International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 02, ISSN No. 2455-2143, Pages 174-177
Published Online June 2023 in IJEAST (http://www.ijeast.com)

DATA SECURITY AND DATA BREACHES

Ishaan Gupta, Ira, Aditya Gusain, Kusum Mahajan, Faisal Rais, Rekha Chaudhary
HMR Institute of Technology & Management,
New Delhi-110036, India

Abstract: **This research paper focuses on the topic of data security and data breaches specifically in web applications. As more businesses and organizations move their operations online, web applications have become a prime target for cyber attacks. This paper examines common vulnerabilities in web applications such as SQL injection, cross-site scripting, and insecure session management and how they can be exploited to gain unauthorized access to sensitive data. Venmo, a popular peer-to-peer mobile payment app, has faced several data breaches in recent years, raising concerns about the safety of user information. This research paper will also examine the causes and consequences of these data breaches and discusses best practices for secure web application development.**

I. INTRODUCTION

In recent years, the use of web applications has increased dramatically, as more businesses and organizations move their operations online. Along with this growth, the number of data breaches has also risen, resulting in the loss of sensitive information, financial losses, and damage to reputation. Data security has become a crucial issue for all organizations that rely on web applications. This research paper will examine the current state of data security and data breaches in web applications, common vulnerabilities, and best practices for secure web application development and deployment. Additionally, this paper will explore the potential solution of decentralizing web applications using blockchain technology. The popularity of mobile payment apps, such as Venmo, has grown rapidly in recent years as they provide a convenient and easy way for individuals to send and receive money. However, with the increasing use of these apps comes the need for heightened data security. Venmo, in particular, has faced several data breaches in recent years, resulting in the exposure of sensitive user information.

II. BACKGROUND

Web applications are a popular target for cyber attacks, as they often contain sensitive information such as personal data, financial information, and intellectual property. Common vulnerabilities in web applications include SQL injection, cross-site scripting, and insecure session management. These vulnerabilities can be exploited by attackers to gain unauthorized access to sensitive data, resulting in data breaches[1].

To mitigate the risks of data breaches, organizations should implement secure coding practices, conduct regular security assessments, and implement access controls. Compliance standards, such as PCI-DSS and HIPAA, also provide guidance on how to secure web applications [2]. However, despite these best practices and compliance standards, data breaches continue to occur.

We our primarily focusing on venmo payment application since it has came under scrutiny multiple times over years. Venmo is a mobile payment app owned by PayPal that allows individuals to easily send and receive money, as well as make purchases using their mobile device. The app has become increasingly popular, with over 40 million active users in the United States alone. However, despite its popularity, Venmo has faced several data breaches in recent years. In 2017, it was reported that a hacker had gained access to the personal information of thousands of Venmo users, including their full names, email addresses, and phone numbers. The hacker then sold this information on the dark web. In 2018, another data breach occurred, resulting in the exposure of over 200 million Venmo transactions, including details such as the amount of money transferred, the recipient's name, and the message associated with the transaction.[3]

III. TECHNOLOGIES USED

React- React is a free and open-source front-end JavaScript library for building user interfaces based on UI components.

Solidity - Solidity is an object-oriented programming language for implementing smart contracts on various blockchain platforms, most notably, Ethereum.

Morralis - Moralis is a leading web3 development platform that offers everything a user needs to create, launch and grow great decentralized applications (dapps) in one place.

Open Zeplin – OpenZeppel in provides security products to build, automate, and operate decentralized applications.

Metamask - MetaMask is a software cryptocurrency wallet used to interact with the Ethereum blockchain.

Node.js - Node.js is a cross-platform, open-source server environment that can run on Windows, Linux, Unix, macOS, and more.

Etherscan - Etherscan is a Block Explorer and Analytics Platform for Ethereum, a decentralized smart contracts platform.

Ethereum - Ethereum is a decentralized, open-source blockchain with smart contract functionality. Ether is the native cryptocurrency of the platform.

