International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT217281

Risks and Threats to Web Applications and Their Preventions: A Theoretical Study on Vital Risks and Threats

Yogesh Kumar*, Anumalla Sandeep Satyanarayana, Ankit Kumar, Vikas Sharma
School of Computer Science and Engineering, Lovely Professional University, Jalandhar, (Punjab) India

Article Info
Volume 7, Issue 2
Page Number: 250-262

Publication Issue :
March-April-2021

Article History
Accepted : 18 April 2021
Published : 24 April 2021

ABSTRACT

With the rapid evolution of technology, almost every business is now online connecting them to the widest and narrow corners of the world. Therefore, instead of physical security, their online security is a pivotal concern the business which all depends on the web applications security. Web application attacks and their risks have become normal since past many years, and the security of web applications has received increased attentions at present. Many attacks work on real time and mostly prevention mechanisms focus on prevention and detection of these attacks on the web applications. This research focuses on giving attention to the top 10 threats that organizations need to know and to ensure the web applications are protected from these risks and attacks.

Keywords : Web Application, Web Application Security, Vulnerabilities, Risks, Threats, Cyber Security.

I. INTRODUCTION

A web application is the system by which user can interact with a service. A three-level web application can have multiple clients in the front-end of the web application, a webserver or API at middle level and a database server at back-end. The client sends the request to the web service then server responds with an appropriate reply or with data from database to the user. Web application security is the primary concern of any web-based business. And various vulnerabilities present inside the application can act as a security threat.

Web application vulnerabilities are system flaws or weakness in a web-based application commonly arise if form inputs are not properly validated or sanitized, misconfigured web servers, and application design flaws, and they can be exploited to compromise the application's security. Vulnerabilities arise because web applications need to interact with multiple users across multiple networks, and that level of accessibility is easily taken advantage of by hackers.

To ensure the security of the web applications, the very first step is to thoroughly study and analyze various vulnerabilities and attack methods. Because only after understanding all the attack methods and

vulnerabilities, we can take effective actions to prevent them.

In this paper we will analyze the current security situation of the web applications. Then we will list the vulnerabilities and threats that can we present inside a web application. After studying the vulnerabilities finally, we will provide some suggestions to prevent the attacks.

The rest of the paper is organized as follows. In section 2 we look at the current situation of web application security. We introduce vulnerabilities and threats in section 3. Suggestions about the security of webapps is in section 4. we draw a conclusion in section 5.

II. CURRENT SITUATION OF WEB APPLICATIONS SECURITY

Recently from last year, covid-19 has forced many organizations to move to work from home which means website security was needed to be more safe and secure. But on the contrast, cyber-attacks increased more than ever before indicating the crucial requirement of web application security. The latest statistics says that 95% of the security attacks were possible due to human errors leading to theft of personal and confidential data. “Increased dependence on web-enabled applications in the form of APIs,” the report said predicts API abuses to be the most frequent attack vector in the future, which spells unwelcome news for lots of organizations. In last year, more than 50% organizations experienced DDOS attacks, 49% experienced the injection attack, 82% had face the malicious bot attacks. SQL injection and Cross Site Scripting (XSS) attack topped last year in web application attacks. A 2003 study had found that there is an attack every 39 seconds on the web, and they were true. And Attacks on web applications are increasing by 13% by every year. In the last year, hacking and phishing attempts were up 37% month-on-month. Phishing attempts have soared by over 600% since the end of February, including traditional impersonation frauds but also business email compromise (BEC) and extortion attacks, according to Barracuda Networks.

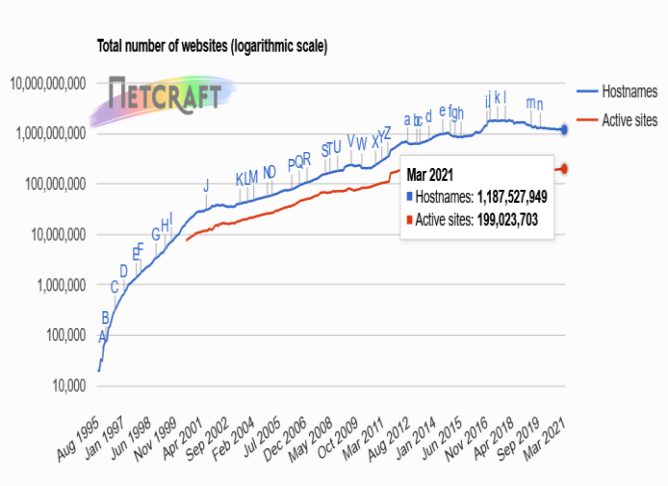

Figure 1. Statistics of websites quantity carried out by Netcraft.

Impact of web application attacks is always crucial and varied differently on different industries.

