iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 Issue: IX Month of publication: September 2024

DOI: https://doi.org/10.22214/ijraset.2024.64033

Defending Against Modern Web Attacks: Strategies for Safeguarding User Identities and Data

Nithin Varam
Palo Alto Networks, USA

STAY SECURE ONLINE
Defending Against Modern Web Attacks:
Learn how to protect your online presence effectively.

Abstract: Organizations must adopt a comprehensive approach to safeguard user identities and sensitive data in the face of increasingly sophisticated web-based attacks. This article presents a multi-faceted strategy for defending against modern web threats, addressing key aspects such as phishing prevention, secure authentication mechanisms, protection against cross-site scripting (XSS) and cross-site request forgery (CSRF) attacks, browser and endpoint security, web application security best practices, and the implementation of a Zero Trust security model. By examining the current landscape of web-based threats and providing actionable recommendations, this article aims to empower organizations with the knowledge and tools necessary to combat evolving cyber threats effectively. The proposed defensive measures encompass user awareness and education, robust technical controls, secure coding practices, and adopting technologies such as multi-factor authentication, biometric authentication, web application firewalls, and browser isolation techniques. Additionally, the article emphasizes the importance of a proactive approach to incident response and threat intelligence, enabling organizations to swiftly detect, contain, and mitigate web-based attacks. By implementing these strategies, organizations can significantly enhance their resilience against modern web threats, safeguarding user identities, protecting sensitive data, and maintaining a secure online presence in an increasingly hostile digital environment.

Keywords: Web-based attacks, Phishing Prevention, Secure authentication, Cross-site scripting (XSS), Zero Trust security

I. INTRODUCTION

In the digital age, web-based attacks have become increasingly sophisticated and prevalent, posing significant threats to user identities and sensitive data [1].

As organizations and individuals rely heavily on web technologies for various aspects of their lives, the need for effective strategies to defend against modern web attacks has never been more critical [2]. Cybercriminals employ a wide range of techniques, including phishing, cross-site scripting (XSS), cross-site request forgery (CSRF), and drive-by downloads, to exploit vulnerabilities and gain unauthorized access to systems and information [3].

The impact of successful web attacks can be severe, leading to identity theft, financial losses, data breaches, and reputational damage [4]. According to recent studies, the global cost of cybercrime is projected to reach $10.5 trillion annually by 2025 [5]. Furthermore, the COVID-19 pandemic has accelerated the shift towards remote work and online activities, expanding the attack surface and creating new opportunities for cybercriminals [6].

Organizations must adopt a multi-layered approach to web security to address these challenges, encompassing user awareness, secure authentication mechanisms, robust coding practices, and proactive threat intelligence [7]. By implementing effective strategies and leveraging advanced technologies, organizations can significantly reduce the risk of falling victim to modern web attacks and safeguard the integrity of user identities and data [8].

This article aims to provide a comprehensive overview of the current landscape of web-based threats and present actionable strategies for defending against them. It will delve into various aspects of web security, including phishing prevention, secure authentication, protection against XSS and CSRF attacks, browser and endpoint security, web application security best practices, Zero Trust web access, incident response, and user education. By examining these topics in detail and providing practical recommendations, this article seeks to empower organizations and individuals with the knowledge and tools necessary to combat modern web attacks effectively and maintain a secure online presence.

II. MODERN WEB ATTACK VECTORS

Web-based attacks have evolved significantly, with attackers employing various sophisticated techniques to compromise user identities and gain unauthorized access to sensitive data. Among the most prevalent web attack vectors are phishing, cross-site scripting (XSS), cross-site request forgery (CSRF), and drive-by downloads [9].