Truffle - Truffle is a world-class development environment, testing framework and asset pipeline for blockchains using the

Ethereum Virtual Machine (EVM), aiming to make life as a developer easier.

Infura- Infura is a service that provides a scalable and reliable way to connect to the Ethereum blockchain without having to run a full node. It allows developers to access the Ethereum network without having to worry about the maintenance and scaling of their own infrastructure.

Remix - Remix is a browser-based integrated development environment (IDE) for writing, testing, and debugging Solidity smart contracts. It allows developers to write code, test contracts, and deploy them to the Ethereum blockchain.

Geth - Geth is a command-line interface (CLI) for running a full Ethereum node on a local machine. It is written in Go and is one of the most widely used clients for running a node on the Ethereum network.

Web3.js - Web3.js is a JavaScript library that allows developers to interact with the Ethereum blockchain from within a web browser. It provides an API for sending transactions, querying the blockchain, and interacting with smart contracts.

IV. IDENTIFY, RESEARCH AND COLLECT IDEA

There are several steps we can take to identify, research, and collect data on this topic.

Identify the problem or challenge we are trying to solve: The first step is to clearly define the problem or challenge that we are trying to address with our project. This will help us to focus our research and ensure that we are collecting ideas that are relevant and useful for our specific goals.

Research existing approaches and techniques: Once we have defined our problem or challenge, we can begin researching existing approaches and techniques for solving it. This can include reading academic papers and articles, reviewing existing software libraries and frameworks, and searching online for relevant resources and examples.

This research paper will be conducted through a review of literature and analysis of publicly available data. The literature review will focus on the data security measures implemented by Venmo and their effectiveness in preventing data breaches. The analysis of publicly available data will include a review of reported data breaches and their impact on Venmo users and the company's reputation. The interviews with industry experts will provide insight into the challenges faced by mobile payment apps in terms of data security and the potential solutions to these challenges.

There are several common causes of data breaches in web applications. One of the most significant is poor security practices, such as using weak passwords or failing to keep software up to date. In addition, web applications may be vulnerable to common attacks, such as SQL injection and cross-site scripting (XSS) [4]. Another cause of data breaches is the use of third-party libraries and frameworks. Many web applications are built using open-source libraries and frameworks, which can introduce vulnerabilities if they are not properly secured.

Consequences of Data Breaches in Web Applications Data breaches can have severe consequences for businesses and individuals. One of the most significant is financial loss, which can occur as a result of data breaches. For example, a business may lose customers or face legal liability if sensitive information is compromised. In addition to financial losses, data breaches can also cause reputational damage. This is particularly true for businesses that handle sensitive information, such as personal data or financial information. A data breach can undermine consumer trust and lead to long- term damage to a business's reputation[5]. Another consequence of data breaches is the impact on individuals whose personal information has been compromised. This can include identity theft, financial loss, and emotional distress.

Potential Solutions: There are several potential solutions to the problem of data security and data breaches in web applications. One approach is to improve security practices, such as using strong passwords and keeping software up to date. In addition, web applications should be built using secure coding practices, such as input validation and error handling. Another approach is to use web application firewalls (WAFs) and intrusion detection and prevention systems (IDPS) to detect and block attacks[6]. These systems canhelp to protect web applications from common attacks, such as SQL injection and XSS. Finally, businesses should have incident response plans in place to quickly and effectively respond to data breaches. This includes identifying and containing the breach, as well as notifying affected parties and taking steps to prevent future breaches.

There are several other methods too that can be used to prevent and mitigate data breaches in web applications. These include: Secure coding practices: Ensuring that web application code is properly tested and free of vulnerabilities is essential in preventing data breaches. This includes validating user input and using secure coding techniques such as input sanitization. Encryption: Encrypting sensitive information both in transit and at rest can help protect against data breaches. This includes using secure protocols such as HTTPS for transmitting information and using encryption algorithms such as AES or RSA for storing information [7]. Access control: Implementing proper access controls can help prevent unauthorized access to sensitive information. This includes using authentication mechanisms such as usernames and passwords, and implementing role-based access controls. Incident response plan: Having a well-designed incident response plan in place can help organizations quickly and effectively respond to a data breach. This includes identifying and containing the breach, as well as taking steps to prevent future breaches.

