HASIL CEK_Smart Payment Application Security Optimization from Cross-Site Scripting (XSS) Attacks Based on Blockchain Technology

by Imam Riad Umar, Lestari3

Submission date: 23-Apr-2022 11:19AM (UTC+0700)
Submission ID: 1817906484
File name: ss-Site_Scripting_XSS_Attacks_Based_on_Blockchain_Technology.pdf (740.39K)
Word count: 5074
Character count: 27238

Terbit *online* pada laman web jurnal :
http://ejournal.amikompurwokerto.ac.id/index.php/telematika/

Telematika

Accreditated SINTA "2" Kemenristek/BRIN, No. 85/M/KPT/2020

Smart Payment Application Security Optimization from Cross-Site Scripting (XSS) Attacks Based on Blockchain Technology

Imam Riadi[1], Rusydi Umar[2], Tri Lestari[3]

[1] Department of Information System, Universitas Ahmad Dahlan
[2,3] Department of Informatics, Universitas Ahmad Dahlan
E-mail:imam.riadi@is.uad.ac.id[1], rusydi@mti.uad.ac.id[2], tri1907048008@webmail.uad.ac.id[3]

ARTICLE INFO

History of the article:
Received February 4, 2021
Revised March 17, 2021
Accepted June 19, 2021
Available online August 31, 2021

Keywords:
IoT
Vulnerabillity
XSS
Security
Blockchain

Correspondece:
Telepon: +62 815-6854-308
E-mail: imam.riadi@is.uad.ac.id

ABSTRACT

The digital era is an era everyone has used technology and they are connected to each other very easily. The Smart Payment application is one of the applications that is developing in the digital era. This application is not equipped with security, so there is a concern that hackers will try to change user or even change user data. One of the possible attacks on this application is a cross-site attack (XSS). It is a code injection attack on the user side. Security in the Smart Payment application needs to be improved so that data integrity is maintained. In this research, security optimization is carried out by implementing blockchain. Blockchain has the advantage in terms of security with the concept of decentralization by utilizing a consensus algorithm that can eliminate and make improvements to data changes made by hackers. The result obtained from this study is the implementation of blockchain to maintain the security of payment transaction data on the Smart Payment application from XSS attacks. It is proven by the results of the vulnerability before and after blockchain implementation. Before the implementation of the vulnerability is found, 1 XSS vulnerability had a high level of overall risk. Meanwhile, the result of the vulnerability after blockchain implementation was not found from XSS attacks (the XSS vulnerability was 0 or not found).

INTRODUCTION

The digital era is an era when everyone has used technology and they are connected to each other very easily. The digital era appears due to technology that is developing rapidly. The Smart Payment application is one of the many applications that is developing in the digital era (Riadi et al., 2020). This application is a non-cash payment of school fees that was created to facilitate the payment transaction process and is a WEB-based application. The transaction process in this application is described in detail as follows; each user has a username and password provided by the school to be able to log into the Smart Payment application. Users can make various payment transactions including the payment for school fees, school uniforms, and construction fees. It can also be used to top up pocket money. Data entered through the system will be sent to the API rest server using Internet of Things (IoT)so that the data storage process is more effective.

IoT is a computational concept about objects in daily life that is connected to the internet and able to identify it selves to other devices. IoT is not only potential to influence lifestyle but also how it works (Setiadi and Muhaemin, 2018). The concept of IoT includes 3 main elements, namely: physical or real

objects that have been integrated with the sensor module, internet connection, and data centers on servers to store data or information from applications (Candra and Wiliam, 2019). The use of objects connected to the internet will collect data which is then collected into big data to be processed and analyzed by government agencies, related companies, and other agencies and then used for their respective interests (Sasmoko and Wicaksono, 2017). IoT has several advantages. IoT can improve user experience, increase device usage and help improve technology to make it more effective in its use (Taylor et al., 2020). The process of sending transaction data to the API rest server using IoT can be seen in Figure 1.