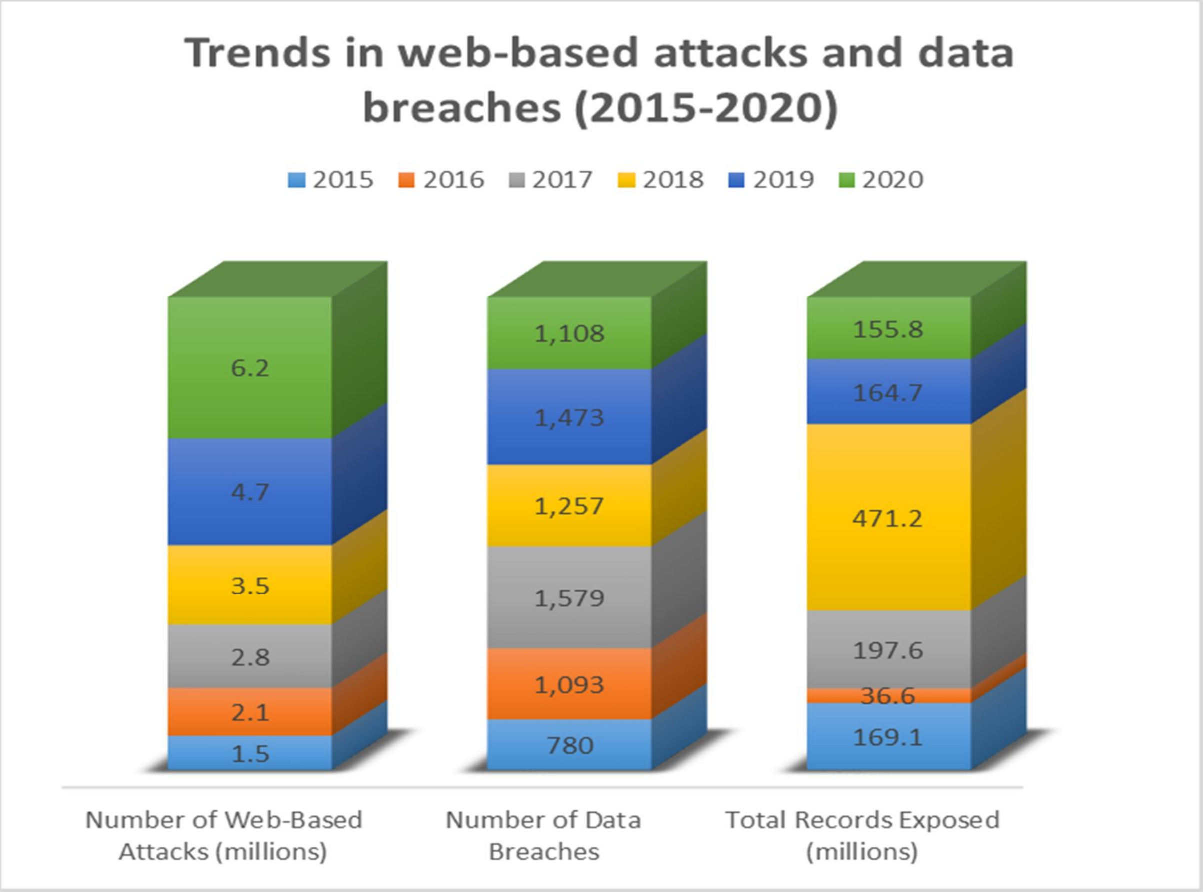

Figure 1: Trends in web-based attacks and data breaches (2015-2020) [75]

A. Phishing attacks

Phishing attacks are a common and effective technique cybercriminals use to trick users into disclosing sensitive information, such as login credentials or financial data [10]. These attacks typically involve sending fraudulent emails or messages that appear to come from legitimate sources, luring victims into clicking on malicious links or providing personal information [11]. Phishing attacks have become increasingly targeted and personalized, with attackers using social engineering tactics to enhance their credibility and success rates.Recent research has highlighted the increasing sophistication of phishing attacks, with cybercriminals employing advanced techniques such as clone and spear phishing. Clone phishing involves replicating legitimate websites or emails with minor modifications to deceive users, while spear phishing targets specific individuals or organizations with highly personalized and convincing messages[12].

B. Cross-Site Scripting (XSS) attacks

Cross-site scripting (XSS) attacks exploit vulnerabilities in web applications to inject malicious scripts into web pages viewed by other users [13]. These scripts can steal user session tokens, manipulate website content, or redirect users to malicious websites [14]. XSS attacks can be classified into three main categories: reflected XSS, stored XSS, and DOM-based XSS [15]. Reflected XSS occurs when malicious scripts are reflected off a web application onto the victim's browser, while stored XSS involves the persistent storage of malicious scripts on the target server [16]. DOM-based XSS exploits vulnerabilities in client-side scripts and does not require the malicious payload to be sent to the server [17].

C. Cross-Site Request Forgery (CSRF) attacks

Cross-site request forgery (CSRF) attacks trick authenticated users into performing unintended actions on a web application [18]. In a CSRF attack, the attacker crafts a malicious link or script that sends a forged HTTP request to the target website, leveraging the user's authenticated session [19]. If the user is logged in and the website lacks proper CSRF protection, the attacker can perform unauthorized actions on behalf of the user, such as changing account settings or initiating financial transactions [20].

D. Drive-by downloads and malvertising

Drive-by downloads occur when users unknowingly download malicious software onto their devices by visiting compromised websites or clicking malicious links [21]. These attacks often exploit vulnerabilities in web browsers, browser plugins, or operating systems to install malware without user interaction [22] silently. Malvertising, or malicious advertising, is a technique used to spread drive-by downloads by injecting malicious code into legitimate online advertising networks. When users view or click on infected ads, they are redirected to websites that host exploit kits, which scan for vulnerabilities and deliver malware payloads.

E. Related-Domain Attacks

Related-domain attackers control a sibling domain of their target web application, e.g., as the result of a subdomain takeover [23, 24]. These attackers can acquire capabilities through different attack vectors and abuse them to compromise web application security by focusing on different angles, including cookies, CSP, CORS, postMessage, and domain relaxation [23, 24]. A large-scale security measurement on the top 50k domains from the Tranco list led to discovering vulnerabilities in 887 sites, quantifying the threats related-domain attackers pose to popular web applications [23, 24].

III. PHISHING PREVENTION AND USER AWARENESS

Phishing attacks remain a significant threat to user security, and effective prevention strategies must focus on both technological solutions and user awareness. By combining robust email filters, anti-phishing tools, and user education programs, organizations can significantly reduce the risk of falling victim to phishing attacks [25].

A. Strategies for identifying phishing attempts

Users play a critical role in preventing phishing attacks by being able to identify suspicious emails and messages. Visual cues, such as spelling and grammatical errors, generic greetings, and a sense of urgency, can often indicate a potential phishing attempt [26]. However, with advances in Generative AI, attackers can draft phishing emails without grammatical errors. Additionally, users should be cautious of emails requesting sensitive information or containing suspicious attachments or links [27]. Domain and URL analysis, such as checking for slight variations in legitimate domain names or the presence of URL shorteners, can also help users identify phishing attempts [28].

B. User education and training programs

User education and training programs are essential for creating an organization's security-aware culture. These programs should teach users about the latest phishing techniques, best practices for avoiding phishing traps, and the importance of reporting suspicious emails [29]. Regular training sessions, simulated phishing exercises, and interactive learning modules can help reinforce security concepts and keep users updated on emerging threats [30]. Organizations should also establish clear policies and procedures for handling phishing incidents and encourage open communication between employees and security teams [31]. Organizations should implement contextual, just-in-time training that provides users with relevant information at the moment of potential risk. Gamification techniques and simulated phishing exercises can increase engagement and retention of security concepts.