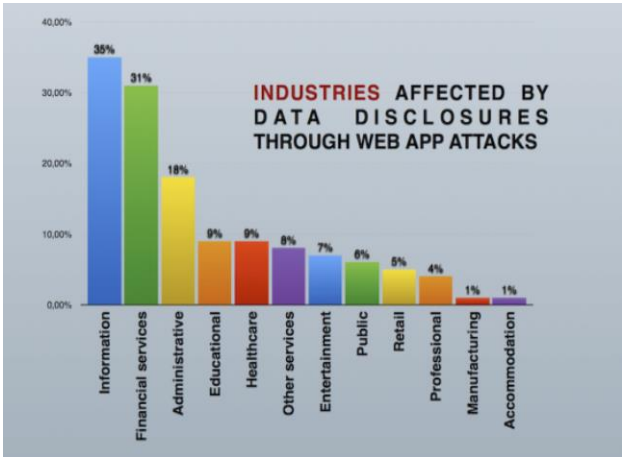

Figure 2. Industries affected by data breaches through web app attack.

The attacks motives are always different for many cyber criminals which are also divided in distinct categories which are hacktivists, state sponsored hackers, organized cyber criminals and some unaffiliated cyber criminals. Below is the graphical representation of involvement of these categories depending on threat actors from last few years:

Figure 3

With the rise in number of websites each day, there is a necessary requirement to understand and implement web application security.

III. COMMON VULNERABILITIES AND THREATS TO WEBAPPS

OWASP Top 10 organization talks about the ranking of the ten most dangerous information security risks for web applications, compiled by a community of industry experts. Below table shows the top 10 most dangerous web application security risks in the year 2021:

TABLE I

2021 TOP 10 POTENTIAL SECURITY RISKS PUBLISHED BY OWASP

Attack No.	Type of attack	Description of attack
1.	Injection	Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query without validating the input.
2.	Broken Authentication	Application methods that concerned with authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to use other users' identities temporarily or permanently.
3.	Sensitive Data Exposure	Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.
4.	XML External Entities	It is a web security vulnerability that allows an attacker to interfere with a web application's processing of XML data. It can allow an attacker to have access to the files on the server filesystem, and to interact with any back end or external systems that the application itself can access.
5.	Broken Access Control	If restrictions and access to what authenticated users can do are often not properly enforced. Attackers can use these flaws to access unauthorized functionality and/or data, such as access other users' accounts, view sensitive files, modify other users' data, change access rights, etc.

6.	Security Misconfigurations	This is a most common issue, commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information.
7.	Cross-Site Scripting (XSS)	XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping or updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.
8.	Insecure Deserialization	Insecure deserialization often leads to remote code execution. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks.
9.	Using components with known vulnerabilities	Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover.
10.	Insufficient Logging and Monitoring	Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data.

A. INJECTION

Injection is the first risk of "top 10 2021". In this subsection, we will give details of the principles of SQL injection attack. Since SQL injection enables the attackers to access database by bypassing firewall restriction and to obtain the administrator rights, the risk of SQL injection is much higher than other risks. Moreover, there are many other attacks that are like SQL injection. They break the security of system, e.g., obtain administrator rights, steal confidential information, and impersonate legitimate users, by using the vulnerabilities of web applications. The so-called SQL injection is to cheat the server to execute malicious SQL commands by inserting SQL commands into the query string of Web form submission or input domain name or page request. SQL injection attacks occur when dynamic SQL statements are constructed using input content to access the database. SQL injection also occurs if the code uses stored procedures, which are passed as strings containing unfiltered user input. Hackers can get access to the website database through the SQL injection attack, and then they can get all the data in the website database. Malicious hackers can tamper with the data in the database through the SQL injection function and even destroy the data in the database. As a web developer, you hate this kind of hacking.

The following is a simple example of SQL injection:

SELECT * FROM users WHERE login = 'admin' AND password = '123'

The assumed ASP code var sql = "SELECT * FROM users WHERE login = '" + formuser + "' AND password = '" + formpwds + "'".

The input

formuser = ' or 1 = 1 -- formpwds = anything

Inquiry code

SELECT * FROM users WHERE usemame =‘, or 1=1 --

AND password = 'anythings'

In the condition statements, whatever the username is, because 1=1 holds forever, SELECT will return all the data in the users list. This will cause the attacker to login the system successfully. In other words, the attacker is easily able to inquiry some confidential data by striking some SQL statements. In the same way, the attacker is also able to delete, inquiry, insert, or update the data of the database by constructing specific SQL statements in the input

IV. SUGGESTIONS ABOUT THE SECURITY OF WEBAPPS

In this section we will provide some suggestions to increase the security of the webapps. The suggestions refer to three aspects: technical means, security detection, and security administration.

