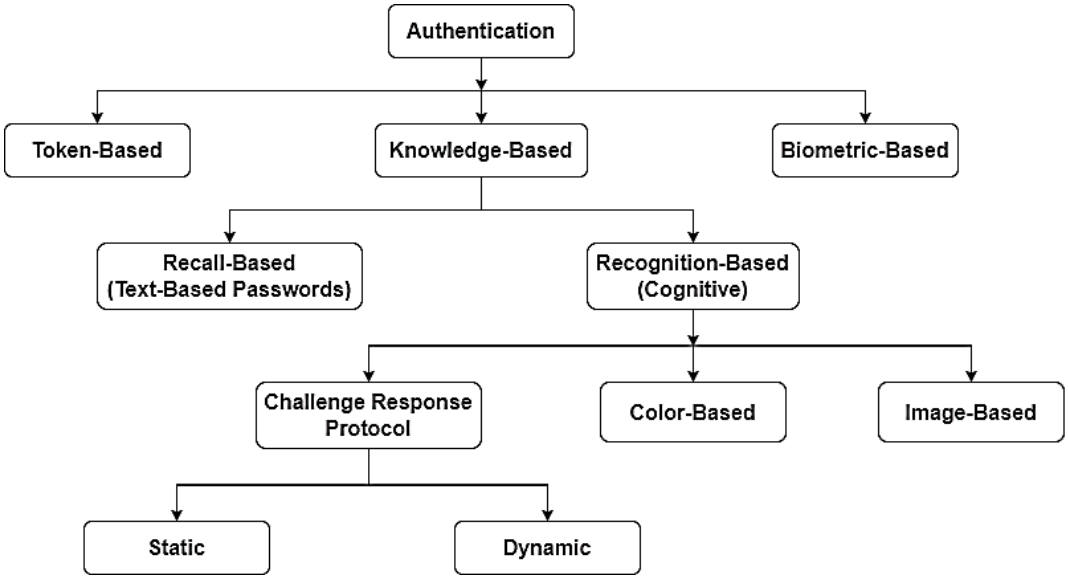

Figure 1: Authentication Types (Alsaiari et al., 2016; Maifi et al., 2015)

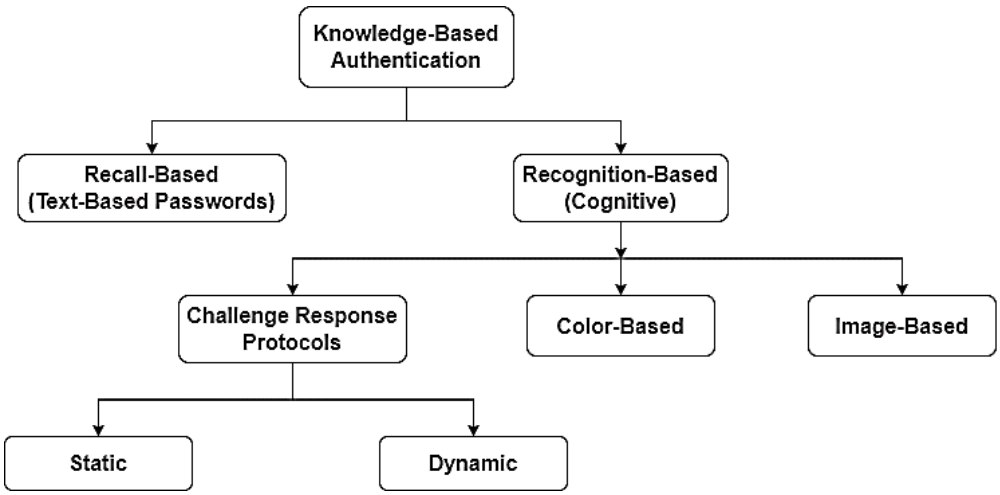

Figure 2: Knowledge-Based Authentication Types
(Alhakami & Alhrbi, 2020; Haque & Imam, 2014; Katsini et al., 2016)

2.1 Recall-based (Static Text-based Passwords)

The recall-based passwords or static text-based passwords are best known through their major implementation. A password as defined by (Aljohani et al., 2016), is a list of alphanumeric characters, which is usually between six to ten characters in length. The scheme makes use of user identification/password form-based authentication. Generally, the claimant proves to the verifier that he/she knows a set of characters, otherwise known as passwords which have been initially stored in a verification table. The verifier then compares it to an already

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

stored value to reveal its authenticity. However, static text-based password authentication suffers from many vulnerabilities and drawbacks and is thereby considered a low-security solution technique when used as a standalone mechanism or without putting the necessary precaution in place. Often, using the same passwords in different social networks is common among users because it simplifies the high rate of password memorability. However, this is considered to be a challenge because of repetition and password circulation (Alhakami & Alhrbi, 2020). Therefore, studies have shown that the scheme is vulnerable to guessing attacks, brute force attacks, or dictionary attacks. Also, according to Hameed (2015) and Zulkarnain et al., (2013), passwords can be stolen by an eavesdropper online to have access to an unauthorized system. This is known as a replay attack. Similarly, physical attacks can also be carried out by an attacker using a keylogger or camera to record a sequence of password as it is being typed (Adamu et al., 2022; Alsaiari et al., 2016). Also, a stolen password hash could be reverted by an attacker, using a list of pre-recorded hashes of the most common passwords (Adamu et al., 2022; Haque & Imam, 2014). However, increasing the strength of a password is a solution to mitigate dictionary attacks or to make brute-force attacks infeasible. As claimed by (He et al., 2020), the length of a password determines the provided security. While this is true, it was also argued that the strength of the password is related to its entropy. Conversely, entropy refers to how users select random passwords from a given key space. It also relates to how difficult passwords can be guessed by attackers (Guan & Chen, 2022; Katsini et al., 2016). However, despite the widespread use of passwords, the main concern is the lack of security during transmission over a channel.

2.2.1 Recall-based (One-Time Passwords)

One-time passwords, popularly known as OTPs, are considered to be an evolution of the traditional static ID/password system (Erdem & Sandikkaya, 2018; Zhao & Luo, 2017). One of the main drawbacks of static passwords is the lack of protection against replay attacks, hence the evolution of OTPs techniques to mitigate the replayability of static passwords by generating a new password for each use (Aravindhan & Karthiga, 2013; Asha & Jingle, 2022; Gong et al., 2013). This scheme makes it suitable to secure different online payments and financial services by generating passwords valid for single use, then expiring (Plata & Calpito, 2020).

2.3 Recognition-Based/Cognitive

2.3.1 Static KBA (shared secret questions)

Aravindhan & Karthiga (2013) and Schechter et al., (2009) study on Static KBA shows that it is based on a pre-agreed set of shared secrets. That is, users select questions to be asked and the corresponding answers. Banks, commercial service providers, and email providers frequently use this as identification verification to enable account access or as a backup, if a user forgets their password (Albayram & Khan, 2016; Alhakami & Alhrbi, 2020; Javed et al., 2014; Rabkin, 2008; Renaud & Just, 2010). But a company or individual employing static KBA must gather the data to be communicated between the claimer and the verifier, i.e., the question and the associated response, as the success of static KBA is predicated on the user's capacity to memorize the secret (Aljohani et al., 2016; Tran, 2022). It is also anticipated that a user's identity would be verified if they know the right answers to the secret question. As shared secret questions can be used to reset account passwords, security risks, including password manipulation and denial of service, are possible (Freimanis, 2019; Park, 2018).

2.3.2 Dynamic KBA (DKBA)

Dynamic KBA on the other hand, is authentication of a high level that uses knowledge questions and answers for identity verification, which also requires no previous contact, as seen in static KBA (Okamoto, 2013; Tran, 2022).

In the DKBA scheme, the end-user has no idea what would be asked, instead, questions and answers are determined by harvesting data in the public record of the user's personal data file also known as a third-party system (Just & Aspinall, 2009; Kavianpour et al., 2019). However, users must supply fundamental identity information such as name, address, date of birth, and the like to start dynamic KBA. Also, the information required to respond to those "out of wallet questions" is not currently stored in the wallet. However, because it is not based on an existing relationship with a user, dynamic KBA has been used in many organizations to authenticate user identities to prevent

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

fraud, compliance/adherence, etc. This gives businesses a means to have greater assurance of identity on client identification during the origination of accounts (Alomar et al., 2017; Plata & Calpito, 2020).

Amongst other areas of KBA use, KBA has been successfully implemented in financial services requiring originating, credit scoring, processing, fraud, and identity theft detection (Basharzad & Fazeli, 2017; Javed et al., 2014; Okamoto, 2013; Polakis et al., 2014; Wu et al., 2021). Nowadays, challenge-response questions and answers are employed at a wide range of websites, including web-based email providers (Yahoo & Google). Online merchant shops like eBay have also effectively implemented KBA to assure client protection, account recovery, and management (Bonneau et al., 2015; Parmar et al., 2022; Pranathi et al., 2018; Schechter et al., 2009). Another significant area of use of KBA by government organizations is to identify and authenticate state citizens (Alhakami & Alhrbi, 2020). KBA is also taken into account as a part of the identity-verification procedures approved for the uniting and strengthening of America, providing the appropriate tools required to intercept and block terrorism (USA Patriot ACT 2001)" by the American government. Despite the wide usage of the KBA in practical applications, a number of issues have been raised regarding the benefits and drawbacks of the authentication techniques in terms of usability and security (Albayram & Khan, 2016; Aljohani et al., 2016; Bonneau et al., 2015; Ghiyamipour, 2021; Renaud & Just, 2010; Towhidi & Masrom, 2009; Tran, 2022).

2.3.3 Color-Based (KBA)

Another form of graphical or recognition-based technique is the color-based authentication method (Rajesh et al., 2015). This technique can be used as a multifactor authentication system capable of mitigating security attacks like dictionary attacks, phishing attacks, and brute force attacks by providing a better and additional layer of security (Mamachan, 2019; Rajesh et al., 2015; Saranya et al., 2017; Verma, 2012). Studies have it that this technique is suitable for sensitive applications like defense and banking applications (Hameed, 2015). However, it has also been confirmed that studies are needed in this area for successful implementations (Hameed, 2015; Mamachan, 2019).

2.3.4 Image-Based (KBA)

Research suggests that the use of images for authentication may be more effective in terms of ease of use and security for some applications. This is because humans tend to recognize images better than remember passwords (Shah & Shah, 2013). Akula & Devisetty, (2004) proposed an imaged-based authentication system that is easy for internet applications to guard against brute force attacks and guessability attacks. The scheme allows first-time users to register by submitting a user ID and an image as credentials before logging into the system. The authors affirmed that the images submitted are not stored in the system, rather, the images are read in bytes and hashed using SHA-1, a secure hashing function. They further claimed that images are generally large files. However, SHA-1 algorithms could produce an output as small as 20 bytes which is secure and require less memory space. Similarly, Ullah et al., (2019) confirmed that quite several online services had deployed the use of images for authentication, e.g. the Bank of America makes use of a site key image together with text-based challenge questions. Hence, the scheme can also be used as a multifactor authentication system in that it could work with other authentication systems to enhance security strength. For example, CAPTCHA is an extension of an image-based authentication system (Kim et al., 2012).