C. Technological solutions for phishing prevention

In addition to user awareness, technological solutions play a crucial role in preventing phishing attacks. Email filters and anti-phishing tools can analyze incoming messages for known phishing indicators, such as suspicious sender addresses, malicious links, or attachments, and automatically quarantine or block potential threats [32]. These tools often utilize machine learning algorithms and threat intelligence feeds to adapt to evolving phishing techniques [33]. Browser extensions and plugins can also provide an additional layer of protection by warning users about suspicious websites or blocking attempts to enter sensitive information on untrusted pages [34]. Implementing easy-to-use reporting mechanisms, such as dedicated email addresses or browser plugins for flagging potential phishing attempts, can significantly enhance an organization's ability to quickly detect and respond to threats.

D. Integration of AI and Machine Learning in Phishing Detection

Artificial intelligence and machine learning technologies are increasingly employed to enhance phishing detection capabilities. These technologies can analyze vast amounts of data to identify patterns and anomalies indicative of phishing attempts, often outperforming traditional rule-based systems. However, organizations must be aware of AI-based systems' potential limitations and biases and implement them as part of a broader security strategy.

E. Cross-organizational Collaboration and Information Sharing

Effective phishing prevention requires collaboration and information sharing among organizations, security researchers, and law enforcement agencies. Participating in threat intelligence sharing platforms and industry-specific information sharing and analysis centers (ISACs) can provide organizations with valuable insights into emerging phishing tactics and enable more proactive defense strategies.

IV. SECURE AUTHENTICATION MECHANISMS

Secure authentication mechanisms are critical for protecting user identities and preventing unauthorized access to sensitive data. By implementing multi-factor authentication (MFA), biometric authentication, and passwordless authentication solutions, organizations can significantly enhance the security of user accounts and reduce the risk of identity theft [35].

Authentication Method	Security Level	User Convenience	Deployment Complexity
Password-based	Low	High	Low
Multi-factor (MFA)	High	Moderate	Moderate
Biometric	High	High	High
Passwordless	High	High	Moderate

Table 1: Comparison of common authentication methods [35]

A. Multi-Factor Authentication (MFA)

Multi-factor authentication (MFA) adds an extra layer of security to the authentication process by requiring users to provide two or more independent factors to verify their identity [36]. These factors typically include something the user knows (e.g., a password), something the user has (e.g., a hardware token or mobile device), and something the user is (e.g., a biometric characteristic) [37]. By combining multiple factors, MFA makes it significantly more difficult for attackers to gain unauthorized access to user accounts, even if one factor is compromised [38]. Common MFA methods include SMS-based one-time passwords (OTPs), authenticator apps, and hardware tokens [39].

B. Biometric Authentication

Biometric authentication leverages unique physical or behavioral characteristics of users to verify their identity [40]. Common biometric factors include fingerprints, facial recognition, iris scans, and voice recognition [41]. Biometric authentication offers several advantages over traditional password-based authentication, such as increased convenience, enhanced security, and resistance to phishing attacks [42]. However, organizations must carefully consider the privacy implications and potential vulnerabilities associated with storing and processing biometric data [43].

C. Passwordless Authentication

Passwordless authentication aims to eliminate the reliance on traditional passwords by using alternative authentication methods, such as biometrics, hardware tokens, or email-based magic links [44]. By removing passwords from the authentication process, organizations can reduce the risk of password-related threats, such as password reuse, weak passwords, and password theft [45]. Passwordless authentication can also improve user experience by streamlining the login process and eliminating the need for users to remember complex passwords [46]. However, organizations must ensure passwordless authentication solutions are properly implemented and secured to prevent potential vulnerabilities [47].

V. PROTECTING AGAINST XSS AND CSRF ATTACKS

Cross-site scripting (XSS) and cross-site request forgery (CSRF) attacks pose significant threats to web application security. To protect against these attacks, organizations must implement secure coding practices, utilize web application firewalls (WAFs), and enforce strict content security policies (CSPs) [48].

A. Secure coding practices

Secure coding practices are essential for preventing XSS and CSRF vulnerabilities in web applications. Developers should thoroughly validate and sanitize all user-supplied input to ensure malicious scripts or characters cannot be injected into web pages [49]. Input validation techniques, such as whitelisting and blacklisting, can help filter out potentially harmful input [50]. Additionally, proper output encoding should render any remaining malicious characters harmless when displayed in the browser. Secure coding frameworks and libraries, such as OWASP ESAPI and Microsoft Anti-XSS, can assist developers in implementing these practices effectively [52].

Research highlights the effectiveness of combining client- and server-side validation techniques [51]. Their study demonstrates that implementing context-aware sanitization can significantly reduce the risk of XSS vulnerabilities.

B. Advanced Detection and Prevention Techniques

Web application firewalls (WAFs) act as a protective barrier between web applications and potential attackers, monitoring and filtering incoming traffic for malicious requests [53]. WAFs can detect and block XSS and CSRF attacks by analyzing request parameters, headers, and payloads for known attack signatures or anomalous behavior [54]. They can also enforce security policies, such as input validation rules and content security policies, to prevent the execution of malicious scripts. WAFs can be deployed as hardware appliances, software solutions, or cloud-based services, providing an additional layer of defense against web-based attacks [56].

Traditional signature-based WAFs have limitations. A study proposes an innovative approach using combinatorial testing to generate attack vectors, demonstrating improved detection rates compared to conventional methods [55].

Furthermore, a study introduces a machine-learning-based approach for detecting malicious requests and classifying attack types. Their research shows promising results in identifying sophisticated XSS and CSRF attacks that may bypass traditional WAFs. [77]

C. Content Security Policies (CSP) and Dynamic Analysis

Content security policies (CSPs) are declarative security mechanisms that allow web application owners to specify which content sources are trusted and can be loaded by the browser [57]. By implementing CSP headers, organizations can restrict the execution of inline scripts, prevent the loading of resources from untrusted sources, and mitigate the impact of XSS attacks [58]. CSPs can be fine-tuned to allow only necessary content sources and block potentially malicious ones, reducing the attack surface of web applications [59]. Modern web browsers support CSP and can enforce these policies to protect users from XSS and other content injection attacks [60].