A. Technical Means

1) Deploy Firewall: Web Application Firewalls (WAFs) are deployed to protect web applications and they offer in depth security if they are configured correctly. A problem arises when there is over-reliance on these tools. A false sense of security can be obtained with the implementation of a WAF.
2) Securing Data: With amount of sensitive data available online increasing day by day, the need for ways to protect the same is increasing too. Following are a few ways of securing the web data:

- Avoid deploying database on the same server on which the application is installed because admin accounts can easily be targeted.
- The files and backup information should be encrypted using industry standard encryption techniques.
- Database can be secured by implementing firewalls and input validations as the attackers can attack using SQL injections.

3) Secure Coding: Secure coding is the practice of writing application that is protected from vulnerabilities. An insecure application is vulnerable to attacks. Attackers can take direct control of a web app. To write secure web apps developer should comply with some industry standards.

- “Never trust the input of the users” is an important criterion while writing an application. Each input should be properly validated.
- Every function should perform only one basic function. The more concise a program is the less error it will give.
- Each error should be recorded securely.

B. Security Detection

Security detection is an important step in increasing web app security. We can check the security of a webapp by analysing the logs.

We can also hire security experts or ethical hackers to test the web app security. Ethical hackers reduce the risk of breaches and also reduces the risk liability that may arise from a breach. By hiring a certified professional, organizations confirm their commitment to security.

A few ways that an ethical hacker can save your organization by reducing losses in case of a breach are –

- An ethical hacker can locate the vulnerability faster to prevent the ongoing attack.
- An ethical hacker can help you with employee fidelity bond or suggest insurance that can reimburse your organizational losses as a result of their activities.

C. Security Administration

Most of organisations pay more attention to the technical protection of web applications and neglect the vulnerabilities that are caused by careless security administration. Therefore, we should strength safety

education, and introduce safety knowledge. Only in this way we can prevent the potential security problems. Another important thing is to build perfect management system, such as enforce training and retraining on the personnel to operating and maintaining system and access the security of the web applications periodically.

V. CONCLUSION

As the world is diving more deeper into web applications and digital era, the security of these web applications is becoming a major concern for not only us individuals but also for major organizations. These security problems are catching attention of cyber criminals, and it is a high time that people get to know about these security issues to protect the data and reputations of many. The reality is also that the security of web applications is becoming more indispensable to the security of systems. In our paper, we introduced many risks and threats and vulnerabilities that can cause damage to web applications and also provided several suggestions which can prevent these problems. We suggested how top global attacks can damage these applications and how we can prevent them in development phase, testing process, maintenance and operations of these web applications. And moreover, what we discussed in our paper can ensure the security of the web applications to an utmost degree, and further reducing any damage for users and any organizations.

VI. REFERENCES

[1]. Clincy, V., & Shahriar, H. (2018). Web Application Firewall: Network Security Models and Configuration. 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). doi:10.1109/compsac.2018.00144

[2]. DivyaniYadav, Gupta, D., Singh, D., Kumar, D., & Sharma, U. (2018). Vulnerabilities and Security of Web Applications. 2018 4th International Conference on Computing Communication and Automation (ICCCA). doi:10.1109/ccaa.2018.8777558

[3]. Ma, L., Zhao, D., Gao, Y., & Zhao, C. (2019). Research on SQL Injection Attack and Prevention Technology Based on Web. 2019 International Conference on Computer Network, Electronic and Automation (ICCNEA). doi:10.1109/iccnea.2019.00042

[4]. You Yu, Yuanyuan Yang, Jian Gu, and Liang Shen Ministry of Public Security Quality Supervision and Testing Center of Security Products for Computer Information System the Third Research Institute of Ministry of Public Security Shanghai, China yuy@mctc.gov.cn

[5]. why Insufficient Logging and Monitoring Can Help Attackers ..., https://apiacademy.co/2020/04/why-insufficient-logging-and-monitoring-can-help-attackers-hide-in-plain-sight/.

[6]. Application Security | goPayroll.net, https://gopayroll.net/security/.

[7]. How to detect Cross Site Scripting Issues (XXS ..., https://www.omnicybersecurity.com/how-to-detect-cross-site-scripting-issues-xxs/.

[8]. https://www.webarxsecurity.com/website-hacking-statistics-2018-february/

[9]. https://www.varonis.com/blog/cybersecurity-statistics/

[10]. https://www.webarxsecurity.com/website-hacking-statistics-2018-february/

[11]. https://www.infosecurity-magazine.com/news/cyberattacks-up-37-over-past-month/

[12]. https://news.netcraft.com/archives/category/web-server-survey/

[13]. https://owasp.org/www-project-top-ten

Cite this article as : Yogesh Kumar, Anumalla Sandeep Satyanarayana, Ankit Kumar, Vikas Sharma, "Risks and Threats to Web Applications and Their Preventions : A Theoretical Study on Vital Risks and Threats ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 2, pp. 432-438, March-April 2021. Available at doi : https://doi.org/10.32628/CSEIT217281
Journal URL : https://ijsrcseit.com/CSEIT217281