2.3.5 Comparative Studies of KBA Methods

All authentication methods/techniques are distinct, and each response is unique to specific needs (Alhakami & Alhrbi, 2020; Tellini & Vargas, 2017). However, different implementations of the same authentication techniques will generate very dissimilar levels of usability and security. For instance, it has been proven theoretically that even with the strongest authentication techniques, if the enrolment/registration stage is poor, it could lead to poor reliability on the user identities. Moreover, each authentication method still has its intrinsic pros and cons see Table 1. Table 2 illustrates a comparative study of KBA methods on two broad criteria and sub-criteria, as seen in Figure 3

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

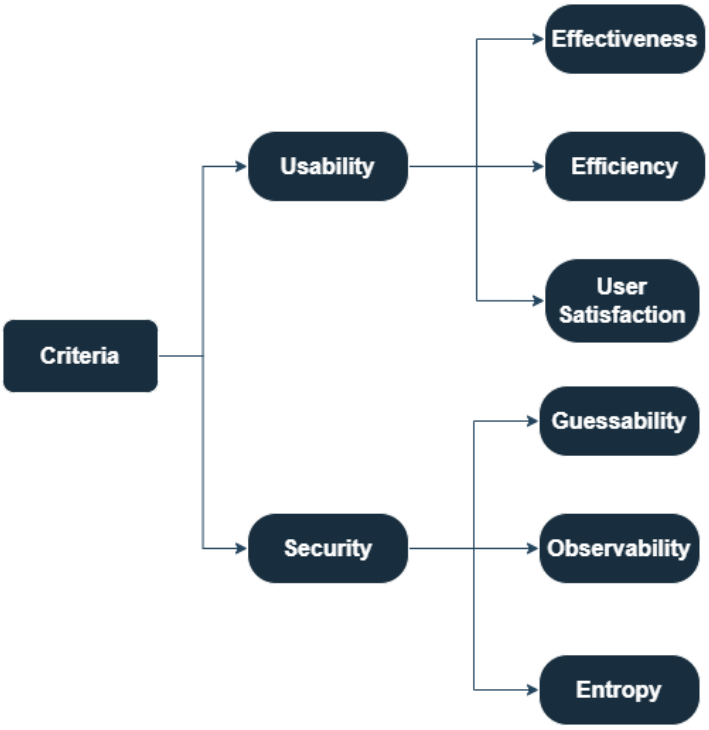

Figure 3: Key Performance Criteria/Evaluation Metrics of KBA Systems

Most KBA systems' performance was evaluated on two broad criteria, usability, and security, that is, with respect to ease of use and security strength of a system, as depicted in Figure 3. Furthermore, three major dimensions were defined with respect to usability, they are:

1. **Effectiveness:** This is associated with memorability, and it is defined in terms of the number of users who forgot answers to the authentication questions and also the number of wrong entries during authentication.
2. **Efficiency:** refers to the time taken to answer authentication questions and the data entry speed.
3. **User satisfaction:** this is not limited to the ease of use of the proposed system. It also refers to user perception and experience when compared to the traditional password authentication system.

However, likewise criterion security, three major indices were also defined to measure the security level of the system, they are:

1. **Guessability:** this is the ability of a fraudster, otherwise known as an attacker, to correctly guess the answers to the challenge-response authentication questions.
2. **Observability:** this simply means the ability of a fraudster or an attacker to observe the correct answers as the authorized user enters them.
3. **Entropy:** this is a security metric per policy used to measure the uncertainty of choices. It also relates to how difficult a fraudster can guess the correct answers to the challenge response questions.

Table 1: Pros and Cons of KBA

Authentication Type/Category	Authentication Mechanism	Pros	Cons
Recall-Based Passwords	Static Text-based Password	Easy to use and does not require any special hardware	Requires high memorability and 100% Accuracy.
Recall-Based Passwords	One Time Password	More Secure than static Text-based Password	Complex implementation

SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Cognitive Based Authentication	Image-Based Authentication	Prevent Brute Force Attacks	Time-consuming
Cognitive Based Authentication	Color-Based Authentication	User friendly	Time and energy consuming
Cognitive Based Authentication	Challenge Response Protocols (Static Questions & Answers)	Easy to remember.	• Questions and answers generated are static i.e., unchanging. • Vulnerable to guessing attacks
Cognitive Based Authentication	Challenge Response Protocols (Dynamic Question& Answers)	Generated authentication questions and answers are dynamic, i.e., changes always. Low memorability	

Table 2: Comparative Study of KBA Methods

Method	Usability			Security		
	Effectiveness	Efficiency	User Satisfaction	Guessability	Observability	Entropy
Static Text-Based Password	Medium	High	High	High	High	
OTP	High	Medium	Medium	Low	Low	
Image-Based	High	Low	Medium	Low	Low	
Color-Based	High	Low	Low	Low	Low	
Challenge Response Q&A (Static)	High	High	High	High	Medium	
Challenge Response Q&A (Dynamic)	High	High	Low	Low	Low	

3. Review Methodology

This research study employed a systematic and descriptive literature review (SDLR), conducted in compliance with the PRISMA method and guidelines see Figure 4. The previously described SDLR offers detailed instructions for disclosing systematic reviews and meta-analyses based on evidence-based sets of data. However, the SDLR complements the existing studies in the areas of KBA and its associated vulnerabilities and how to address these vulnerability issues of KBA. The process is described in detail and summarized in the following section.

3.1 Search Criteria for Selection of Primary Studies

The SDLR covered the peer-reviewed trend of publications and articles over the past 18 years, i.e., starting from the year (2005-2023), including the most recent articles from top journals, conference proceedings, edited books, etc. Studies were carried out by searching for keywords in reputable electronic databases and search engines. The defined keywords include:

- Authentication methods/Techniques

SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

- Knowledge-Based Authentication
- Text-Based Passwords
- Static KBA
- Dynamic KBA
- Recognition/Recall based Passwords
- Token-Based Authentication
- Image-Based Authentication
- Knowledge-Based Authentication Vulnerabilities
- Knowledge-Based Authentication Performance Metrics

In the same way, searches were done through search engines using the paper title, abstract, or keywords. Yet, some search engines produced a large number of irrelevant results due to a lack of advanced choices. Also, the addition of indexing data sources made it possible to search for publications broadly. As a result, even those publications that were not initially included in publishing outlets were later retrieved in full after the inclusion criteria were assessed. The reason why broad guidelines were adopted was to make sure that there were no potentially relevant studies mistakenly excluded from further assessment. However, the screening stage greatly reduced the number of full texts and papers that were eventually evaluated, allowing the focus to be on relevant papers. Hence, the above-defined keywords were retrieved from the following databases:

- IEEE Explore
- ACM Digital Library
- Elsevier Science Direct
- Springer Nature
- Taylor & Francis

3.2 Eligibility Criteria and Study Evaluation

The research papers were evaluated in light of a set of predetermined inclusion and exclusion criteria to ascertain the eligibility of all retrieved literature. The inclusion and exclusion standards used for the studies are outlined in Table 3. The full texts of the studies identified were acquired during the evaluation phase. Nonetheless, the study was disregarded if the entire text couldn't be located. Similarly, to this, before the examination, all duplicates that existed in multiple databases were eliminated. Additionally, for each database, up to 500 results were screened. The results outcome was, therefore, sorted out by relevancy before the inspection. In comparison, irrelevant papers were excluded following a defined set of guidelines during the screening stage. The guidelines are:

- If various types and categories/techniques/methods of KBA are mentioned or examined, this should be included; otherwise excluded.
- If KBA threats, vulnerabilities, attacks, and measures are addressed, then they should be considered hence excluded.
- If vulnerability assessments of KBAs and diverse issues and approaches are employed in different works, otherwise, excluded.

All these guidelines helped identify and analyze KBAs, highlighting their main strengths and flaws. It also helped to interpret all available research relevant to the research area and incorporated current work in a manner that is fair and seen to be fair.

Next is the stage of analysis, an article evaluated was expected to meet almost, if not all, the criteria for inclusion and none of the exclusion. Also, the requirements for original peer-reviewed papers are the publications' quality and content. Needless to say, the research questions were defined in accordance to:

- ✓ The state of the art of KBA, with respect to usability and security, and secondly,

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

- ✓ The theoretical and empirical evaluation of KBA vulnerability assessments of past studies.

Furthermore, to ascertain whether a particular paper meets up with the criteria, the full text of the paper was examined. Moreover, some databases (electronic) offered advanced search options that allowed the exclusion of certain papers, for example, (non-English papers and patents) without having to examine the full texts. But, the final set of papers is made up of studies that included experimental and theoretical evaluation of KBA vulnerabilities, which helped in quality assessment carried out to identify research bias and validation of experimental data. See Figure 5 for the publication trend for 17 years.

Table 3: Inclusion and Exclusion Criteria for the Studies

Inclusion Criteria	Exclusion Criteria
Peer-reviewed publications	Publications covering different Authentications in general
Original research studies	Publications not in the English Language
Publications on different KBA systems and methods	Secondary research and other non-relevant publications.
Publications addressing/assessing KBA threats, attacks, and vulnerabilities.	Patents and grey literature.
Research studies published within the years (2005- 2023)	

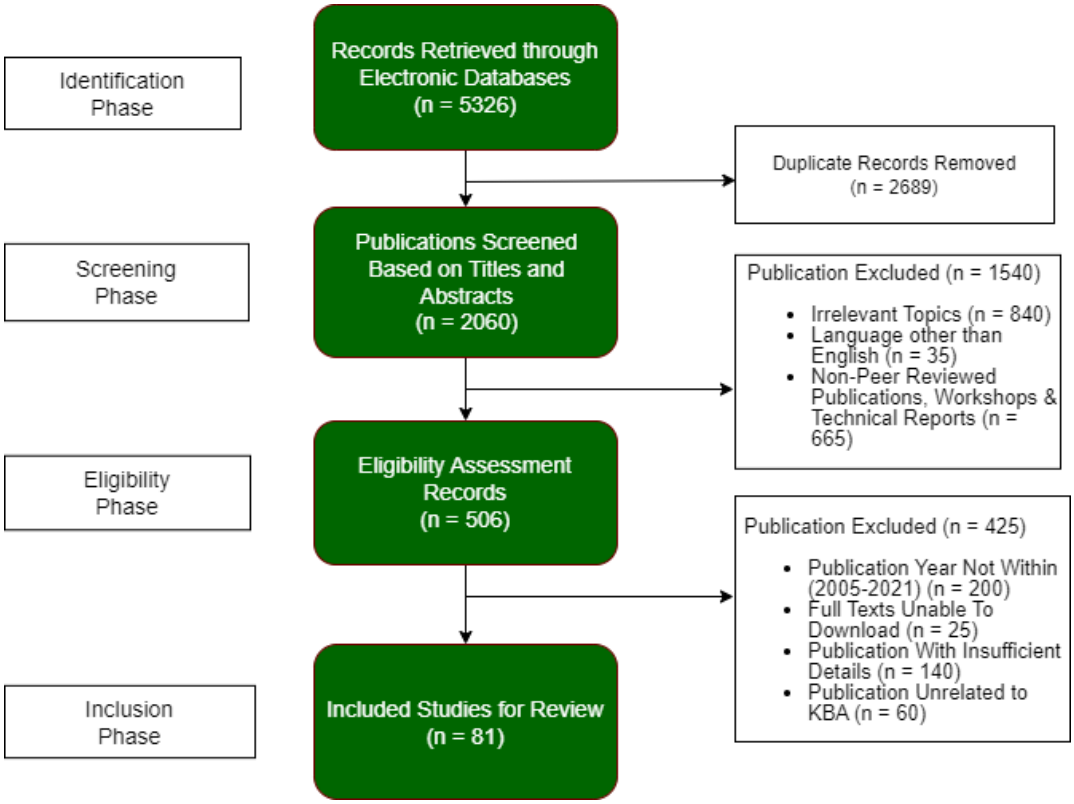

Figure 4: PRISMA Model

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

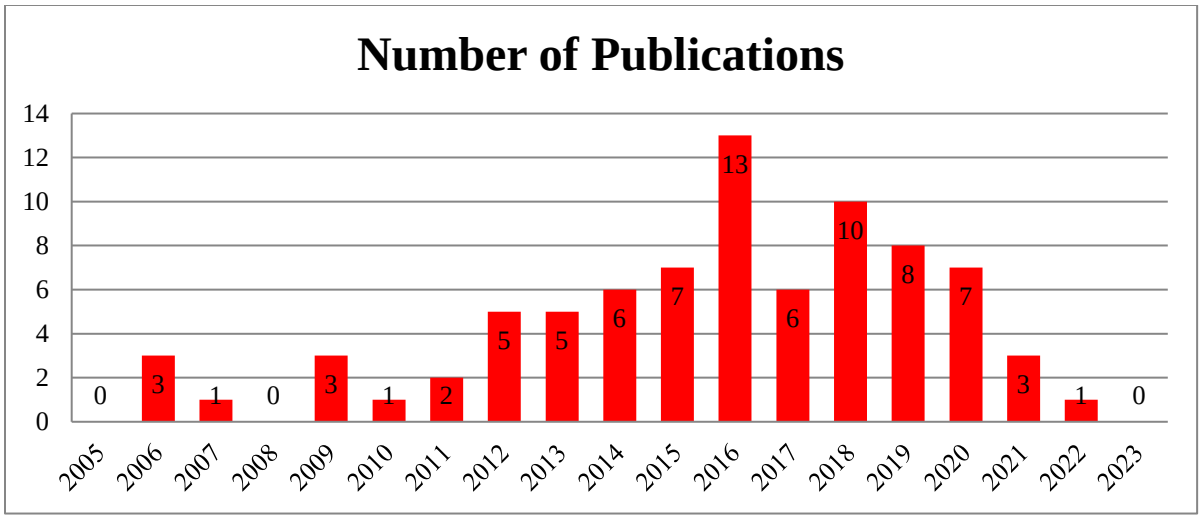

Figure 5: Trend of Publications (2005-2023)

3.3 Knowledge-Based Authentication Vulnerabilities, Attacks, and Mitigations

Some various attacks and threats exploit vulnerabilities or flaws of different KBA systems (Alhaidary & Rahman, 2016; Freimanis, 2019; Kim et al., 2020). This section describes the typical KBA vulnerabilities, attacks, and possible mitigations. To evaluate the security flaws or vulnerabilities of KBA, results from works of literature published in reputable journals and conferences were inspected carefully following the general guidelines of a systemic literature review process (Ahmed & A. Hikal, 2019; Bošnjak & Brumen, 2020). The research papers were collected using reliable literature search engines such as ACM Library, IEEE Explore, Google Scholar, Springer, and Elsevier. Search queries were formed in association with the following domain keywords (security vulnerabilities, authentication attacks, and KBA flaws).