However, recent research reveals that many deployed CSPs are inadequate regarding directive coverage and secure use. Their study employs clustering techniques to analyze CSP effectiveness, highlighting the need for more comprehensive and properly configured policies [78].

To address the challenges in CSP implementation, a study proposes DiffCSP, a differential testing framework for finding CSP enforcement bugs. Their approach has successfully identified numerous security and functional bugs in browser CSP implementations, emphasizing the importance of continuous testing and improving CSP enforcement [79].

Table 2: Effectiveness of various XSS prevention techniques [48]

Prevention Technique	Reflected XSS	Stored XSS	DOM-based XSS
Input validation	Effective	Partially effective	Not effective
Output encoding	Effective	Effective	Not effective
Content Security Policy (CSP)	Effective	Effective	Partially effective
HTTPOnly cookies	Not effective	Not effective	Effective
Browser-based filters	Partially effective	Partially effective	Not effective

Table 2 summarizes the effectiveness of various XSS prevention techniques against different XSS attacks.

D. Integrated Defense Strategies

An effective defense against XSS and CSRF attacks requires an integrated approach. Research has proposed a comprehensive framework that combines biometric authentication, data splitting techniques, and encryption to enhance security in cloud environments. Their approach demonstrates improved protection against XSS and CSRF attacks in distributed systems. Additionally, recent work by Junaid Latief Shah (2024) highlights the importance of addressing emerging attack vectors. Their research proposes novel techniques for detecting and preventing sophisticated scripting attacks that exploit evolving web technologies.

VI. SECURING WEB BROWSERS AND ENDPOINTS

Securing web browsers and endpoints is crucial for protecting users from web-based attacks, as attackers often target these components first. By implementing browser security best practices, utilizing endpoint detection and response (EDR) solutions, and leveraging browser isolation techniques, organizations can significantly reduce the risk of compromise [61].

A. Browser security settings and configurations

Proper configuration of browser security settings can help mitigate the risk of web-based attacks. Users and organizations should ensure that web browsers are up-to-date with the latest security patches and updates to address known vulnerabilities [62]. Disabling unnecessary browser extensions and plugins can also reduce the attack surface and prevent potential exploitation [63]. Enabling built-in security features, such as pop-up blockers, phishing filters, and automatic security updates, can further enhance browser security [64]. Organizations should also consider implementing browser security policies and guidelines to ensure consistent and secure configuration across all endpoints [65].

Organizations should implement a comprehensive browser security policy that includes:

1) Regular updates and patch management: Ensuring browsers are kept up-to-date with the latest security patches is critical for addressing known vulnerabilities.

2) Extension and plugin management: Disabling unnecessary browser extensions and plugins reduces the attack surface. Organizations should maintain a whitelist of approved extensions and regularly audit installed add-ons.

3) Security feature enablement: Activating built-in security features such as pop-up blockers, phishing filters, and automatic updates enhances overall browser security.

4) Content security policies: Implementing content security policies (CSP) can help prevent cross-site scripting (XSS) attacks and other injection-based vulnerabilities.

5) HTTPS enforcement: Configuring browsers to enforce HTTPS connections and implementing HTTP Strict Transport Security (HSTS) helps protect against man-in-the-middle attacks and SSL stripping.

B. Endpoint Detection and Response (EDR) solutions

Endpoint detection and response (EDR) solutions provide continuous monitoring and threat detection capabilities for endpoints, including web browsers [66]. EDR tools can identify and alert security teams to suspicious activities, such as unauthorized browser extensions, malicious script execution, or abnormal network traffic [67]. By analyzing endpoint behavior and leveraging machine learning algorithms, EDR solutions can detect and respond to advanced web-based threats that may evade traditional security controls [68]. In the event of a compromise, EDR tools can facilitate rapid incident response and containment, minimizing the impact of web-based attacks on the organization [69].

C. Isolation techniques

Browser isolation techniques aim to separate web browsing activities from the underlying endpoint and network, preventing potential threats from reaching sensitive resources [70]. One common approach is to use virtual machines (VMs) or containers to run web browsers in isolated environments, effectively sandboxing them from the host system [71]. If a web-based attack compromises the browser, the isolation layer contains the threat and prevents it from spreading to other network parts [72]. Another approach is to use remote browser isolation, where browsing sessions are rendered on a remote server and only a visual representation is sent to the user's device, further reducing the risk of compromise [73]. Browser isolation can protect against drive-by downloads, malvertising, and other web-based threats that target endpoint vulnerabilities [74]. Also consider:

1) Application-level isolation: Some solutions allow specific browser instances or tabs to run in isolated environments.
2) Network-level isolation: Implementing network segmentation and micro segmentation can isolate browser traffic and limit the potential impact of a compromise.
3) Hardware-based isolation: Leveraging hardware-based virtualization technologies can provide stronger isolation guarantees than software-only solutions.