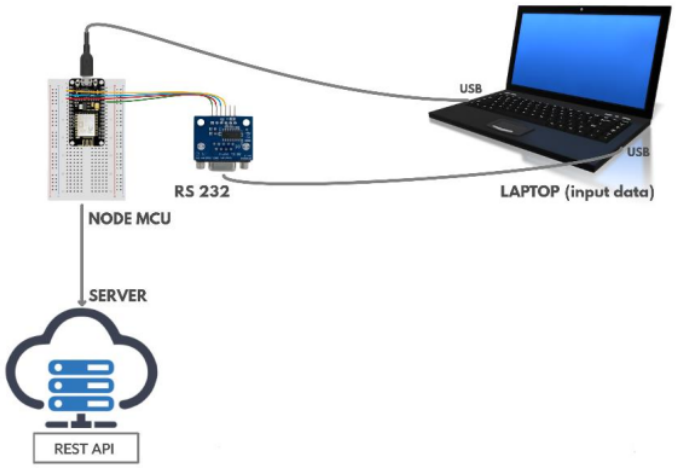

Figure 1. Data Sending Process to Server

Figure 1 describes data that is entered by the user via a laptop and forwarded to the REST API server using IoT. The Smart Payment application that is used for the data input process is not equipped with security, so there is a concern that hackers will try to change, delete, and even steal user data. Data destruction by hackers is of course very detrimental to users of the Smart Payment application. Therefore, this application needs to be equipped with security.

One of the attacks that might attack this application is a cross-site-scripting (XSS) attack, which is a code injection attack on the client-side by means of a website page or web application (Riadi and Raharja, 2019). Hackers will execute malicious scripts in the victim's browser by entering malicious code into a legitimate web page or web application. This attack can be performed using JavaScript, VBScript, ActiveX, Flash, and other client-side languages (Yunanri and Riadi, 2018). Forums, comment fields, and message boards are commonly used by attackers to post links to create malicious scripts (Rusdiana, Banta, and Sanusi, 2019). The script will then attack when the victim clicks on the link. XSS is often used to steal session cookies, allowing an attacker to impersonate a victim (Haryadi, Priyanto, and Anra, 2017). In this way, hackers can find out sensitive data belonging to the victim (Yulianingsih, 2017).

Security optimization in the Smart Payment application is carried out by implementing blockchain. Blockchain plays a role in securing payment transaction data entered by users through the Smart Payment application with analysis such as the following; a hacker sends a malicious script to the victim's email where the script usually contains information that is accompanied by a link that can make the victim affected to click on the link if the victim is careless or not careful.When the link is clicked, it will display a form that contains user data such as username and password. It then can be used by hacker to log into the smart

payment application. If the attack is successful, the data is sent to the hacker's web so that the hacker can use it to log into the smart payment application and disguise himself as a user who was successfully hacked. In this section the hacker can make changes and steal data, so the blockchain will start working. Changes to data by hackers will initially seem successful, but actually it is not. Blockchain uses a decentralized concept that utilizes a consensus algorithm where all transactions will be equalized on every blockchain network data manipulation efforts are useless, how often do they do except to attack all blockchain members (Rahardja et al., 2020) Figure 2 is a simulation of attacks and how to overcome them using blockchain technology.