Recommendations: Based on the research findings, it is recommended that Venmo implements additional security measures such as end-to-end encryption and regular security audits. Additionally, the company should increase transparency and communication with users regarding data breaches and their impact. Furthermore, Venmo should consider hiring third-party security experts to conduct regular security assessments of the platform to identify potential vulnerabilities and provide recommendations for mitigating them.

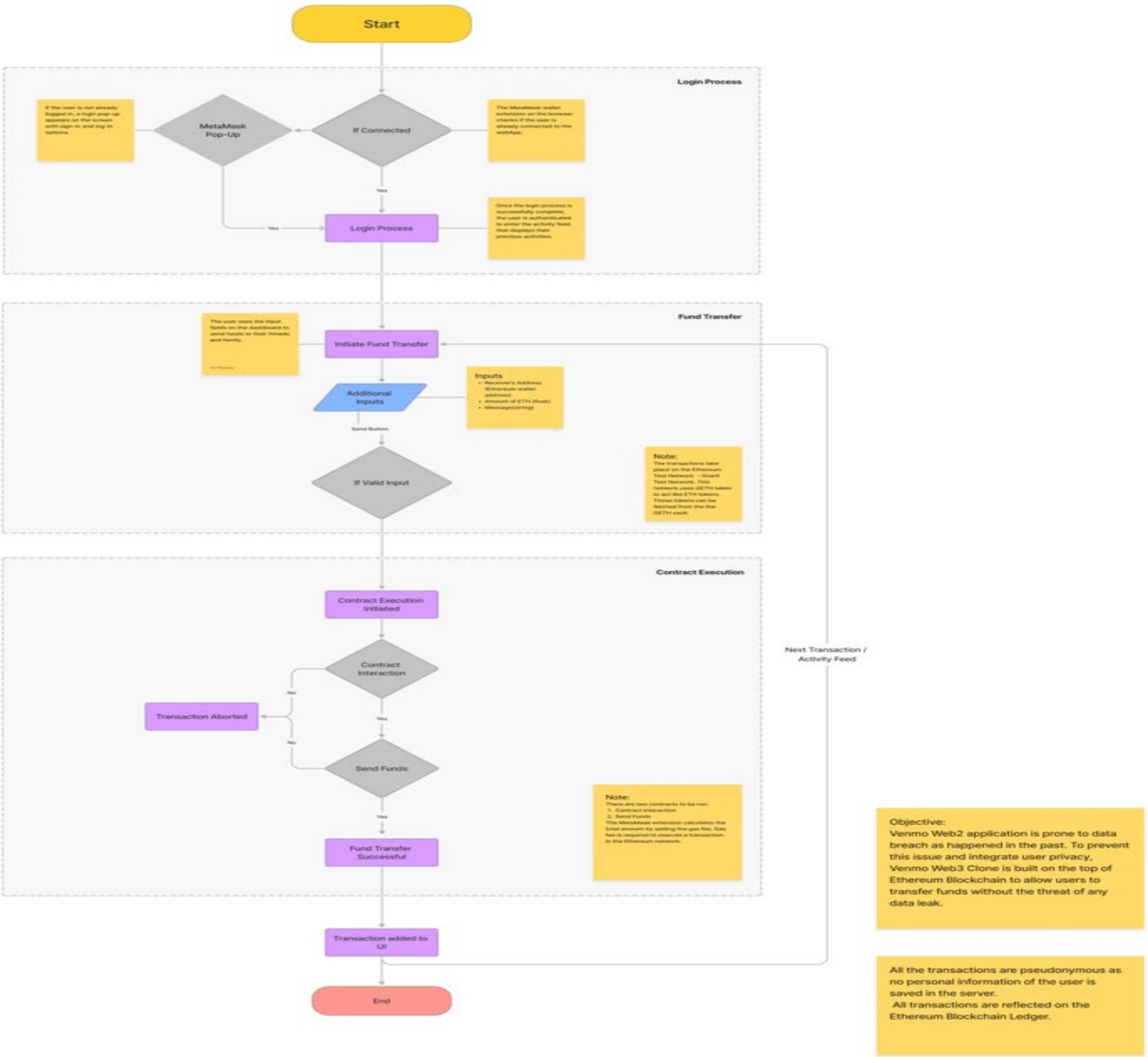
V. CONCLUSION

The use of web applications continues to grow, and with it the risks of data breaches. Ensuring the security of web applications is essential in protecting sensitive information. By implementing secure coding practices, encryption, access control and incident response plan, organizations can help prevent and mitigate data breaches in web applications. However, as the technology and the attackers evolve, it is crucial to stay updated with the latest security measures and best practices to keep the data secure.

Decentralized applications (dApps) have the potential to revolutionize various industries by providing a more transparent, secure, and decentralized way of managing and governing digital assets and services. While dApps are still in their infancy, their potential impact is significant and we will likely see continued growth and development in the future. [8]

This research paper has analyzed the data security measures implemented by Venmo and evaluated their effectiveness in preventing data breaches. It has also examined the impact of data breaches on Venmo users and the company's reputation. The research has revealed that although Venmo's data security measures are robust, they have not been enough to prevent data breaches. The paper also highlights the challenges faced by other web apps in terms of data security and the potential solutions to these challenges.

VI. DATA FLOW DIAGRAM



If the user is not already logged in, a login pop-up appears on the screen with sign-in and log-in options.

The MetaMask wallet extension on the browser checks if the user is already connected to the webApp.