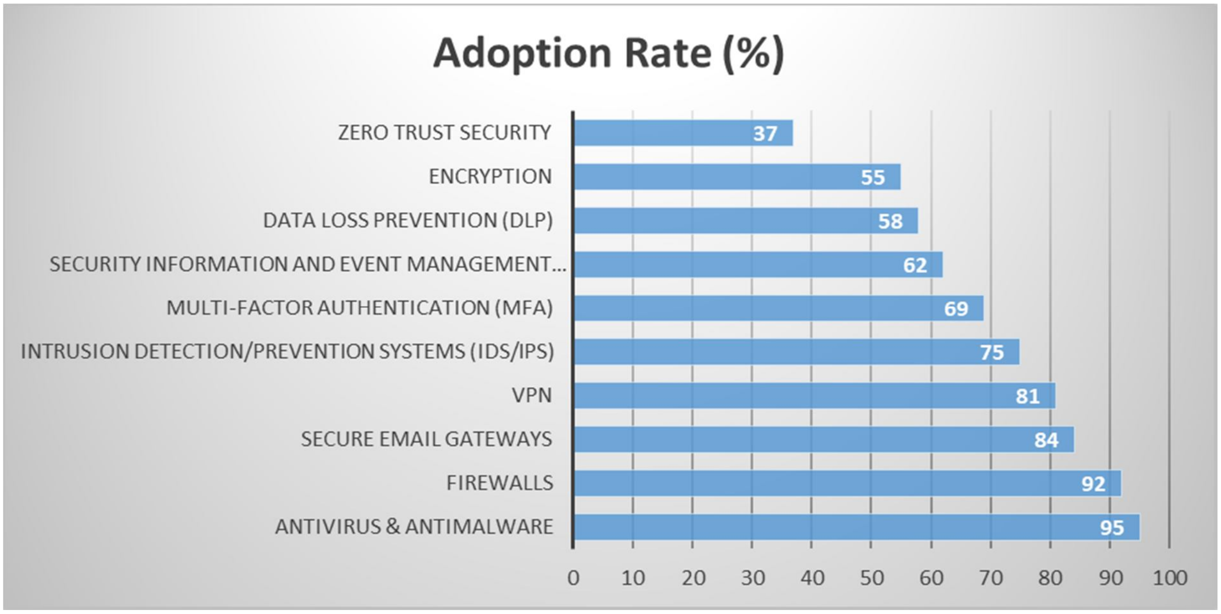

Figure 2: Adoption rates of security measures by organizations (2020) [76]

D. Advanced Browser Security Technologies

Recent research has introduced several advanced technologies for enhancing browser security:

1) Just-in-Time (JIT) hardening: Implementing JIT hardening techniques in browser JavaScript engines can mitigate certain classes of memory corruption vulnerabilities.
2) Site Isolation: Chromium-based browsers have implemented Site Isolation, which renders each site in a separate process, providing stronger protection against side-channel attacks and certain types of vulnerabilities.
3) WebAssembly sandboxing: Improving the sandboxing of WebAssembly modules can enhance protection against potential vulnerabilities in web applications.

4) Browser fingerprinting protection: Implementing techniques to reduce browser fingerprinting can enhance user privacy and make it more difficult for attackers to track and target specific users.

VII. CONCLUSION

Defending against modern web attacks requires a holistic and proactive approach that combines technical controls, secure coding practices, user education, and robust incident response capabilities. Organizations can significantly enhance their resilience against evolving web-based threats by implementing the strategies outlined in this article. A multi-layered defense encompassing secure authentication mechanisms, protection against XSS and CSRF attacks, browser and endpoint security, and adopting a Zero Trust security model provides a comprehensive framework for safeguarding user identities and sensitive data. Regular security audits, penetration testing, and vulnerability management ensure that web applications remain secure and compliant with industry standards. Furthermore, fostering a culture of security awareness through ongoing user education and training programs is crucial for preventing phishing attacks and promoting secure online behavior. As web-based threats evolve, organizations must remain vigilant, adapt their defensive strategies, and leverage the latest technologies and threat intelligence to stay ahead of potential attackers. By prioritizing web security and investing in robust defensive measures, organizations can protect their digital assets, maintain customer trust, and mitigate the risk of costly data breaches and reputational damage in an increasingly complex and challenging online landscape.

REFERENCES

[1] A. K. Sood and R. J. Enbody, "Crimeware-as-a-service—A survey of commoditized crimeware in the underground market," Int. J. Crit. Infrastruct. Prot., vol. 6, no. 1, pp. 28-38, Mar. 2013, doi: 10.1016/j.ijcip.2013.01.002.

[2] S. Gupta, A. Singhal, and A. Kapoor, "A literature survey on social engineering attacks: Phishing attack," in Proc. Int. Conf. Comput., Commun. Autom. (ICCCA), Apr. 2016, pp. 537-540, doi: 10.1109/CCAA.2016.7813778.

[3] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: A literature survey," IEEE Commun. Surveys Tuts., vol. 15, no. 4, pp. 2091-2121, 4th Quart., 2013, doi: 10.1109/SURV.2013.032213.00009.

[4] R. Heartfield and G. Loukas, "A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks," ACM Comput. Surv., vol. 48, no. 3, pp. 1-39, Feb. 2016, doi: 10.1145/2835375.

[5] Cybersecurity Ventures, "Cybercrime to cost the world $10.5 trillion annually by 2025," Press Release, Nov. 2020. [Online]. Available: https://cybersecurityventures.com/cybercrime-damage-costs-10-trillion-by-2025/

[6] K. Lallie, "The impact of COVID-19 on cybersecurity," World Economic Forum, May 2020. [Online]. Available: https://www.weforum.org/agenda/2020/05/covid-19-cybersecurity-disruption/

[7] A. Sedgewick, M. Souppaya, and K. Scarfone, "Guide to application whitelisting," NIST Special Publication 800-167, 2015, doi: 10.6028/NIST.SP.800-167.

[8] Y. Chahid, M. Benabdellah, and A. Azizi, "Internet of things security," in Proc. Int. Conf. Wireless Technol., Embed. Intell. Syst. (WITS), Apr. 2017, pp. 1-6, doi: 10.1109/WITS.2017.7934655.

[9] S. Gupta, A. Singhal, and A. Kapoor, "A literature survey on social engineering attacks: Phishing attack," in Proc. Int. Conf. Comput., Commun. Autom. (ICCCA), Apr. 2016, pp. 537-540, doi: 10.1109/CCAA.2016.7813778.