Also, similar keywords like vulnerabilities and flaws are considered and both are used interchangeably. However, to exclude outdated literature, the scope of the search is limited to research studies published from 2005 to 2023. the number of searched works of literature for each database is as follows ACM Library = 105; IEEE Explore = 514; Springer Link = 1008. Elsevier = 1000, and Google Scholar = 2070; with a Total = 4697. However, after the initial search, a consistent keywords validation on the searched literature was done based on the same keywords because search engines may have processed each query differently in each database. Therefore, the first filtering returned 2008 literature. A quick examination of each research paper's title and abstract was then conducted because not all of the literature were inside the parameters of this research objective. As a result, a second filtering also produced 500 works of literature. Similarly, based on the review criteria of inclusion and exclusion detailed review was carried out on the filtered literature works to ascertain whether they fit within the scope of the objective. Finally, 60 works of literature were selected as the basis for the vulnerability taxonomy of KBA systems. Figure 6 shows the total number of finally selected articles from the academic databases after search criteria filtering.

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

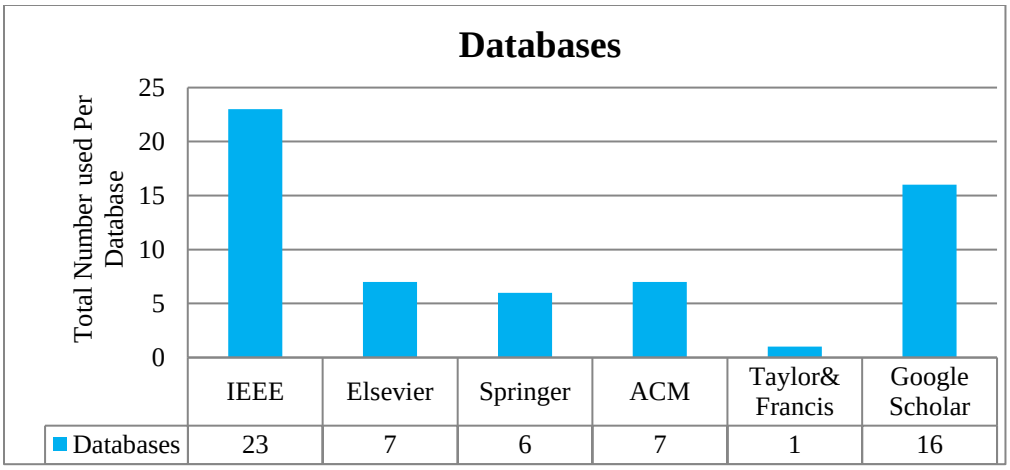

Figure 6: Total number of final filtered articles used from the database query search.

3.3.1 Case studies/Taxonomy of Vulnerabilities of KBA

Authentication systems are said to be vulnerable when they have potentialities for infiltration, which may result from weaknesses within the codes that make up the system (Awoleye et al., 2014). On the other hand, some underlying technicalities may cause other defects that could prohibit the successful operation and performance of the authentication systems (Katsini et al., 2016). Several weaknesses and vulnerabilities from the general security and authentication systems have reportedly been incorporated into KBA systems, which are specific sorts of authentication systems. These include, among others, SQL injection (Ahmed & A. Hikal, 2019; Lee & Kim, 2020); cross-site scripting (Alomar et al., 2017; Bošnjak & Brumen, 2020). Yet, Attackers are now focusing on the internet due to the popularity of the web among millions of internet users (Kadham & Sreenivasa Ravi, 2018; Kim et al., 2012). Awoleye et al., (2014) and Moen et al., (2007) both conducted research that found that 82% of websites around the world were susceptible to *cross-site scripting (XSS)* and query *language injection (SQL)*. Attackers may use these vulnerabilities to alter databases and introduce phishing attacks on weak servers. Awoleye et al. further affirmed that 90% of European, 49% of African, 76% of North American, and 85% of Asian websites are susceptible to web application attacks. Abirami et al., (2016) carried out a survey and revealed that SQL was top security vulnerability, predominant in web applications and it is expedient to protect our data stored from SQL injection attacks. This is because people rely mostly on social media communication and this increases the storage of electronic data in data warehouse. Hence, this makes data in databases, more prone to SQL injection attacks which have the will-power to destroy such data completely if no measures are put in place.

Also, Shinde & Ardhapurkar, (2016), analyzed various authentication vulnerabilities, corresponding attacks, and mitigations by combining Vulnerability Assessment and Penetration Testing (VAPT), with SQL and cross-site scripting, topping the list of vulnerabilities. He also proposed a literature survey on various VAPT techniques used by different researchers to find vulnerabilities in internet applications. Similarly, Alhaidary & Rahman, (2016) investigated and analyzed possible flaws and mitigations for password-based authentication systems using an Offline Personal Authentication Device (OffPAD). To mitigate attacks like man-in-the-middle and phishing, this device aims to give users access to tools that securely manage identification and authentication procedures for online transactions. However, he concluded that there are many other reasons an organization should implement authentication, such as monitoring, content filtering for both incoming and outgoing communications, access control, timing, and policy-making.

Cross-site scripting is the most well-known vulnerability in online applications and is very malicious since it opens the door for a variety of assaults, according to Taha & Karabatak, (2018) warnings. As a result, they suggested a strategy to stop cross-site scripting. Furthermore, it was noted that developers should use context-dependent output encoding to reduce the risk of XSS. Encoding and HTML special characters, like tag opening and closure, are two examples. In other words, changing (&lt or > with &gt), etc. Moreover, verifying user-provided

SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

data can help to mitigate the XSS issue. In 2019, Freimanis assessed the vulnerability of authentication techniques in a large-scale computer system to discover vulnerabilities in a target system using penetration testing techniques. It was concluded that administrators must be well-informed about existing flaws and how to use their accounts by restricting users to only necessary resources (Freimanis, 2019). Bošnjak & Brumen, (2020), however, carried out a systematic literature review (SLR) and a meta-analysis, in compliance with Kitchenham's guidelines, on how various shoulder surfing experiments were used to explore password vulnerabilities. To share information about current and emerging shoulder-surfing-resistant authentication techniques. Additionally, it encourages researchers to focus more on systematically evaluating any novel KBA authentication techniques that may be put forth, particularly with regard to how vulnerable they are to shoulder surfing attacks. After scanning no less than 130,000 applications, OWASP Top 10 presented the ten most prevalent web application security issues in 2021. It was found that nearly 68% of applications had flaws that fit into OWASP's Top 10 list of vulnerabilities. OWASP is not just a list but a technique that assesses each class of flaw using ***OWASP risk rating methodology*** and also provides guidelines for preventing possible attacks. This includes Cross-Site Scripting, Insecure Deserialization, Insufficient Authentication, Sensitive Data Exposure, XML External Entities, Security Misconfiguration, and Broken Access Control. Insufficient Logging and Monitoring. It was later determined that developers would be able to design secure programs that protect users' data from attackers by writing codes and performing comprehensive testing with these issues at heart. A thorough review of KBA techniques for augmented reality head-mounted displays was carried out by (Düzgün et al., 2022), and 31 distinct authentication schemes were gathered from 18 pertinent research papers. The evaluation metrics that were utilized to analyze the authentication systems were shown in the reviews' conclusions. The OWASP Top 10 Vulnerabilities for the previous 18 years are shown in Table 4, keeping in mind that this rating is always published every three to four years of the previous release. While Figure 7 depicts the trend and shift in the paradigm of vulnerabilities between 2017 and 2021.

Table 4: Top 10 Vulnerabilities as presented by Open Web Application Security Project (OWASP)

Vulnerability position	RANK IN THE YEAR					
	2004	2007	2010	2013	2017	2021
A1	Unvalidated input	Cross-site Scripting (XSS)	Injection Flaws	Injection Flaws	Injection Flaws	Broken Access Control
A2	Broken Access Control	Injection Flaws	Cross-site Scripting (XSS)	Broken Authentication & Session Mgt	Broken Authentication & Session Mgt	Cryptographic Failures
A3	Broken Authentication & Session Mgt	Malicious File Execution	Broken Authentication & Session Mgt	Cross-site Scripting (XSS)	Sensitive Data Exposure	Injection Flaws
A4	Cross-site scripting (XSS)	Insecure Direct Object Ref.	Insecure Direct Object Ref.	Insecure Direct Object Ref.	XML External Entities (XXE)	Insecure Design
A5	Buffer Overflow	Cross-site Request Forgery (CSRF)	Cross-site Request Forgery (CSRF)	Security Misconfiguration	Broken Access Control	Security Misconfiguration
A6	Injection Flaws (SQL)	Info leakage and Improper Error Handling	Security misconfiguration	Sensitive Data Exposure	Security misconfiguration	Vulnerable and outdated component

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

A7	Improper Error Handling	Broken Authentication & Session Mgt.	Failure to Restrict URL access	Missing Function Level Access Control	Cross-site scripting (XSS)	Broken Authentication & Session Mgt.
A8	Insecure Storage	Insecure Cryptographic Storage	Unvalidated Redirects and Forwards	Cross-site Request Forgery (CSRF)	Insecure Deserialization	Software and Data Integrity Failures
A9	App Denial of Service	Insecure Communication	Insecure Cryptographic Storage	Using Known Vulnerable Components	Using Known Vulnerable Components	Insufficient Logging and Monitoring
A10	Insecure Config. Mgt	Failure to Restrict URL access	Insufficient Transport Layer Protection	Unvalidated Requests & Forwards	Insufficient Logging and Monitoring	Server-Side Request Forgery (SSRF)

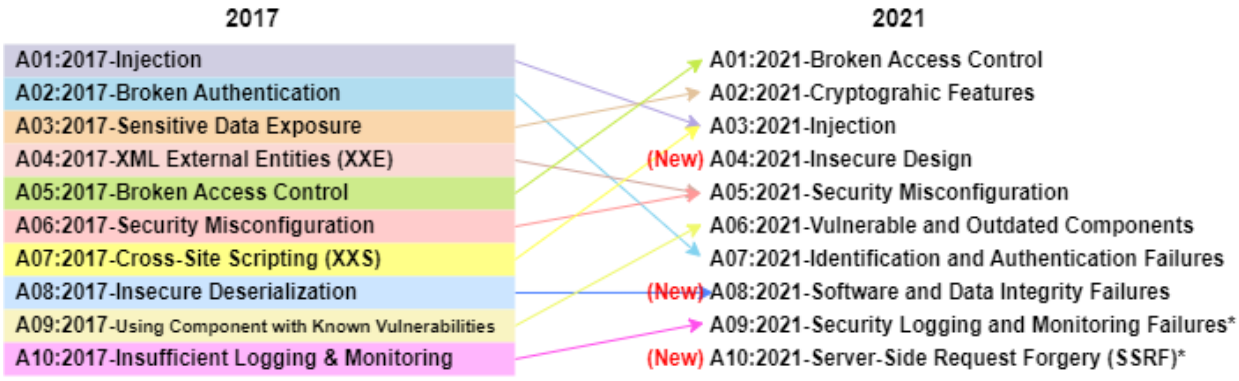

Figure 7: Latest OWASP Top 10 Vulnerability Trend (OWASP, 2021)

Apart from the aforementioned vulnerabilities, (Lee & Kim, 2020) therefore, classified other vulnerability taxonomy according to the developers' intentions such as:

A) Intentional
B) Inadvertent

A) **Intentional Vulnerabilities:** These vulnerabilities are grouped as ***malicious*** **and** ***non-malicious.***
Malicious vulnerabilities signify weaknesses that are deliberately created, and they include:

i. **Trapdoor:** as a result of KBA's flexibility, the system may contain codes (source) that allow illegal access to the system, which may occur at both framework and application level (Babkin & Epishkina, 2018; Lee & Kim, 2020; K. Patel, 2019). An example of such is when a user installs an application comprising malicious codes which are capable of redirecting to an undesirable website.
ii. **Conspirator:** this is a result of a KBA system made up of components that conspire through the exchange of activities to exploit the functionalities of the system or gain access to sensitive materials (Lee & Kim, 2020). For instance, an application component with access to contact information could deliberately give out information such as photos, messages, and contact numbers over the internet.
iii. **Logic bomb:** Lee & Kim, (2020) categorizes this vulnerability as a malicious source code that is designed to disrupt the entire system.