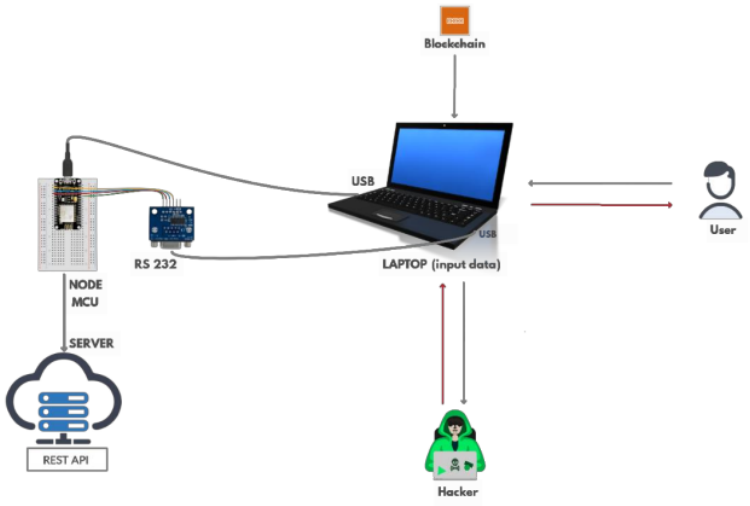

Figure 2. Blockchain attack analysis and implementation

Security optimization using blockchain technology has been carried out by previous researchers. In this research blockchain technology is used to increase security to avoid Man in the Middle Attack (MitM). The attacks as the result of the blockchain technology implementation have a block hash mechanism that can be used to close vulnerabilities in the authentication process (Busthomi et al., 2020). The hash block mechanism converts the authentication payload data in the form of plaintext into ciphertext data by converting the data into an encryption block (Winarto, 2019). The data blocked cannot be read so that it can guarantee the security and confidentiality of the payload data (Hu, Palit, and Handojo, 2019). Based on the results obtained, the implementation of Blockchain technology has succeeded in securing authentication payload data in an information system. Almost the same as previous research, in this study blockchain technology is used to secure applications from XSS attacks.

RESEARCH METHODS

This research is conducted in several stages. The first stage is the collection of research equipment. The tools needed in the previous study is used for IoT design including NodeMCU, RS232, USB cable, and laptop. Second, the tool used for vulnerability testing is the XSS Vulnerability Scanner software. The third tools used for blockchain planning and implementation is VisualStudio software. The following is a brief explanation regarding the tools used in the study.

1. NodeMCU can be analogous to the Arduino board of the ESP826 and has also packaged the ESP8266 into a compact board with various features like a microcontroller plus the ability to

access Wifi as well as a USB to serial communication chip, so for programming only a USB data cable extension is required. It is similar to the cable used in the data cable and charging cable for Android smartphones (Arief, 2019). The NodeMCU used in this study can be seen in Figure 3.

Figure 3. NodeMCU for designing IoT

The advantages of NodeMCU are low cost, integrated support for WiFi networks, a smaller board size, and lower energy consumption. Following are the basic specifications of NodeMCU; Tensilica 32bit Microcontroller, 4KB Flash Memory, 3.3V Operating Voltage, 7-12V Input Voltage, Digital I / O 16, Analog Input 1 (10 Bit), Interface UART 1, Interface SPI 1, Interface I2C 1 (Rozi et al., 2018). NodeMCU programm is installed through several step. The micro USB cable is connected to the NodeMCU USB port and a laptop USB port connected to the internet so that the download process can be carried out; if it is detected on the computer you will see USB-SERIAL CH340 on the device manager, for the com number it can be different on each computer (Stevenson and Aitken., 2019)

2. RS232 is a data transmission series communication standard between two electronic devices (Arief and Sundara, 2017). Serial data communication is done by sending data bit by bit sequentially. RS232 can be seen in Figure 4.

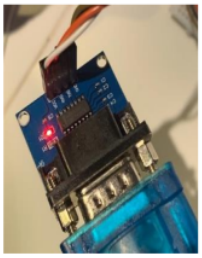

Figure 4. RS232 for designing IoT

Figure 3 is the RS232 used in the study. RS232 has been connected to a laptop using an HDMI to VGA cable and NodeMCU is connected with a cable that has been adjusted.

3. USB and laptop are two tools which are used to assist the IoT design process. As previously explained, RS232 must be connected to the laptop using a USB cable so that the input data can be read and processed by RS232 and NodeMCU.
4. XSS Vulnerability Scanner is a vulnerability testing tool for web applications and network infrastructure that is strong and strongly integrated so that with this tool, vulnerability testing becomes more effective and easy (Putra, 2017) (Suartana, Wahanani, and Sandy, 2015)
5. VisualStudio is used to design blockchain using the node.js programming language.