[10] A. Aleroud and L. Zhou, "Phishing environments, techniques, and countermeasures: A survey," Comput. Secur., vol. 68, pp. 160-196, Jul. 2017, doi: 10.1016/j.cose.2017.04.006.

[11] G. Stringhini and O. Thonnard, "That ain't you: Blocking spearphishing through behavioral modelling," in Proc. 12th Int. Conf. Detection Intrusions Malware, Vulnerability Assessment (DIMVA), Jul. 2015, pp. 78-97, doi: 10.1007/978-3-319-20550-2_5.

[12] https://www.semanticscholar.org/paper/18f2f67975a7f6d05f1c30e42b2a3fd444324eca

[13] P. Vogt, F. Nentwich, N. Jovanovic, E. Kirda, C. Kruegel, and G. Vigna, "Cross site scripting prevention with dynamic data tainting and static analysis," in Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS), Feb. 2007, doi: 10.14722/ndss.2007.23055.

[14] S. Lekies, B. Stock, and M. Johns, "25 million flows later: Large-scale detection of DOM-based XSS," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Nov. 2013, pp. 1193-1204, doi: 10.1145/2508859.2516703.

[15] J. Grossman, R. Hansen, P. D. Petkov, A. Rager, and S. Fogie, "XSS attacks: Cross site scripting exploits and defense," Syngress, 2007, doi: 10.1016/B978-1-59749-154-9.X5000-7.

[16] A. Barth, C. Jackson, and J. C. Mitchell, "Robust defenses for cross-site request forgery," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Oct. 2008, pp. 75-88, doi: 10.1145/1455770.1455782.

[17] S. Gupta and B. B. Gupta, "Cross-site scripting (XSS) attacks and defense mechanisms: Classification and state-of-the-art," Int. J. Syst. Assur. Eng. Manag., vol. 8, no. 1, pp. 512-530, Jan. 2017, doi: 10.1007/s13198-015-0376-0.

[18] X. Li and Y. Xue, "A survey on server-side approaches to securing web applications," ACM Comput. Surv., vol. 46, no. 4, pp. 1-29, Mar. 2014, doi: 10.1145/2541315.

[19] R. Wang, S. Chen, and X. Wang, "Signing me onto your accounts through Facebook and Google: A traffic-guided security study of commercially deployed single-sign-on web services," in Proc. IEEE Symp. Secur. Privacy (SP), May 2012, pp. 365-379, doi: 10.1109/SP.2012.30.

[20] N. Jovanovic, E. Kirda, and C. Kruegel, "Preventing cross site request forgery attacks," in Proc. 2nd IEEE Int. Conf. Secur. Privacy Commun. Netw. (SecureComm), Aug. 2006, pp. 1-10, doi: 10.1109/SECCOMW.2006.359531.

[21] M. Egele, E. Kirda, and C. Kruegel, "Mitigating drive-by download attacks: Challenges and open problems," in Proc. Open Res. Problems Netw. Secur. (iNetSec), Mar. 2009, pp. 52-62, doi: 10.1007/978-3-642-05437-2_5.

[22] M. Cova, C. Kruegel, and G. Vigna, "Detection and analysis of drive-by-download attacks and malicious JavaScript code," in Proc. 19th Int. Conf. World Wide Web (WWW), Apr. 2010, pp. 281-290, doi: 10.1145/1772690.1772720.

[23] https://www.semanticscholar.org/paper/d8ddc9fb26837b917171de576085ad08bd19440e

[24] https://arxiv.org/abs/2012.01946

[25] S. Purkait, "Phishing counter measures and their effectiveness - Literature review," Inf. Manag. Comput. Secur., vol. 20, no. 5, pp. 382-420, 2012, doi: 10.1108/09685221211286548.

[26] [A. Aleroud and L. Zhou, "Phishing environments, techniques, and countermeasures: A survey," Comput. Secur., vol. 68, pp. 160-196, Jul. 2017, doi: 10.1016/j.cose.2017.04.006.

[27] B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, "Fighting against phishing attacks: State of the art and future challenges," Neural Comput. Appl., vol. 28, no. 12, pp. 3629-3654, Dec. 2017, doi: 10.1007/s00521-016-2275-y.

[28] A. K. Jain and B. B. Gupta, "Comparative analysis of features based machine learning approaches for phishing detection," in Proc. 3rd Int. Conf. Comput. Sustain. Glob. Dev. (INDIACom), Mar. 2016, pp. 2125-2130, doi: 10.1109/CSGRC.2017.8038573.

[29] J. G. Roney, "Effective security awareness training for end-users: A comprehensive approach," in Proc. InfoSecCD, 2017, pp. 1-9, doi: 10.1145/3136825.3136832.

[30] P. Kumaraguru et al., "Teaching Johnny not to fall for phish," ACM Trans. Internet Technol., vol. 10, no. 2, pp. 1-31, May 2010, doi: 10.1145/1754393.1754396.

[31] G. L. Orgill, G. W. Romney, M. G. Bailey, and P. M. Orgill, "The urgency for effective user privacy-education to counter social engineering attacks on secure computer systems," in Proc. 5th Conf. Inf. Technol. Educ. (CITC5), Oct. 2004, pp. 177-181, doi: 10.1145/1029533.1029577.

[32] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, "A survey of phishing email filtering techniques," IEEE Commun. Surveys Tuts., vol. 15, no. 4, pp. 2070-2090, 4th Quart., 2013, doi: 10.1109/SURV.2013.030713.00020.

[33] A. Bergholz et al., "Improved phishing detection using model-based features," in Proc. 5th Conf. Email Anti-Spam (CEAS), Aug. 2008, pp. 1-10.

[34] C. Ludl, S. McAllister, E. Kirda, and C. Kruegel, "On the effectiveness of techniques to detect phishing sites," in Proc. 4th Int. Conf. Detection Intrusions Malware, Vulnerability Assessment (DIMVA), Jul. 2007, pp. 20-39, doi: 10.1007/978-3-540-73614-1_2.