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

***Non-Malicious Vulnerabilities*:** these vulnerabilities are often not recognized by the developers but are known to be the consequences of some features that were intentionally added to the KBA systems (Lee & Kim, 2020). However, they are grouped as intentional as a result of the way they were fashioned into the system as important system requirements. An example is a functional usability requirement incorporated without considering the security requirement, which could lead to the system being vulnerable. These include:

i. **Covert Channels**: these are grouped into intentional and non-malicious because they are not a result of bugs in the system's implementation but rather an implication of flaws in the system's design (Lee & Kim, 2020). That is, they are not maliciously designed and this could be a result of storage manipulation or time modulation in resource sharing for various authentication sessions.
ii. **Open Event Channel:** occurs when a component of a KBA system deliberately exposes its communication channel to communicate with other components, which could thus be exploited by malicious components, thereby exposing the channels to attacks (Lee & Kim, 2020).

B) **Unintentional Vulnerabilities (IVs)**: IVs are purely software bugs, and testing can find and fix these errors. When the system is implemented, some defects could go undetected and cause damage. They consist of:

i. **Inadequate Concurrency (IC):** These are frequently called event anomalies in which malicious components send fake (spoofed) events to corrupt the victim's memory location. As an illustration, consider concurrently sending two separate components that have access to the same memory region as the component of the target, with no assurance that both will be processed.
ii. **Unsafe Event (UE):** This results from an authentication event that contains sensitive information but is inadequately protected (Lee & Kim, 2020; Ling et al., 2017). For instance, an attacker can conveniently intercept or eavesdrop on an event with sensitive information that is not adequately protected or encrypted.
iii. **Unsafe Interface (UI):** this can happen when the KBA interface cannot filter or handle received authentication events or information (Lee & Kim, 2020). This could as a result, expose the system's component to spoofing, thereby causing the system's malfunction.
iv. **Inadequate Configuration:** Lee & Kim, (2020) classified this flaw as a system configuration vulnerability that occurs as a result of a KBA system not completely verifying a particular component or claimant before authorizing access. This may, however, allow unsafe or unauthorized access between components.
v. **Inadequate Resource Management:** this flaw occurs as a result of distributed clusters of different components being used at the same time in an already complex security environment (Lee & Kim, 2020). This could lead to complex resource management. Also, it could happen when a KBA system allocates a resource to a component in an untimely manner. This could lead to a Denial-of-Service (DoS) attack.

Others include:

i. **Broken Authentication:** Patel, (2019) claimed that authentication and session management are important parts of web application security, and vulnerabilities in this aspect can lead to the exposure of users' credentials. This could cause serious damage to the authentication system, such as hijacking administration or user accounts, privacy violations, etc. As also described by the open web application security project, (OWASP, 2021) claimed that incorrectly implemented authentication could pose a huge security risk, that is, if an attacker succeeds, he may be able to have access to legitimate users' identities. Ahmed & A. Hikal, (2019) and Goswami et al., (2014) reiterated that configuration errors and insecure access control practices are difficult to detect using automated processes, but penetrating testing is said to have the capability of detecting missing authentication. In contrast, other methods should be used to determine configuration problems. However, multifactor authentication was also

SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

suggested to mitigate this vulnerability. This can be achieved by implementing scans to detect and remove errors before codes are deployed.

ii. **Query Language Injection (SQL):** OWASP, (2021) listed SQL as a type of injection vulnerability. Injection happens when adversaries exploit insecure codes. This technique allows codes into an application program via data input by the client to the application without proper filtering of malicious script characters. However, because of the inability of the application program to differentiate codes inserted from its legitimate codes, attackers tend to exploit this vulnerability by accessing confidential information as if they are trusted users. A successful SQL exploit can read, delete, update, and modify sensitive data from the database or shut down the whole database management system (DBMS). It is, therefore, recommended that application security testing can expose injection vulnerabilities (Ahmed & A. Hikal, 2019; Awoleye et al., 2014).

iii. **Cross-Site Scripting:** According to Moen et al., (2007) & OWASP, (2021), attackers take advantage of APIs to send or retrieve information from an application with cross-site scripting. This enables the adversaries to hijack user accounts, spread worms and Trojan horses, gain access to users' browsing histories, and then control browsers remotely. Studies have shown that training developers such as in data encryption and validation could mitigate this flaw.

iv. **Using Components with Known Vulnerabilities:** This flaw occurs when an attacker can exploit unsecured third-party APIs and their components, irrespective of how to secure one's codes are. Although there are APIs that allow developers to gain access to third-party services like Google maps and are known to be great time savers and secure, while some rely on insecurity transmission techniques (OWASP, 2021; K. Patel, 2019). However, a static analysis, coupled with a software composition analysis, is said to mitigate such flaws. This enables developers to find susceptible components in their programs before they publish.

v. **Insufficient Logging and Monitoring:** Failure to report/log errors or attacks and poor monitoring practices could expose authentication systems to threats and attacks. Attacks are said to depend on a lack of monitoring and slow remedy time to be able to lunch attacks before they get noticed. To mitigate these flaws, it is expedient to ensure that all access control, login, and server-side validation failures are logged to point out suspicious activities. Research suggested penetration testing to identify areas of the system's application with inadequate logging and also adopting efficient monitoring practices (Ahmed & A. Hikal, 2019; Pate & Jalgaon, 2018; K. Patel, 2019).

vi. **Unencrypted Password:** The password is one of the KBA security measures used in online applications. Without the correct encryption, sensitive and important data can be intercepted by hackers. Because of this, it is difficult to intercept crucial information as it is exchanged between two parties when data is encrypted before transmission (Alhakami & Alhrbi, 2020; Awoleye et al., 2014; Bonneau et al., 2015; Towhidi & Masrom, 2009).

Figure 8 depicts the summary of the taxonomy for this study.

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

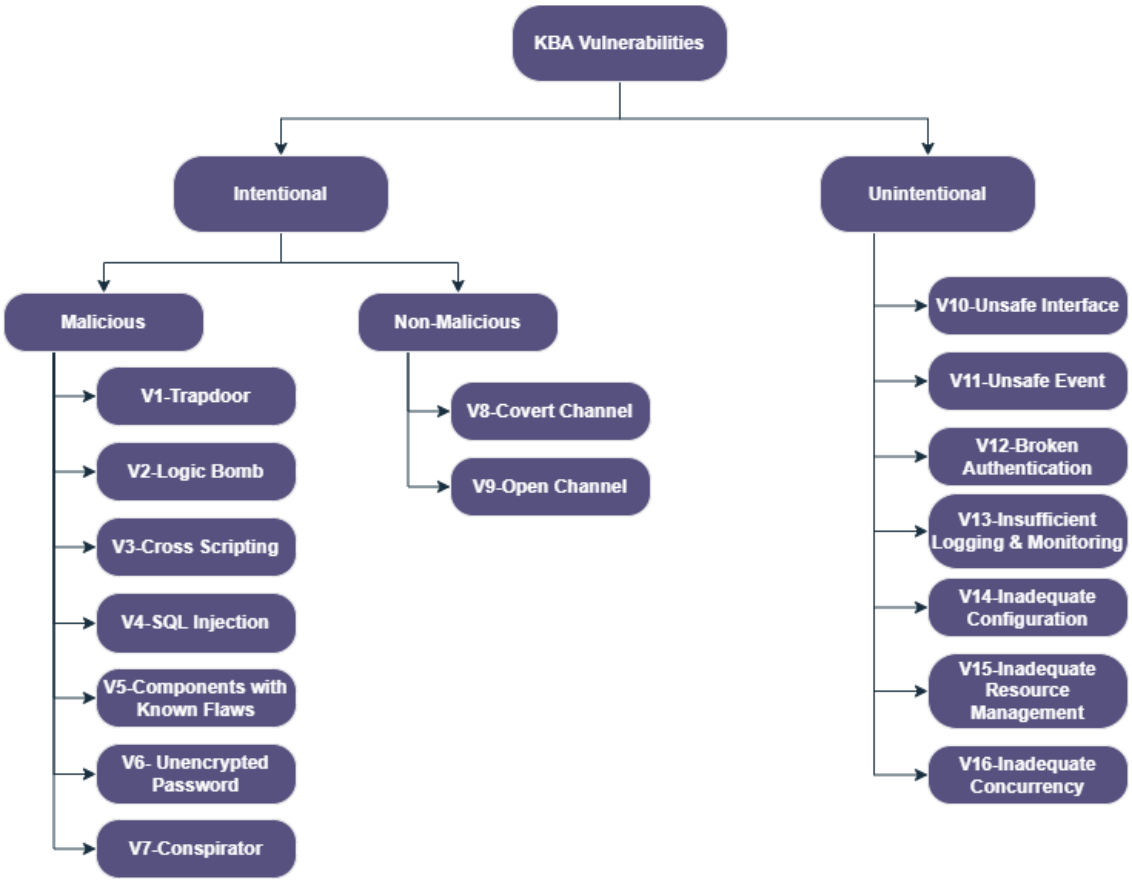

Figure 8: Vulnerability Taxonomy for KBA Systems

3.3.2 Common KBA Attacks

According to the existing research, the vulnerabilities mentioned above can be exploited by the following types of attacks. Developers or end users may experience the following attacks, which have been recognized by KBA systems and techniques:

1. ***Brute force attack (A1)****:* textual and graphical passwords are susceptible to this attack, also known as a guessing or exhaustive search attack. The attack uses probabilistic techniques, meaning that there is a greater than zero chance that the outcome will be the same as expected when an object is chosen randomly from a group of objects. Towhidi & Masrom, (2009) also described Brute force as a way for an attacker to use trial and error techniques to explore a target user's all possible passwords. This attack has two versions which are ***dictionary-based attack (A11***) and ***rainbow table password attack (RTPA (A12))*** (Lee & Kim, 2020; Pate & Jalgaon, 2018). In a ***dictionary-based attack,*** the attacker develops a possible dictionary of textual or graphical passwords and further tries to compromise a target's account using his or her username and passwords in the created dictionary. In textual passwords, the fraudster achieves his/her aim by creating a memorable dictionary of words, like date of birth as passwords. However, Towhidi & Masrom, (2009) affirmed that free wordlists or software tools could be used to automate dictionary attacks. On the other hand, in graphical passwords, the adversary creates a dictionary of popular images that could attract legitimate users as passwords (Shah & Shah, 2013; Towhidi & Masrom, 2009; Xiaoyuan et al., 2005). It has also been found that there exist several algorithms, such as bottom-up and top-down visual attention, that display the visual attention of users. This causes the attention of targets to shift towards hotspots such as shapes that are recognizable, pictures, bright colors, and pictures that are likely to be chosen by target users (Saranya et al., 2017; Verma, 2012). Finally, the attacker thereby uses software to check the login page for a username when a dictionary is created and then

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

tries a possible corresponding password from hundreds of passwords in the dictionary. Also, the other version of brute force attack is the ***rainbow table password attack,*** this attack is perpetrated by an attacker calculating the value of saved hashed passwords. This is done by pre-computing the hashes of millions of passwords and then storing them in a rainbow table, to find the correct password to a particular username (Awoleye et al., 2014; Jin, 2015; Towhidi & Masrom, 2009), and this is an attack common with one-time passwords. The rainbow table's goal is to employ reduction and hashing functions. A hash function is used to convert plaintexts into hashes, and a reduction function shortens the length of the hash function to a predetermined number. Although the table is notorious for taking a long time and using a lot of space to create data, once the attacker obtains the table, it speeds up attacks by quickly cracking a lot of passwords (Alhakami & Alhrbi, 2020; Kim et al., 2020). To mitigate this attack, password management policies should be adopted, and these policies ensure users change their passwords more often. Past studies have also proposed CAPTCHA mechanisms to verify whether the claimed passwords presented are actually from human users or BOTs/automated programs. The CAPTCHA application is a challenge-response program that generates a test to determine whether a third party is a human or a computer, making it simple for people but challenging for computers to get around (Alomar et al., 2017; Tellini & Vargas, 2017). Also, to prevent RTPA, it's been recommended that a random character called ***salt*** should be added before password hashing, why because, whenever a salt key is long, compromising the password would be much more difficult for the attacker (Bulling et al., 2012; Haque & Imam, 2014; Hassan et al., 2020). Every user's salt value is stored in the database, enabling the server to produce a fresh challenge for each login. As a result, the rainbow table either includes all salt combinations, making it impossible for the attacker to process it when seeking the right salt for each of his hashing operations.