After the research tools are collected, it is continued with IoT design. IoT that is designed and connected to data, is tested for vulnerability using the XSS Vulnerability Scanner to find vulnerabilities,

especially from XSS attacks. If a vulnerability is found in the application, a blockchain is implemented to increase its security. After the blockchain implementation has been successfully carried out, it is continued with another vulnerability testing to prove whether the blockchain implementation to secure the IoT was successful. The flow carried out in this study can be seen in Figure 5.

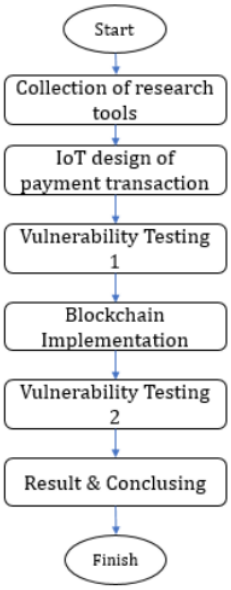

Figure 5. Research Methodology

Figure 5 is the flow or stages of the research carried out. First, research tools are collected, followed by IoT planning, Then, vulnerability testing is carried out followed by blockchain implementation and carried out with vulnerability testing after blockchain implementation. Finally, it is ended with drawing results and conclusions.

RESULTS AND DISCUSSION

This section shows the Payment Transaction IoT optimization process from an XSS attack. The first stage is the collection of research tools. The second stage is the determination of IoT design using RS232 and NodeMCU. The third stage is vulnerability testing before optimization is carried out. The fourth stage is the implementation of blockchain. The fifth stage of vulnerability testing after optimization is carried out. The first is the collection of equipment, the tools needed in this study can be seen in Table 1.

Table 1. Tools

No	*Name*	*Category*	*Information*
1	NodeMCU	Hardware	To design the IoT
2	RS232	Hardware	To design the IoT
3	Cable USB	Hardware	To design the IoT
4	Laptop	Hardware	To design the IoT
5	XSS Vulnerability Scanner	Software	For Pentes
6	VisualStudio	Software	For design Blockchain

Table 1 contains the research tools needed in research. Number 1 is the NodeMCU which is used to design the IoT. Number 2 is RS232 which is used to change the data entered by the user so that it can be read by NodeMCU. Numbers 3 and 4 are USB cables and laptops, both of these tools are used to assist the

IoT design process. Number 5 XSS Vulnerability Scanner to test the vulnerability of payment transactions IoT. Number 6 VisualStudio is used to design a blockchain that will be implemented in the payment transaction IoT.

Second, the design of payment transaction IoT which is used as a demonstration of sending cryptocurrency transaction data. The device is set up so that it can send data through sensors, namely data on the sender, receiver and nominal amount sent. IoT is designed using NodeMCU assisted by RS232. The IoT design in this study can be seen in Figure 6.

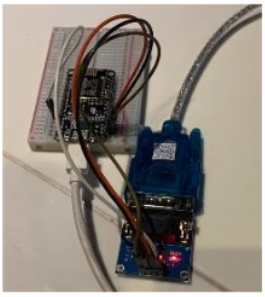

Figure 6. Payment Transaction IoT Design

Figure 5 is an RS232 image that is connected to NodeMCU using a cable with the rules as described in Table 2.

Table 2. RS232 to NodeMCU cable management

RS232	*Connected*	*NodeMCU*
GND	------→	GND
VCC	------→	3V3
RXD	------→	RX
TXD	------→	TX

Table 2 describes the cable arrangement for connecting RS232 with NodeMCU, where GND on RS232 is connected to GND on NodeMCU, VCC is connected to 3V3, RDX with RX, and TXD with TX. Data transmission from RS232 can be done in one or two directions. The data transfer rate is quite low. The maximum is only 19200 bits per second. Laptops are used to send virtual account data and amounts. The data is received by RS232 to be converted first. Then, the results are sent to NodeMCU for embedded c ++ code in which there are transaction data settings, URL end front Rest API, and API configuration. Data that has been installed in NodeMCU can be sent directly to the server. After the IoT is successfully designed, the next step is vulnerability testing on the IoT. Vulnerability testing is carried out twice. The first test is before blockchain implementation and the second test is carried out after the blockchain implementation. This vulnerability testing is carried out using XSS Vulnerability Scanning. The results of the first vulnerability testing can be seen in Figure 7.