Once the login process is successfully complete, the user is authenticated to enter the activity feed that displays their previous activities.

The user uses the input fields on the dashboard to send funds to their friends and family.

Inputs
- Receiver's Address (Ethereum wallet address)
- Amount of ETH (float)
- Message(string)

Note:
The transactions take place on the Ethereum Test Network - Goerli Test Network. This network uses GETH token to act like ETH tokens. These tokens can be fetched from the the GETH vault.

Note:
There are two contracts to be run:
1. Contract Interaction
2. Send Funds
The MetaMask extension calculates the total amount by adding the gas fee. Gas fee is required to execute a transaction in the Ethereum network.

Objective:
Venmo Web2 application is prone to data breach as happened in the past. To prevent this issue and integrate user privacy, Venmo Web3 Clone is built on the top of Ethereum Blockchain to allow users to transfer funds without the threat of any data leak.

All the transactions are pseudonymous as no personal information of the user is saved in the server.
All transactions are reflected on the Ethereum Blockchain Ledger.

Fig. 1 Data Flow Diagram

VII. REFERENCES

[1]. OWASP Homepage. Top 10 Web Application Security Risks. https://owasp.org/Top10/,last accessed on 2022/11/15.

[2]. National Institute of Standards and Technology. (2018). NIST SP 800-53: Security and Privacy Controls for Federal Information Systems and Organizations. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/ NIST.SP.800- 53r4.pdf

[3]. Hacker Exposes 200 Million Venmo Transactions" (2016, August 12). Retrieved from https://www.wired.com/2016/08/, last accessed on 2022/10/27.

[4]. Data Breach 101: Top 5 Reasons it Happens - WHOA.com last accessed on 2022/11/18.

[5]. IBM.Cost of a Data Breach Report 2022,https://www.ibm.com/downloads/cas/3R8N1DZ J, last accessed on 2023/01/15.

[6]. Rashmi Bhardwaj, article "how is ips/ids different from waf".

[7]. Phillip Williams.A survey on security in internet of things with a focus on the impact of emerging technologies Author links open overlay panel , https://doi.org/10.1016/j.iot.2022.100564.

[8]. Decrypt homepage,https://decrypt.co/resources/what-are-decentralized-applications-dapps, last accessed on 2022/12/01.