2. ***Sniffing Attack (A2):*** this attack is known to exploit vulnerabilities found in the design of applications to expose users' information. The input and output data are monitored and eavesdropped on using a sniffer to carry out this attack (Babkin & Epishkina, 2018). To be able to collect codes, the attacker also configures his or her Network Interface Card (NIC) in promiscuous mode using sniffer software (Haque & Imam, 2014). Similarly, to this, an attacker can sniff the user ID of a recognition-based authentication system like an image-based system, which can subsequently be used to attack the image gallery. This attack can be averted by encrypting sensitive information properly. Another way is by using IP Security Protocol (IPSec) which secures data at the network layer through authentication and encryption of each packet during communication (Alomar et al., 2017; Hassan et al., 2020; Wu et al., 2021).

3. ***Spoofing Attack (A3):*** this attack technique could be in two forms, identity spoofing or content spoofing. To deceive the user, an attacker uses these techniques to masquerade or fake his/her identity or message as a legitimate one. For instance, in content spoofing, the attacker changes the content of a legitimate page to display his/her fake message. This can be found when an attacker changes an authorized account number to his/her own on a bank portal. Likewise, in identity spoofing, the attacker .impersonate a legitimate user (Awoleye et al., 2014; Haque & Imam, 2014; Hassan et al., 2020; Kim et al., 2012; Lu et al., 2021).

    3.1 In a **man-in-the-middle attack (A31),** the attacker starts sniffing packets or modifies messages from the source and then sends the fake message to the destination using the spoofing method to intercept (between the source and destination or the client and the server). While the attacker was actually in charge of the conversation, the two legitimate parties believed they were speaking directly to each other (Hassan et al., 2020; Towhidi & Masrom, 2009). Denial of service (DoS) attacks and DNS poisoning could result from this attack (Wu et al., 2021). Therefore, any passwords that are based on text or graphics can be seen by an attacker, especially when information or data is transferred using the TCP protocol without encryption.

    3.2 ***On the other hand, a phishing attack (A32)*** is an act where a fraudster pretends to be authorized to steal a user's confidential information (Aravindhan & Karthiga, 2013; Jakobsson & Ratkiewicz, 2006; Wu et al., 2021). An example is when a fake website is designed by an attacker for a particular bank portal and then tries to convince a large random number of legitimate bank users. This is done

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

by sending out spam emails for users to visit the cracked website. Phishing can also be achieved through social engineering, which will be discussed below. Furthermore, phishing is achieved in graphical passwords by creating a spoofed (fake) login page and then simulating the login area for drawing passwords. Once the user's password is drawn, the sketch can be used on the original website (Aravindhan & Karthiga, 2013). This attack can be prevented by limiting the IP addresses that send information to only an identified list of IP addresses (Aravindhan & Karthiga, 2013; Jakobsson & Ratkiewicz, 2006; Wu et al., 2021).

4. ***Shoulder Surfing Attack (A4):*** As the name implies, (Bošnjak & Brumen, 2020; Bulling et al., 2012; Ghiyamipour, 2021; Haque & Imam, 2014; Pate & Jalgaon, 2018) confirmed that passwords could be discovered by spying over a person's shoulder. This attack is common in crowded areas, where people queue behind one another, for example, in cyber cafes or ATM points. There are also instances where cameras placed near ATMs are used to record inputted ***personal identification numbers*** (PINs) by customers. However, the best way to prevent your passwords or PINs from being remembered or recorded by attackers is to shield the keypad adequately (Alomar et al., 2017; Hassan et al., 2020).

5. ***Social Engineering Attack (A5):*** This attack is considered one of the oldest attacks that involve technical and psychological ways of tricking target users into providing confidential information (Haque & Imam, 2014). This can be achieved when an attacker successfully lures a target user to describe his/her password over the phone or via email (Alomar et al., 2017). Although this attack is somehow harder in graphical passwords, why because it is always difficult to verbalize a click point, color, or shape (Ghiyamipour, 2021; Verma, 2012). However, mitigation of this attack, according to (Towhidi & Masrom, 2009), because it relates to neither bugs nor system vulnerabilities, but it could be mitigated through user awareness, training, and putting in place security policies (Alhakami & Alhrbi, 2020).

6. ***Guessing Attack (A6):*** The reliance on KBA techniques like passwords and shared secrets and challenge response questions and answers for verification of identities are, no doubt, susceptible to guessing attacks (Alhakami & Alhrbi, 2020; Alomar et al., 2017; Awoleye et al., 2014; Jin, 2015; Polakis et al., 2014; Schechter et al., 2009). Guessing attack occurs when an attacker tries to guess passwords which could be based on either name of pets, family names or date of birth, and so on, by manipulating the input of one or more data. It has also been confirmed that information that is publicly accessible on social networking sites is useful for guessing attacks (Alomar et al., 2017; Jagtap et al., 2019; Javed et al., 2014). Nonetheless, several research works have assessed the capacity of intrusion to guess or suggested ways to make guessing more difficult. Additionally, there are ideas to lessen the likelihood that an attacker will guess the responses to challenge questions in social KBA systems (Alomar et al., 2017; Goswami et al., 2014; Javed et al., 2014; Polakis et al., 2014).

7. ***Authentication Exploitation (A7)****:* According to Towhidi & Masrom, (2009), this attack uses two strategies. They are the exploitation of session variables, such as IDs and other login credentials, and authentication bypass (A71) (A72). Bypassing authentication occurs when an attacker uses the assistance of a user or authorized person to gain access to resources or a service. An attacker can strengthen this by first physically accessing a target computer and changing the settings to grant administrator capabilities. Bypassing all authentication steps, this privilege allows access to files and folders (Awoleye et al., 2014). Another scenario is when an attacker bypasses an authentication page of a financial system by directly typing the URL of the page, which as a result, brings the next page after the authentication page (Aravindhan & Karthiga, 2013)**.** While the exploitation of session variables, also called session side hijacking (Goswami et al., 2014; Kim et al., 2020) occurs when an attacker controls the communication channels of two endpoints by assuming the identity of a legitimate user after the legit user has established a new session and successfully authenticated (He et al., 2020)**.** In addition, the attacker can install a malicious code called Trojan horse on the target system to monitor and record passwords, be they textual or graphical, and user names. However, if the transfer of information is in Secure socket layer (SSL), it would be hard for the fraudster to intercept. Hence, the attacker captures the information using **a *replay attack***. In this attack, the intruder hijacks a valid session ID or password

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

and reuses it, especially when a legit user uses an encrypted password. Therefore, static authentication techniques are vulnerable to replay attacks, and the best way to mitigate this attack is to adopt dynamic authentication techniques (Albayram & Khan, 2016; Aljohani et al., 2016).

The categories of attacks described above can be represented in Figure 9 below.

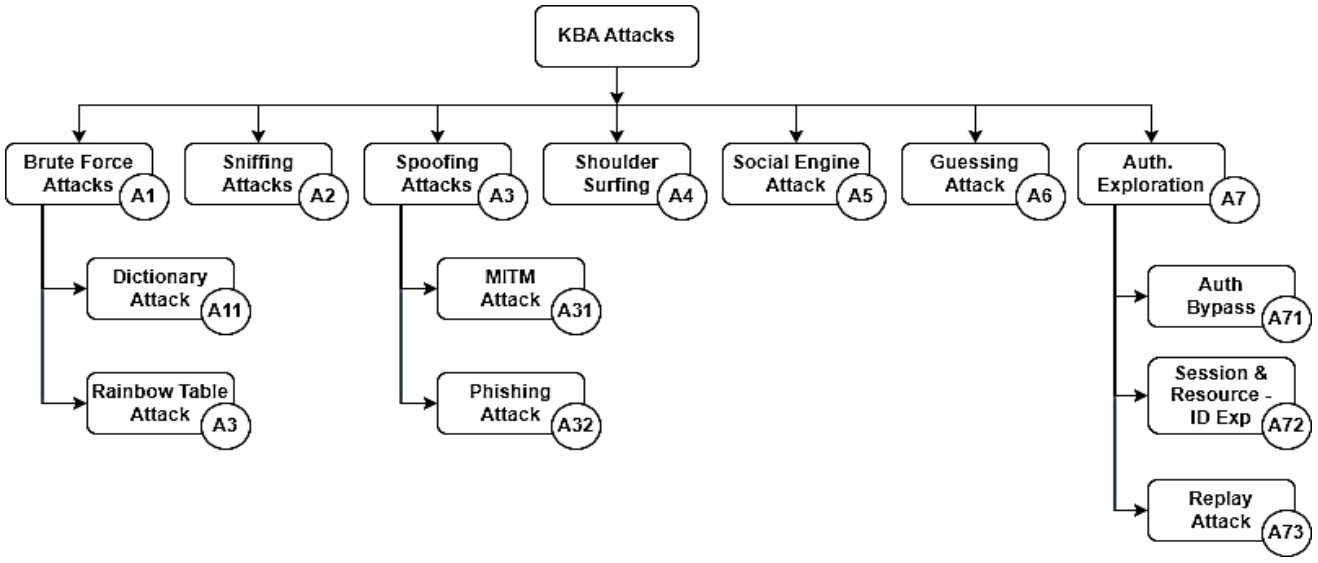

Figure 9: Taxonomy of KBA Attacks

4. Vulnerability Assessment and Penetration Testing (VAPT)

V***ulnerability Assessment*** is the process of investigating security loopholes, weaknesses, and inadequacies in a computing infrastructure, web, or network applications (Khan & Parkinson, 2018; Kim et al., 2020; K. Patel, 2019). It is also a process of identifying, defining, and classifying vulnerabilities in computer systems, web applications, and network infrastructures. This process helps organizations, researchers, and security analysts to swiftly eliminate security issues before adversaries can exploit these flaws for either malicious purposes or monetary gains (Ahmed & A. Hikal, 2019; Freimanis, 2019).

4.1 Categories of Vulnerability Assessment (VA)

The following are the different categories of Vulnerability Assess;

a. **Manual VA:** This is conducted by a human expert who relies on specific instructions or studies to discover security flaws. It, however, requires effort, time, and resources. Also, heavily dependent on the knowledge of the expert (Alomar et al., 2017; Awoleye et al., 2014; Bošnjak & Brumen, 2020; Khan & Parkinson, 2018; K. Patel, 2019; Wu et al., 2021).
b. **Assistive VA:** This is performed with the aid of scanning tools or an up-to-date framework that looks for the most important security flaws. However, the category has some limitations, rendering it static and outdated. They include a lack of adaptability and compatibility issues, routine tool maintenance, and occasionally failure to provide the necessary information regarding the breadth and depth of tests about the system's security (Ahmed & A. Hikal, 2019).
c. **Fully Automated VA:** lastly, we have the most popular, fully automated VA, which makes use of artificial intelligence methods to generate expert-like decisions in the absence of human assistance. It has been considered the most preferred evaluation method because it reduces time and cost for end users. However, it requires additional studies to improve automated knowledge and representation that are needed to develop prevention techniques capable of securing computing infrastructures (Alhaidary &

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Rahman, 2016; Alhakami & Alhrbi, 2020; Freimanis, 2019; Katsini et al., 2016; K. Patel, 2019; Wu et al., 2021).