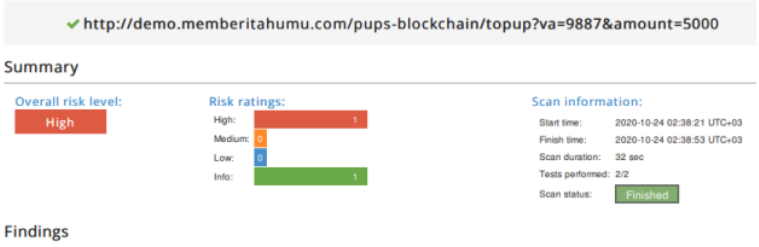

Figure 7. Vulnerability Testing Results

Figure 7 shows a summary of vulnerability testing on the webserver demo.memberumu.com/pups-blockchain. The report from the results of the vulnerability testing shows that the webserver has one XSS vulnerability which has a high level of overall risk. It shows that the security of IoT payment transactions must be improved so that it can avoid XSS attacks. The next step is the implementation of blockchain, which is carried out to increase security on IoT payments, namely API data stored on the demo.memberitahumu.com/pups-blockchain webserver. The blockchain implementation process can be seen in Figure 8.

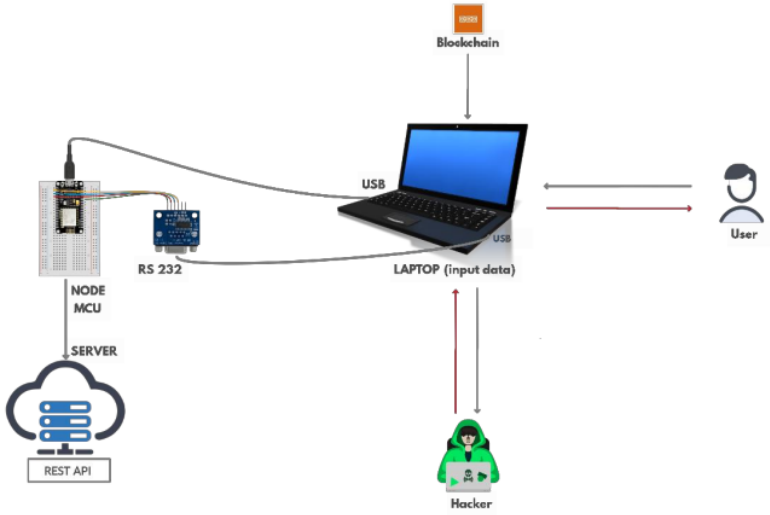

Figure 8. Blockchain Implementation

Figure 8 describes a hacker who tried to hack the Smart Payment application and the blockchain was used to overcome the attack. The hacking process is described as follows; A hacker sends a malicious script to the victim's email where the script usually contains information that is accompanied by a link that can make the victim affected click on the link if the victim is careless or not careful. When the link is clicked, it will display a form containing user data complete with data that can be used to log into the smart payment application such as username and password. If the attack is successful, the data is sent to the hacker's web so that the hacker can use it to log into the smart payment application and disguise himself as a user who was successfully hacked. In this section the hacker can make changes and steal data, so the blockchain will start working. Changes to data by hackers will initially seem successful, but because blockchain uses a decentralized concept that utilizes a consensus algorithm where all transactions will be equalized on every blockchain network data manipulation efforts are useless, whenever they do except to attack all Blockchain members. The following are the steps for creating a blockchain.

First, a block consisting of a cryptographic hash of index, timestamp, data, preceding Hash and hash is created. The nonce can be seen as program code 1.