[35] P. A. Grassi et al., "Digital identity guidelines: Authentication and lifecycle management," NIST Special Publication 800-63B, Jun. 2017, doi: 10.6028/NIST.SP.800-63b.

[36] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang, "Targeted online password guessing: An underestimated threat," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Oct. 2016, pp. 1242-1254, doi: 10.1145/2976749.2978339.

[37] N. Gunson, D. Marshall, H. Morton, and M. Jack, "User perceptions of security and usability of single-factor and two-factor authentication in automated telephone banking," Comput. Secur., vol. 30, no. 4, pp. 208-220, Jun. 2011, doi: 10.1016/j.cose.2010.12.001.

[38] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," Proc. IEEE, vol. 91, no. 12, pp. 2021-2040, Dec. 2003, doi: 10.1109/JPROC.2003.819611.

[39] S. Machani, R. Philpott, S. Srinivas, J. Kemp, and J. Hodges, "FIDO UAF architectural overview," FIDO Alliance Proposed Standard, Dec. 2014. [Online]. Available: https://fidoalliance.org/specs/fido-uaf-v1.0-ps-20141208/fido-uaf-overview-v1.0-ps-20141208.html

[40] A. K. Jain, K. Nandakumar, and A. Ross, "50 years of biometric research: Accomplishments, challenges, and opportunities," Pattern Recognit. Lett., vol. 79, pp. 80-105, Aug. 2016, doi: 10.1016/j.patrec.2015.12.013.

[41] A. K. Jain and K. Nandakumar, "Biometric authentication: System security and user privacy," Computer, vol. 45, no. 11, pp. 87-92, Nov. 2012, doi: 10.1109/MC.2012.364.

[42] A. K. Jain, A. Ross, and K. Nandakumar, "Introduction to biometrics," Springer, 2011, doi: 10.1007/978-0-387-77326-1.

[43] V. M. Patel, N. K. Ratha, and R. Chellappa, "Cancelable biometrics: A review," IEEE Signal Process. Mag., vol. 32, no. 5, pp. 54-65, Sep. 2015, doi: 10.1109/MSP.2015.2434151.

[44] J. Lang et al., "Security keys: Practical cryptographic second factors for the modern web," in Proc. Int. Conf. Financial Cryptogr. Data Secur. (FC), Feb. 2016, pp. 422-440, doi: 10.1007/978-3-662-54970-4_25.

[45] E. Stobert and R. Biddle, "The password life cycle: User behaviour in managing passwords," in Proc. 10th Symp. Usable Privacy Secur. (SOUPS), Jul. 2014, pp. 243-255, doi: 10.1145/2660267.2660360.

[46] J. Bonneau, C. Herley, P. C. Van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in Proc. IEEE Symp. Secur. Privacy (SP), May 2012, pp. 553-567, doi: 10.1109/SP.2012.44.

[47] D. Florêncio, C. Herley, and P. C. Van Oorschot, "An administrator's guide to internet password research," in Proc. 28th Large Installation Syst. Admin. Conf. (LISA), Nov. 2014, pp. 35-52.

[48] A. Barth, C. Jackson, and J. C. Mitchell, "Robust defenses for cross-site request forgery," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Oct. 2008, pp. 75-88, doi: 10.1145/1455770.1455782.

[49] J. Weinberger, P. Saxena, D. Akhawe, M. Finifter, R. Shin, and D. Song, "A systematic analysis of XSS sanitization in web application frameworks," in Proc. 16th European Symp. Res. Comput. Secur. (ESORICS), Sep. 2011, pp. 150-171, doi: 10.1007/978-3-642-23822-2_9.

[50] R. Pelizzi and R. Sekar, "Protection, usability and improvements in reflected XSS filters," in Proc. ACM Symp. Inf., Comput. Commun. Secur. (ASIACCS), May 2012, pp. 5-5, doi: 10.1145/2414456.2414457.

[51] https://www.semanticscholar.org/paper/b661b35ce6421c7065a113860606bc9af835379a

[52] J. Bozic and F. Wotawa, "PURITY: A planning-based security testing tool," in Proc. IEEE Int. Conf. Softw. Quality, Rel. Secur. (QRS), Jul. 2015, pp. 46-55, doi: 10.1109/QRS.2015.17.

[53] I. Ristic, "Web application firewalls primer," in Proc. Black Hat USA, Jul. 2005, pp. 1-42.

[54] S. Lekies, B. Stock, M. Wentzel, and M. Johns, "The unexpected dangers of dynamic JavaScript," in Proc. 24th USENIX Secur. Symp. (USENIX Security), Aug. 2015, pp. 723-735.

[55] https://www.semanticscholar.org/paper/008f897ace40c54ede1d9e978f83d0e9c3316a1d

[56] R. Arora, S. Goel, and R. Mittal, "Automation of application layer DDoS attack detection using honeypots and machine learning," in Proc. 12th Int. Conf. Inf. Syst. Secur. (ICISS), Dec. 2016, pp. 219-238, doi: 10.1007/978-3-319-49806-5_11.

[57] S. Stamm, B. Sterne, and G. Markham, "Reining in the web with content security policy," in Proc. 19th Int. Conf. World Wide Web (WWW), Apr. 2010, pp. 921-930, doi: 10.1145/1772690.1772784.

[58] D. Hausknecht, J. Magazinius, and A. Sabelfeld, "May I? - Content Security Policy endorsement for browser extensions," in Proc. 13th Int. Conf. Detection Intrusions Malware, Vulnerability Assessment (DIMVA), Jul. 2015, pp. 261-281, doi: 10.1007/978-3-319-20550-2_14.