4.2 Penetration Testing (PT)

It is a known fact that cyber-attacks have increased drastically but sadly, many organizations do not still actively monitor the security of their infrastructures, so they remain prey to attackers. Vulnerability Assessment on the other hand, is often carried out using penetration testing techniques to identify vulnerabilities in an attempt to attack the system, find vulnerabilities, and mitigate them (Ahmed & A. Hikal, 2019; Awoleye et al., 2014; Khan & Parkinson, 2018). PT is often called ethical hacking, red teaming, or pen testing used to discover vulnerabilities (Freimanis, 2019). Freimanis claimed that PT is usually carried out by security experts and can be done in many ways: ***black-box testing, grey-box testing,*** and ***white-box testing.*** In black-box testing, the testers do not have any prior knowledge of the authentication system, while in grey-box testing, the tester is allowed to have partial knowledge of the system, but in the latter, the tester is given full knowledge of the system. PT is carried out in multiple phases, which will be discussed in the later part of the report.

The goal of PT is to find out the level of security and then enhance the security level of the systems being tested afterward.

In (K. Patel, 2019), the author combined VA and PT to make the VAPT process, see Figure 10. While (Freimanis, 2019) discussed extensively the seven essential steps involved in penetration testing. They are:

i. Pre-Engagement Interactions
ii. Intelligence Gathering
iii. Threat Modeling
iv. Vulnerability Analysis
v. Exploitation
vi. Post exploitation
vii. Reporting

***Pre-Engagement Interactions*:** In every PT is important to define the test's scope and set clear boundaries with the client/tester. That is, the attacker must be informed of the type of attacks, necessary credentials, and so on (Ahmed & A. Hikal, 2019; K. Patel, 2019).

***Intelligence Gathering*:** This step is for gathering the necessary information. It is the most important step in PT because when the tester understands the complete system on a high level, it enables him or her to understand the vulnerabilities in the system. This step usually starts with port scanning by using tools like Nmap to identify all open hosts and ports.

Threat Modeling: This is the next phase after completing the previous steps. At this stage, the attacker should have a good knowledge of the system and can now direct attacks, such as the ones previously discussed in the former part of the report, on the parts of the system that are most critical.

Vulnerability Analysis: the analysis stage occurs when the hosts and the threats/attacks have been enumerated; it is now the time to look for possible flaws. This can be done manually by an experienced tester or by using automated tools to scan huge databases for known vulnerabilities.

Exploitation: This step is where an attacker or tester penetrates the security of the system by exploiting a flaw. The exploitation is to gain root privileges on the target system; however, other successful attacks could also be the crashing of servers and filtration of documents or credentials, which could pave the way for other attacks. This can be achieved with weak authentications and SQL injections using tools like Metasploit (Al-Ahmad et al., 2019; Genge et al., 2015).

Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Post exploitation: As discussed earlier, the main purpose of PT is to gain root privileges on the target system, that is, having the liberty to do anything, such as having access to more credentials by using known vulnerabilities to be able to gain access to other systems. This could be done by reusing already existing credentials.

Reporting: The last but not least stage of PT, is to report the findings to the client. This is usually done by generating a report that informs the client about the security situation of the system and recommending actions to be taken.

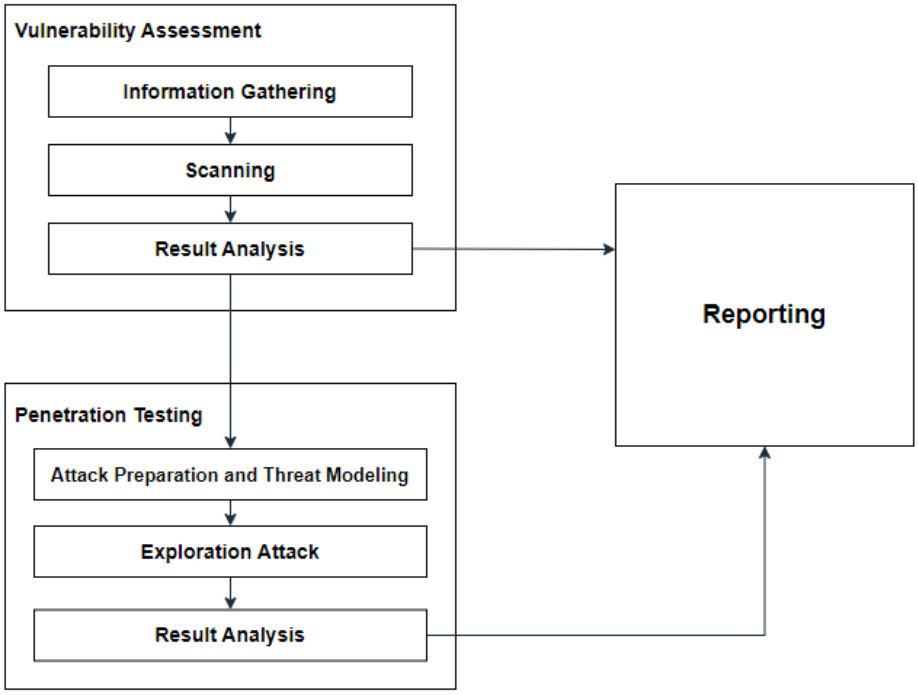

Figure 10: VAPT Process Diagram (K. Patel, 2019)

4.3 Vulnerability Assessment Tools

There are many tools available and suitable for vulnerability assessment and ethical hacking techniques. Each tool has unique functionalities, but the most impactful and suitable for KBA are presented below.

NMAP: Network Mapper, an open source, is a cross-platform tool that can work with any operating system. It is a freely available tool to scan live hosts on a network and services running on the target system. Zenmap is the graphical interface for Nmap, which can be used for a better overview.

Metasploit: It is an open-source exploitation framework for known vulnerabilities, with both free and commercial versions, owned by Rapid7. This is an advanced tool and also the most popular specifically for penetration testing, according to (K. Patel, 2019).

Mimikatz: This is a tool by Benjamin Delpy written in C language to extract and manipulate users' credentials in windows. It requires local administrative privileges to read from it. However, running Mimikatz on a server is not advisable as an anti-virus could detect it. But it could be run remotely, together with other tools like Metasploit, without having to write code to the disk.

Bloodhound: It is a scanning and mapping tool for querying three things on the network: administrators on the target network, membership of users and groups, and user sessions, that is, who is logged in and where Data/information can therefore be retrieved the network without the administrative permissions, by creating a graph to calculate the shortest paths to escalate privileges on the target system/network (Freimanis, 2019; K. Patel, 2019).

SLUJST
Sule Lamido University Journal of Science & Technology Vol. 8 No. 1 [June, 2024], pp. 34-61
https://doi.org/10.56471/slujst.v7i.485
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Nessus: This is a commercial vulnerability assessment tool that organizations widely use for the scanning of network systems and web applications (K. Patel, 2019). Nessus produces a full report that is detailed with respect to user-defined policies. Also, research has it that Nessus updates its security plugins regularly.

Burp Suite: This is also a tool used for web application testing that is available both as freeware and commercial versions. It is used to carry out brute-force attacks by intercepting and manipulating ongoing requests and responses.

Accuentix: A commercial tool like Nessus is used to identify vulnerabilities in web applications, available for both Windows and Linux operating systems. Its key feature is to support a single-page application.

Harvester: The harvester is used for gathering information in the vulnerability evaluation phase. It collects information such as hostnames, usernames, email-ids, sub-domains, and so on. Harvester is a very efficient tool for phishing in email security.

Beef: An acronym for Browser Exploitation Framework. It is a freeware tool used for penetration testing based on browser vulnerabilities. It helps in identifying zombie browsers in real time. This tool aids in providing commands and interfaces to control an individual as well as a group of victims.

Zed *Attack Proxy* (ZAP): A tool developed by OWASP, created to scan web applications. It's an inbuilt operating system and also freeware like Kali and Parrot (OWASP, 2021; K. Patel, 2019). Its features include scanning, a spider to crawl all the web application pages, and proxy intercepting.

SQLMAP: This is an open-source tool for penetration testing to identify and detect flaws in SQL databases. The tool is so effective that it can provide all the information stored in an SQL database. Also, a tool is available for all operating systems. A Summary of Vulnerability Case Studies is given in Table 5.

Table 5: Summary of Vulnerability Case Studies

References	Vulnerability	Attack	Method of Assessment	Mitigation
(Babkin & Epishkina, 2018; Lee & Kim, 2020; K. Patel, 2019)	V1-V2	A1-A7		
(Al-Khurafi & Al-Ahmad, 2016; Awoleye et al., 2014; Pranathi et al., 2018; Taha & Karabatak, 2018)	V3	A3; A7; A71; A72	Penetration Testing Systematic Literature Review (SLR)	Encoding; validation, user-awareness
(Abirami et al., 2016; Ahmed & A. Hikal, 2019; Al-Khurafi & Al-Ahmad, 2016; Awoleye et al., 2014; K. Patel, 2019)	V4	A2; A3; A31; A32; A7	Vulnerability Assessment & Penetration Testing (VAPT)	Data Encryption &Validation; Training.
(Ahmed & A. Hikal, 2019; Goswami et al., 2014; Khan & Parkinson, 2018; Kim et al., 2012; OWASP, 2013, 2017, 2021; Pate & Jalgaon, 2018; K. Patel, 2019)	V5	A1-A7	Penetration Testing SLR	Software Analysis Encryption; validation; Testing; Multifactor Authentication.
(Alhakami & Alhrbi, 2020; Awoleye et al., 2014; Bonneau et al., 2015; Lee & Kim, 2020; K. Patel, 2019; Towhidi & Masrom, 2009)	V6-V7	A1; A2; A3; A4; A6	VAPT SLR	Encryption CAPTCHA; SALTING

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

(Lee & Kim, 2020; Olawoyin et al., 2020)	V8	-	Penetration Testing	User awareness/training
(Goswami et al., 2014; Lee & Kim, 2020; Taha & Karabatak, 2018)	V9	A1-A7	VAPT	Encryption; Policy Enforcement
	V10&V11	A1-A7	VAPT	Access Control; Encryption; Policy Enforcement; Testing
(Ahmed & A. Hikal, 2019; Al-Khurafi & Al-Ahmad, 2016; Awoleye et al., 2014; Goswami et al., 2014; Lee & Kim, 2020; K. Patel, 2019; Pranathi et al., 2018; Shinde & Ardhapurkar, 2016)	V12	A3; A6; A7	VAPT	DKBA; Multifactor Authentication
(Ahmed & A. Hikal, 2019; Lee & Kim, 2020; Pate & Jalgaon, 2018; K. Patel, 2019; Pranathi et al., 2018; Shinde & Ardhapurkar, 2016; Wu et al., 2021)	V13-V16	A1-A7	VAPT	Detection of Anomalies; Logging& Effective monitoring Practices.

5. Findings

While vulnerability assessment of existing authentication methods is good and essential to aid comprehensive evaluation and comparison among authentication techniques, the systematic literature review observed that the level of penetration and attacks on web applications across the trending vulnerabilities identified are high. Over the past years, it has also been observed that the most prominent web application vulnerabilities are SQL injection, broken authentication & session management, and cross-site scripting (XSS). However, in the last four years, things changed, according to the latest release of OWASP Top 10, 2021. It has been revealed that the Broken Access Control vulnerability moved from the fifth position to the first position #1 after 94% of applications tested were mapped to Broken Access Control. Also, Cryptographic Failures, better known as Sensitive Data Exposure, the shift from third to second position #2. Here, cryptography failures are again in the spotlight because they frequently expose private information. The fact that SQL injection, which had the second-most occurrences in applications, dropped to third place #3 is the most intriguing finding, though. Cross-site scripting (XSS) has been incorporated into this category, though. When 90% of the applications tested were mapped to security misconfiguration, it rose from sixth place in the previous edition to position #5, and XML External Entities are now included in this category. Due to the assistance and increased availability of standardized frameworks, the vulnerability for broken authentication, now known as identification and authentication failures, is now falling from its previous second position. Knowing Server-Side Request Forgery shows a low incidence rate, while security logging and monitoring, previously insufficient logging and monitoring is not well represented. Finally, new categories of vulnerabilities, such as Insecure Design, and Software & Data Integrity Failures, have also been introduced.