Program code 1 *Block Structure*

```
1.    const SHA256 = require("crypto-js/sha256");
2.    class CryptoBlock {
3.    constructor(index, timestamp, data, precedingHash = " ") {
4.    this.index = index;
5.    this.timestamp = timestamp;
6.    this.data = data;
7.    this.precedingHash = precedingHash;
```

```
8.    this.hash = this.computeHash();
9.    this.nonce = 0;
10.   }
```

The constructor method contains the parameters index, timestamp, data, and precedingHash. An index is a unique number that tracks the position of each block on the entire blockchain. Timestamp keeps records of the time each completed transaction occurred. Data are completed with information about the transaction, such as details of the sender, details of the recipient, and the amount of transaction. PrecedingHash points to the hashes of previous blocks on the blockchain, which is important in maintaining blockchain integrity. The initial block on the blockchain is called a genesis block, usually having an index of 0, referring to the first block created on the network so that no previous block was assigned to it. Whenever a block is integrated with another chain, it must refer to the previous block. The integrity of the blockchain must be maintained to ensure the hash of the current block point to the hash of the previous block. After the first block is successfully created the next step is to add a new block. The previous hash and the new hash must be set to the same as the hash of the last block in the chain thus ensuring the chain is anti-broken. Because the new block's properties change with each new computer, it is important to compute the cryptographic hash again. After updating the hash, the new block is pushed into the blockchain array. The next step is to test the blockchain, namely by adding several blocks to the blockchain. Parameter data used by objects in the form of sender details, recipient details, and transacted quantities can be seen as in program code 2.

Program code 2 *Blockchain Testing*

```
1.    let smashingCoin = new CryptoBlockchain();
2.
3.    console.log("smashingCoin mining in progress....");
4.    smashingCoin.addNewBlock(
5.    new CryptoBlock(1, "01/01/2021", {
6.    sender: "Afifa Fitiya Janeeta Caharani",
7.    recipient: "Administrasi Sekolah",
8.    quantity: 0
9.    })
10.   );
11.
12.   smashingCoin.addNewBlock(
13.   new CryptoBlock(2, "01/01/2021", {
14.   sender: "Jojon Karyadi",
15.   recipient: "Administrasi Sekolah",
16.   quantity: 2900000
17.   })
18.   );
```

If the blockchain is run, it will look like display 1.

Display 1 Blockchain Testing

```
1.    {
2.    "blockchain": [
3.    {
4.    "index": 0,
5.    "timestamp": "01/01/2021",
6.    "data": "Initial Block in the Chain",
7.    "precedingHash": "0",
8.    "hash": "8f14d9b42b74dc2e6565fa35a46efada5c50ab4481d4289beceec822abf8ddf2",
9.    "nonce": 0
10.   },
11.   {
12.   "index": 1,
13.   "timestamp": "01/01/2021",
14.   "data": {
15.   "sender": "Afifa Fitiya Janeeta Caharani",
16.   "recipient": "Administrasi Sekolah",
17.   "quantity": 0
18.   },
```

```
19.   "precedingHash":
      "8f14d9b42b74dc2e6565fa35a46efada5c50ab4481d4289beceec822abf8ddf2",
20.   "hash": "0000a4243ae53c83fc455439853388115b45aa8c4f6f0a50ed00cb9a48a971a9",
21.   21.             "nonce"               : 67880
22.   22.
23.   },
24.   {
25.   "index": 2,
26.   "timestamp": "01/01/2021",
27.   "data": {
28.   "sender": "Jojon Karyadi",
29.   "recipient": "Administrasi Sekolah",
30.   "quantity": 2900000
31.   },
32.   "precedingHash":
      "0000a4243ae53c83fc455439853388115b45aa8c4f6f0a50ed00cb9a48a971a9",
33.   "hash" : "00001357b54f839572377d2c044670f0dc3c0704b2155343acb120d14e5c74d1",
34.   "nonce" : 207584
35.   },
```

Display 1 shows that each block refers to the hash of the previous block. After seeing and testing that the blockchain is functioning, it is necessary to verify the integrity of the blockchain. As it has been explained that the main characteristic of a blockchain, once a block is added to the chain, the chain cannot be changed without canceling the integrity of the other chains. Program Code 3 is a way to verify blockchain integrity in the Smart Payment application.