[59] S. Van Acker, D. Hausknecht, and A. Sabelfeld, "Measuring login webpage security," in Proc. ACM Symp. Appl. Comput. (SAC), Apr. 2017, pp. 1753-1760, doi: 10.1145/3019612.3019798.

[60] M. Vasek and T. Moore, "Identifying risk factors for webserver compromise," in Proc. 18th Int. Conf. Financial Cryptogr. Data Secur. (FC), Mar. 2014, pp. 326-345, doi: 10.1007/978-3-662-45472-5_21.

[61] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, "Cutting the gordian knot: A look under the hood of ransomware attacks," in Proc. 12th Int. Conf. Detection Intrusions Malware, Vulnerability Assessment (DIMVA), Jul. 2015, pp. 3-24, doi: 10.1007/978-3-319-20550-2_1.

[62] Y. Cao, V. Rastogi, Z. Li, Y. Chen, and A. Moshchuk, "Redefining web browser principals with a configurable origin policy," in Proc. 43rd Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN), Jun. 2013, pp. 1-12, doi: 10.1109/DSN.2013.6575349.

[63] X. Dong, M. Tran, Z. Liang, and X. Jiang, "AdSentry: Comprehensive and flexible confinement of JavaScript-based advertisements," in Proc. 27th Annu. Comput. Secur. Appl. Conf. (ACSAC), Dec. 2011, pp. 297-306, doi: 10.1145/2076732.2076775.

[64] P. Eckersley, "How unique is your web browser?" in Proc. 10th Int. Symp. Privacy Enhancing Technol. (PETS), Jul. 2010, pp. 1-18, doi: 10.1007/978-3-642-14527-8_1.

[65] A. P. Felt, R. W. Reeder, H. Almuhimedi, and S. Consolvo, "Experimenting at scale with Google Chrome's SSL warning," in Proc. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI), Apr. 2014, pp. 2667-2670, doi: 10.1145/2556288.2557292.

[66] B. AIC, H. Mendes, M. Barros, P. Pinto, and J. Martins, "Towards a unified security ontology," in Proc. 13th Eur. Conf. Cyber Warfare Secur. (ECCWS), Jul. 2014, pp. 13-20.

[67] S. More, M. Matthews, A. Joshi, and T. Finin, "A knowledge-based approach to intrusion detection modeling," in Proc. IEEE Symp. Secur. Privacy Workshops (SPW), May 2012, pp. 75-81, doi: 10.1109/SPW.2012.26.

[68] D. Huang, M. Xu, J. Xing, K. Ye, and S. Yu, "Smart detection on abnormal behavior in cloud applications," in Proc. 3rd Int. Conf. Cloud Comput. Intell. Syst. (CCIS), Nov. 2014, pp. 637-642, doi: 10.1109/CCIS.2014.7175820.

[69] M. N. Hossain, S. M. Milajerdi, J. Wang, B. Eshete, R. Gjomemo, R. Sekar, S. Stoller, and V. N. Venkatakrishnan, "SLEUTH: Real-time attack scenario reconstruction from COTS audit data," in Proc. 26th USENIX Secur. Symp. (USENIX Security), Aug. 2017, pp. 487-504.

[70] C. Reis and S. D. Gribble, "Isolating web programs in modern browser architectures," in Proc. 4th ACM European Conf. Comput. Syst. (EuroSys), Apr. 2009, pp. 219-232, doi: 10.1145/1519065.1519090.

[71] A. Moshchuk, H. J. Wang, and Y. Liu, "Content-based isolation: Rethinking isolation policy design on client systems," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Nov. 2013, pp. 1167-1180, doi: 10.1145/2508859.2516722.

[72] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in Proc. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI), Apr. 2006, pp. 581-590, doi: 10.1145/1124772.1124861.

[73] Y. Cao, X. Pan, Y. Chen, and J. Zhuge, "JShield: Towards real-time and vulnerability-based detection of polluted drive-by download attacks," in Proc. 30th Annu. Comput. Secur. Appl. Conf. (ACSAC), Dec. 2014, pp. 466-475, doi: 10.1145/2664243.2664253.

[74] K. Vikram, A. Prateek, and B. Livshits, "Ripley: Automatically securing web 2.0 applications through replicated execution," in Proc. 16th ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Nov. 2009, pp. 173-186, doi: 10.1145/1653662.1653683.

[75] R. Verizon, "Data breach investigations report," Verizon Business, Tech. Rep., 2021. [Online]. Available: https://www.verizon.com/business/resources/reports/dbir/

[76] CyberEdge Group, "2021 cyberthreat defense report," CyberEdge Group, Tech. Rep., Mar. 2021. [Online]. Available: https://cyber-edge.com/cdr/

[77] Fatima Omar, Dalia Ahmed, Omar Elnakib, Mohamed Ahmed, Nada Farhan, Hanan Hindy, Mahmoud Abdel-Hamid, Yehia Badawi, “Towards a User-Friendly Web Application Firewall.” International Conference on the Internet, Cyber Security and Information Systems, November 2023. [Online]. Available: https://www.semanticscholar.org/paper/a24ea5af7bbe666b173721464d9911a563f4d8d0

[78] Mengxia Ren, Chuan Yue, “Coverage and Secure Use Analysis of Content Security Policies via Clustering,” European Symposium on Security and Privacy, July 2023. [Online]. Available: https://www.semanticscholar.org/paper/bb6ae1901b970765e7e11282cb35ecd121b0e7f2

[79] Seongil Wi, Trung Tin Nguyen, Jihwan Kim, Ben Stock, Sooel Son, “DiffCSP: Finding Browser Bugs in Content Security Policy Enforcement through Differential Testing,” Network and Distributed System Security Symposium, 2023. [Online]. Available: https://www.semanticscholar.org/paper/59a309a04aae6cbe877db8bfcb680b05f19bb2c0