6. Conclusion

The security of any web application depends on a variety of authentication techniques. A serious vulnerability in this area could lead to the theft of an administrator's or user's account, a breach of privacy, or the loss of sensitive data to intruders. Thus, it has been discovered that vulnerability assessment helps give organizations remediation process security. Additionally, it helps to perform several security checks with automation tools quickly. However, vulnerability analysis has several drawbacks, such as the inability to recognize logical attack paths. In other words, automated tools can only produce analysis following the standards established by the testers. Policies that are poorly defined may produce extraneous data that contain false positives. Yet, it is important to

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

remember that, just as a thorough analysis of the experimental designs employed in the works of literature can substantially impact the outcome, so too do experimental vulnerability assessment designs. This will help to clarify how choices can impact the analysis and inferences drawn from such investigations.

Acknowledgement

The authors acknowledged the anonymous reviewers for their insightful comments which help to improve the quality of the paper.

References

Abirami, J., Devakunchari, R., & Valliyammai, C. (2016). A top web security vulnerability SQL injection attack - Survey. In *ICoAC 2015 - 7th International Conference on Advanced Computing*. IEEE. https://doi.org/10.1109/ICoAC.2015.7562806

Adamu, H., Mohammed, A. D., Adepoju, S. A., & Aderiike, A. O. (2022). A Three-Step One-Time Password, Textual and Recall-Based Graphical Password for an Online Authentication. *Proceedings of the 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development, NIGERCON 2022*, 1–5. https://doi.org/10.1109/NIGERCON54645.2022.9803122

Adeniran, T. C., Jimoh, R. G., Abah, E. U., Faruk, N., & Alozie, E. (2023). Evaluation of an Enhanced Dynamic Knowledge-Based Authentication System (EDKBA) in the Era of Social Media. *2023 2nd International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)*, *1*, 1–5. https://doi.org/10.1109/ICMEAS58693.2023.10379352

Ahmed, S. F., & A. Hikal, N. (2019). A Review of Cyber-security Measuring and Assessment Methods for Modern Enterprises. *JOIV : International Journal on Informatics Visualization*, *3*(3). https://doi.org/10.30630/joiv.3.3.241

Akula, S., & Devisetty, V. (2004). Image based registration and authentication system. *Proceedings of Midwest Instruction and …*, *4*. http://www.micsymposium.org/mics_2004/Akula.pdf

Al-Ahmad, A. S., Kahtan, H., Hujainah, F., & Jalab, H. A. (2019). Systematic Literature Review on Penetration Testing for Mobile Cloud Computing Applications. *IEEE Access*, *7*, 173524–173540. https://doi.org/10.1109/ACCESS.2019.2956770

Al-Khurafi, O. B., & Al-Ahmad, M. A. (2016). Survey of Web Application Vulnerability Attacks. *Proceedings - 2015 4th International Conference on Advanced Computer Science Applications and Technologies, ACSAT 2015*, 154–158. https://doi.org/10.1109/ACSAT.2015.46

Alaidaros, H., & Albeedh, S. (2022). Towards Studying the Relationship between Job Satisfaction and Organizations' Information Security. *International Conference on Intelligent Technology, System and Service for Internet of Everything, ITSS-IoE 2022*, 1–6. https://doi.org/10.1109/ITSS-IoE56359.2022.9990932

Albayram, Y., & Khan, M. M. H. (2016). Evaluating smartphone-based dynamic security questions for fallback authentication: a field study. *Human-Centric Computing and Information Sciences*, *6*(1), 16. https://doi.org/10.1186/s13673-016-0072-3

Alhaidary, M., & Rahman, S. M. M. (2016). Security vulnerability analysis and corresponding mitigation for password-based authentication using an offline personal authentication device. *Proceedings of the 2016 IEEE International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET 2016*, 842–849. https://doi.org/10.1109/WiSPNET.2016.7566251

Alhakami, H., & Alhrbi, S. (2020). Knowledge based authentication techniques and challenges. *International Journal of Advanced Computer Science and Applications*, *11*(2), 727–732. https://doi.org/10.14569/ijacsa.2020.0110291

Aljohani, N., Shelton, J., Roy, K., & Esterline, A. (2016). Robust password system based on dynamic factors. *Proceedings of the 6th International Conference on Information Communication and Management, ICICM 2016*, 30–34. https://doi.org/10.1109/INFOCOMAN.2016.7784210

Alomar, N., Alsaleh, M., & Alarifi, A. (2017). Social Authentication Applications, Attacks, Defense Strategies and Future Research Directions: A Systematic Review. *IEEE Communications Surveys and Tutorials*, *19*(2),

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

1080–1111. https://doi.org/10.1109/COMST.2017.2651741

Alozie, E., Imoize, A. L., Olagunju, H. I., & Faruk, N. (2023). Introduction to emerging security and privacy schemes for dense 6G wireless communication networks. In *Security and Privacy Schemes for Dense 6G Wireless Communication Networks* (pp. 1–29). Institution of Engineering and Technology. https://doi.org/10.1049/PBSE021E_ch1

Alsaiari, H., Papadaki, M., Dowland, P., & Furnell, S. (2016). Graphical One-Time Password (GOTPass): A usability evaluation. *Information Security Journal*, *25*(1–3), 94–108. https://doi.org/10.1080/19393555.2016.1179374

Alsaleem, B. O., & Alshoshan, A. I. (2021). Multi-Factor Authentication to Systems Login. *Proceedings - 2021 IEEE 4th National Computing Colleges Conference, NCCC 2021*, 1–4. https://doi.org/10.1109/NCCC49330.2021.9428806

Aravindhan, K., & Karthiga, R. R. (2013). One-time Password: A Survey. *International Journal of Emerging Trends in Engineering and Development Issue 3*, *1*(3), 613–623.

Asha, H. P., & Jingle, I. D. J. (2022). One Time Password-Based Two Channel Authentication Mechanism Using Blockchain. In *Lecture Notes in Networks and Systems* (Vol. 462, pp. 229–237). Springer. https://doi.org/10.1007/978-981-19-2211-4_20

Awoleye, O. M., Ojuloge, B., & Ilori, M. O. (2014). Web application vulnerability assessment and policy direction towards a secure smart government. *Government Information Quarterly*, *31*(S1), S118–S125. https://doi.org/10.1016/j.giq.2014.01.012

Babkin, S., & Epishkina, A. (2018). One-Time Passwords: Resistance to Masquerade Attack. *Procedia Computer Science*, *145*, 199–203. https://doi.org/10.1016/j.procs.2018.11.040

Basharzad, S. N., & Fazeli, M. (2017). Knowledge based dynamic password. *2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI)*, *2018-Janua*, 0367–0372. https://doi.org/10.1109/KBEI.2017.8325004

Bonneau, J., Bursztein, E., Caron, I., Jackson, R., & Williamson, M. (2015). Secrets, lies, and account recovery: Lessons from the use of personal knowledge questions at google. *WWW 2015 - Proceedings of the 24th International Conference on World Wide Web*, 141–150. https://doi.org/10.1145/2736277.2741691

Bošnjak, L., & Brumen, B. (2020). Shoulder surfing experiments: A systematic literature review. *Computers and Security*, *99*, 102023. https://doi.org/10.1016/j.cose.2020.102023

Bulling, A., Alt, F., & Schmidt, A. (2012). Increasing the security of gaze-based cued-recall graphical passwords using saliency masks. *Conference on Human Factors in Computing Systems - Proceedings*, 3011–3020. https://doi.org/10.1145/2207676.2208712

Düzgün, R., Noah, N., Mayer, P., Das, S., & Volkamer, M. (2022). SoK: A Systematic Literature Review of Knowledge-Based Authentication on Augmented Reality Head-Mounted Displays. *ACM International Conference Proceeding Series*, 1–12. https://doi.org/10.1145/3538969.3539011

Erdem, E., & Sandikkaya, M. T. (2018). OTPaaS-One time password as a service. *IEEE Transactions on Information Forensics and Security*, *14*(3), 743–756. https://doi.org/10.1109/TIFS.2018.2866025

Freimanis, D. (2019). Vulnerability Assessment of Authentication Methods in a Large-Scale Computer System. *Degree Project Computer Science and Engineering*.

Genge, B., Graur, F., & Enăchescu, C. (2015). Non-intrusive Techniques for Vulnerability Assessment of Services in Distributed Systems. *Procedia Technology*, *19*, 12–19. https://doi.org/10.1016/j.protcy.2015.02.003

Ghiyamipour, F. (2021). Secure graphical password based on cued click points using fuzzy logic. *Security and Privacy*, *4*(2). https://doi.org/10.1002/spy2.140

Gong, L., Pan, J., Liu, B., & Zhao, S. (2013). A novel one-time password mutual authentication scheme on sharing renewed finite random sub-passwords. *Journal of Computer and System Sciences*, *79*(1), 122–130. https://doi.org/10.1016/j.jcss.2012.06.002

Goswami, S., Misra, S., & Mukesh, M. (2014). A Replay Attack Resilient System for PKI Based Authentication in Challenge-Response Mode for Online Application. *Proceedings - 2014 3rd International Conference on Eco-Friendly Computing and Communication Systems, ICECCS 2014*, 144–148. https://doi.org/10.1109/Eco-friendly.2014.104

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Guan, A., & Chen, C. M. (2022). A Novel Verification Scheme to Resist Online Password Guessing Attacks. *IEEE Transactions on Dependable and Secure Computing*, *19*(6), 4285–4293. https://doi.org/10.1109/TDSC.2022.3174576

Hameed, A. S. S. S. (2015). Color Based Authentication Scheme for Publically Disclosable Entities. *International Journal of Engineering Research and Applications*, *5*(1), 23–25. https://www.ijera.com/papers/Vol5_issue1/Part - 6/E501062325.pdf

Haque, A., & Imam, B. (2014). A New Graphical Password : Combination of Recall & Recognition Based Approach. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, *8*(2).

Hassan, M. A., Shukur, Z., & Hasan, M. K. (2020). An Improved Time-Based One Time Password Authentication Framework for Electronic Payments. *International Journal of Advanced Computer Science and Applications*, *11*(11), 359–366. https://doi.org/10.14569/IJACSA.2020.0111146

He, J., Han, D., & Li, K. C. (2020). On one-time cookies protocol based on one-time password. *Soft Computing*, *24*(8), 5657–5670. https://doi.org/10.1007/s00500-019-04138-5

Jagtap, A., Sarwade, A., Jhalawat, H., & Chaudhari, M. R. V. (2019). Secret Questions based Authentication System for Android Smartphone. *International Journal on Recent and Innovation Trends in Computing and Communication*, *7*(4), 01–03. https://doi.org/10.17762/ijritcc.v7i4.5275

Jakobsson, M., & Ratkiewicz, J. (2006). Designing ethical phishing experiments: A study of (ROT13) rOnl query features. In *Proceedings of the 15th International Conference on World Wide Web* (pp. 513–522). ACM. https://doi.org/10.1145/1135777.1135853

Javed, A., Bletgen, D., Kohlar, F., Durmuth, M., & Schwenk, J. (2014). Secure fallback authentication and the trusted friend attack. *Proceedings - International Conference on Distributed Computing Systems*, *30-June-20*, 22–28. https://doi.org/10.1109/ICDCSW.2014.30

Jin, L. (2015). *Threats & Vulnerabilities in Online Social Networks*. http://www.sis.pitt.edu/jjoshi/courses/IS2621/Spring15/SP_SN.pdf

Just, M., & Aspinall, D. (2009). Personal choice and challenge questions: A security and usability assessment. In *SOUPS 2009 - Proceedings of the 5th Symposium On Usable Privacy and Security*. ACM. https://doi.org/10.1145/1572532.1572543

Kadham, N. R., & Sreenivasa Ravi, K. (2018). A lightweight One Time Password (OTP) based smart learning in Internet of Things. *International Journal of Engineering and Technology(UAE)*, *7*, 480–483. https://doi.org/10.14419/ijet.v7i2.7.10867

Katsini, C., Belk, M., Fidas, C., Avouris, N., & Samaras, G. (2016). Security and usability in knowledge-based user authentication: A review. *ACM International Conference Proceeding Series*, 0–5. https://doi.org/10.1145/3003733.3003764

Kavianpour, S., Shanmugam, B., Azam, S., Zamani, M., Narayana Samy, G., & De Boer, F. (2019). A Systematic Literature Review of Authentication in Internet of Things for Heterogeneous Devices. *Journal of Computer Networks and Communications*, *2019*, 1–14. https://doi.org/10.1155/2019/5747136

Khan, S., & Parkinson, S. (2018). *Review into State of the Art of Vulnerability Assessment using Artificial Intelligence*. Springer International Publishing. https://doi.org/10.1007/978-3-319-92624-7_1

Khurana, M., Aggarwal, R., Rani, R., & Khurana, V. (2022). Understanding password vulnerabilities - A mathematical approach. *AIP Conference Proceedings*, *2357*(1), 100012. https://doi.org/10.1063/5.0080658

Kim, H., Han, J., Park, C., & Yi, O. (2020). Analysis of vulnerabilities that can occur when generating one-time password. *Applied Sciences (Switzerland)*, *10*(8). https://doi.org/10.3390/APP10082961

Kim, H., Tang, J., & Anderson, R. (2012). Social authentication: Harder than it looks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7397 LNCS*, 1–15. https://doi.org/10.1007/978-3-642-32946-3_1

Lee, Y. K., & Kim, D. (2020). A Taxonomy for Security Flaws in Event-Based Systems. *Applied Sciences*, *10*(20), 7338. https://doi.org/10.3390/app10207338

Ling, C. H., Lee, C. C., Yang, C. C., & Hwang, M. S. (2017). A secure and efficient one-time password authentication scheme for WSN. *International Journal of Network Security*, *19*(2), 177–181.

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

https://doi.org/10.6633/IJNS.201703.19(2).02

Lu, Y., Yu, K., & Lv, X. (2021). Image encryption with one-time password mechanism and pseudo-features. *Multimedia Tools and Applications*, *80*(10), 15041–15055. https://doi.org/10.1007/s11042-021-10522-x

Maifi, M., Khan, H., Kentros, S., Nguyen, N., & Jiang, R. (2015). *Designing challenge questions for location-based authentication systems : a real-life study*. *March 2016*. https://doi.org/10.1186/s13673-015-0032-3

Mamachan, M. D. B. R. (2019). Session Authentication Using Color Schemes and Images. *International Journal of Science and Research (IJSR)*, *8*(3), 1796–1799.

Mell, P., Scarfone, K., & Romanosky, S. (2006). Common vulnerability scoring system. *IEEE Security and Privacy*, *4*(6), 85–89. https://doi.org/10.1109/MSP.2006.145

Moen, V., Klingsheim, A. N., Simonsen, K. I. F., & Hole, K. J. (2007). Vulnerabilities in e-governments. *International Journal of Electronic Security and Digital Forensics*, *1*(1), 89. https://doi.org/10.1504/IJESDF.2007.013595

Mohan, S., & Harun, N. Z. (2022). Jeev Time : Secure Authentication Using Integrated Face Recognition in Social Media Applications. *JOURNAL OF SOFT COMPUTING AND DATA MINING*, *3*(2), 19–30.

O, V. B. (2015). *Authentication Scheme for Passwords using Color and Text*. *3*(3), 316–323.

Okamoto, M. (2013). Knowledge-Based Authentication using Twitter Can We Use Lunch Menus As Passwords? *International Journal of Network Security & Its Applications*, *5*(5), 1–10. https://doi.org/10.5121/ijnsa.2013.5501

Olawoyin, L. A., Faruk, N., Oloyede, A. A., Popoola, S. I., Adeniran, T. C., Surajudeen-Bakinde, N. T., & Abdulkarim, A. (2020). Secure transmission in an unfriendly environment. *ICT Express*, *6*(2), 104–108. https://doi.org/10.1016/j.icte.2019.09.001

OWASP. (2013). *OWASP Top Ten Vulnerabilities 2013*.

OWASP. (2017). 2017 Top 10. *The OWASP Foundation*, 4. https://owasp.org/www-project-top-ten/2017/Top_10

OWASP. (2021). OWASP Top 10 2021. In *Synopsys* (p. 1). https://www.synopsys.com/glossary/what-is-owasp-top-10.html

Park, C. S. (2018). One-time password based on hash chain without shared secret and re-registration. *Computers and Security*, *75*, 138–146. https://doi.org/10.1016/j.cose.2018.02.010

Parmar, V., Sanghvi, H. A., Patel, R. H., & Pandya, A. S. (2022). A Comprehensive Study on Passwordless Authentication. *International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2022 - Proceedings*, 1266–1275. https://doi.org/10.1109/ICSCDS53736.2022.9760934

Pate, S. E., & Jalgaon, D. (2018). A survey of Possible Attacks on Text & Graphical Password Authentication Techniques. *International Journal of Scientific Research in Computer Science and Engineering*, *6*(1), 77–80. https://www.isroset.org/spl_pub_paper/IJSRCSE/16-SEPATE-IJSRCSE.pdf

Patel, B., Sarwar, A., & Chavan, S. (2021). Graphical Password Authentication Using Colour Login Technique. *International Research Journal of Engineering and Technology*, 3650–3653. www.irjet.net

Patel, K. (2019). A survey on vulnerability assessment penetration testing for secure communication. *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, *Icoei*, 320–325. https://doi.org/10.1109/ICOEI.2019.8862767

Plata, I. T., & Calpito, J. L. (2020). Application of time-based one time password (TOTP) algorithm for human resource e-leave tracking web app. *International Journal of Scientific and Technology Research*, *9*(3), 4070–4077.

Polakis, I., Ilia, P., Maggi, F., Lancini, M., Kontaxis, G., Zanero, S., Ioannidis, S., & Keromytis, A. D. (2014). Faces in the distorting mirror: Revisiting photo-based social authentication. *Proceedings of the ACM Conference on Computer and Communications Security*, 501–512. https://doi.org/10.1145/2660267.2660317

Pranathi, K., Kranthi, S., Srisaila, A., & Madhavilatha, P. (2018). Attacks on Web Application Caused by Cross Site Scripting. *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology, ICECA 2018*, *Iceca*, 1754–1759. https://doi.org/10.1109/ICECA.2018.8474765

Rabkin, A. (2008). Personal knowledge questions for fallback authentication: Security questions in the era of Facebook. In *SOUPS 2008 - Proceedings of the 4th Symposium on Usable Privacy and Security* (pp. 13–23). ACM. https://doi.org/10.1145/1408664.1408667

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Rajesh, N., S, S., Varshini, V., Bhavani N, B., & Pradeep, D. (2015). Color Code Based Authentication And Encryption. *International Journal of Advanced Research in Computer and Communication Engineering*, *4*(5), 403–405. https://doi.org/10.17148/IJARCCE.2015.4588

Renaud, K., & Just, M. (2010). Pictures or questions? Examining user responses to association-based authentication. *Proceedings of the 2010 British Computer Society Conference on Human-Computer Interaction, BCS-HCI 2010*, 98–107. https://doi.org/10.14236/ewic/hci2010.14

Rice, A. (Facebook S. (2012). Social Authentication. *CanSecWest 2012*. https://securelist.com/cansecwest-lets-talk-about-non-targeted-attacks/32212/

Sadikan, S. F. N., Ramli, A. A., & Fudzee, M. F. M. (2019). A survey paper on keystroke dynamics authentication for current applications. *AIP Conference Proceedings*, *2173*(1), 20010. https://doi.org/10.1063/1.5133925

Sani, S. I. A., Alhassan, J. K., & Mohammed, A. S. (2022). Graphical Based Authentication Method Combined with City Block Distance for Electronic Payment System. In S. Misra & C. Arumugam (Eds.), *Lecture Notes on Data Engineering and Communications Technologies* (Vol. 109, pp. 289–323). Springer International Publishing. https://doi.org/10.1007/978-3-030-93453-8_13

Saranya, P., Sharavanan, S., Vijai, R., & Balajee, R. M. (2017). Authentication scheme for session passwords using color and image. *International Journal on Smart Sensing and Intelligent Systems*, *2017*(Specialissue), 590–603. https://doi.org/10.21307/ijssis-2017-272

Schechter, S., Brush, A. J. B., & Egelman, S. (2009). It's no secret Measuring the security and reliability of authentication via "secret" questions. *Proceedings - IEEE Symposium on Security and Privacy*, 375–390. https://doi.org/10.1109/SP.2009.11

Shah, P., & Shah, P. R. (2013). Image Based Authentication System Image based Authentication System General Terms. *International Journal of Computer Applications*, 975–8887. https://www.researchgate.net/publication/310236116

Shinde, P. S., & Ardhapurkar, S. B. (2016). Cyber security analysis using vulnerability assessment and penetration testing. *IEEE WCTFTR 2016 - Proceedings of 2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare*. https://doi.org/10.1109/STARTUP.2016.7583912

Sikiru, I. A., Olawoyin, L. A., Faruk, N., Oloyede, A. A., Abdulkarim, A., Olayinka, I. Y., Sowande, O. A., Garba, S., & Imoize, A. L. (2023). Physical layer security using boundary technique for emerging wireless communication systems. *Security and Privacy*, *6*(3), e288. https://doi.org/10.1002/spy2.288

Taha, T. A., & Karabatak, M. (2018). A proposed approach for preventing cross-site scripting. *6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding*, *2018-Janua*, 1–4. https://doi.org/10.1109/ISDFS.2018.8355356

Tellini, N., & Vargas, F. (2017). *Two-Factor Authentication: Selecting and implementing a two-factor authentication method for a digital assessment platform*. https://www.diva-portal.org/smash/get/diva2:1104740/FULLTEXT01.pdf

Towhidi, F., & Masrom, M. (2009). A Survey on Recognition Based Graphical User Authentication Algorithms. *ArXiv Preprint ArXiv*, *6*(2). http://arxiv.org/abs/0912.0942

Tran, T. (2022). *Security or Usability: A literature study about strategies for how users choose passwords*. https://www.diva-portal.org/smash/get/diva2:1682924/FULLTEXT01.pdf

Ullah, A., Xiao, H., & Barker, T. (2019). A study into the usability and security implications of text and image based challenge questions in the context of online examination. *Education and Information Technologies*, *24*(1), 13–39. https://doi.org/10.1007/s10639-018-9758-7

Verma, P. (2012). icAuth. *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*, 329–330. https://doi.org/10.1145/2166966.2167037

Wakili, A., Bakkali, S., & Faruk, N. (2023). AI-enhanced context-aware optimization of RPL routing protocol for IoT environments. *2023 2nd International Conference on Multidisciplinary Engineering and Applied Science, ICMEAS 2023*, *1*, 1–6. https://doi.org/10.1109/ICMEAS58693.2023.10379229

Wu, L., Cai, H. J., & Li, H. (2021). SGX-UAM: A secure unified access management scheme with one time passwords via intel SGX. *IEEE Access*, *9*, 38029–38042. https://doi.org/10.1109/ACCESS.2021.3063770

Xiaoyuan, S., Ying, Z., & Owen, G. S. (2005). Graphical passwords: A survey. In *Proceedings - Annual Computer*

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Security Applications Conference, ACSAC (Vol. 2005, pp. 463–472). IEEE. https://doi.org/10.1109/CSAC.2005.27

Zangooei, T., Mansoori, M., & Welch, I. (2012). A hybrid recognition and recall based approach in graphical passwords. *Proceedings of the 24th Australian Computer-Human Interaction Conference, OzCHI 2012*, *November*, 665–673. https://doi.org/10.1145/2414536.2414637

Zhao, D., & Luo, W. (2017). One-time password authentication scheme based on the negative database. *Engineering Applications of Artificial Intelligence*, *62*, 396–404. https://doi.org/10.1016/j.engappai.2016.11.009

Zulkarnain, S., Idrus, S., Cherrier, E., Rosenberger, C., Zulkarnain, S., Idrus, S., Cherrier, E., Rosenberger, C., Zulkarnain, S., Idrus, S., Cherrier, E., & Rosenberger, C. (2013). A Review on Authentication Methods To cite this version : *Australian Journal of Basic and Applied Sciences*, *7*(5), 95–107.