Program code 3 Blockchain Integrity Verification

```
1.    checkChainValidity() {
2.    for (let i = 1; i < this.blockchain.length; i++) {
3.    const currentBlock = this.blockchain[i];
4.    const precedingBlock = this.blockchain[i - 1];
5.
6.    if (currentBlock.hash !== currentBlock.computeHash()) {
7.    return false;
8.    }
9.    if (currentBlock.precedingHash !== precedingBlock.hash) return false;
10.   }
11.   return true;
12.   }
13.   }
```

Hash is very important for ensuring blockchain validity and security because any change in block content will result in the production of a new hash, invalidating the blockchain. The checkChainValidity () method uses an if-statement to verify that the hash of each block has been tampered with. Starting from the first block created then iterates over the entire blockchain and checks its validity, block genesis is not checked because it is hardcoded. This method verifies whether the hashes of any two consecutive blocks point to each other. If the blockchain's integrity has not been compromised it returns true; if not or an anomaly occurs, the value is false.

The last step taken is vulnerability testing to be carried out again after blockchain implementation is carried out. This aims to prove whether the blockchain implementation to increase IoT payment transactions has been successfully carried out. The results of this vulnerability testing can be seen in Figure 9.

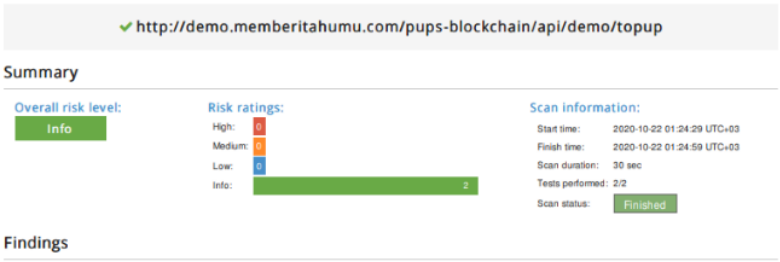

Figure 9. Vulnerability Testing Results after Blockchain Implementation

Figure 9 shows the results of vulnerability testing after blockchain implementation is carried out. The results of the vulnerability testing show the overall level of risk is the absence of vulnerabilities from XSS attacks. This shows that the optimization of IoT security for payment transactions using blockchain is a hundred percent successful.

CONCLUSIONS AND RECOMMENDATIONS

Each member on the blockchain network has a key pair to make transactions. Every data input becomes a block that is connected to become a chain. Therefore, every member is not allowed to change or delete every block of data. The chain containing interconnected block blocks will be carried out by consensus so that blocks containing invalid data can be ignored by other members and even deleted from the blockchain network automatically. Each block with a transaction value that is different from the majority value in a blockchain network will be eliminated and the value stored in each blockchain member data will be corrected. Therefore, if an attacker tries to change the transaction data on the blockchain network, it will be eliminated and the collected data that got some changes will be repaired automatically. Before blockchain implementation was carried out, the overall risk level was found 1 XSS vulnerability gap which had a high level of overall risk, then the result of vulnerability testing after blockchain implementation was not found vulnerabilities from XSS attacks. The conclusion that can be drawn from this research is that blockchain implementation can improve the security of IoT payment transactions from cross-site scripting (XSS) attacks.

REFERENCES/BIBLIOGRAPHY

A. Haryadi, H. Priyanto, and H. Anra, "Designing News Insertion Application with Internet Content Adaptation Protocol," vol. 5, no. 3, pp. 1–6, 2017. https://jurnal.untan.ac.id /index.php/justin/article/view/20575

A. Winarto, "E-Transcript Design with Blockchain Technology," *Pros. Semin. Nas. Pakar*, vol. 0, no. 0, pp. 1-37.1–1.37. 6, 2019, [Online]. Available: https://www.trijurnal.lemlit.trisakti.ac.id/ pakar/article/view/4176%0Ahttps://www.trijurnal.lemlit.trisakti.ac.id/pakar/article/view/4176/331 6.

A. Y. W, Yunanri, Imama Riadi, "Vulnerability Detection Analysis on Open Journal System Web Server Using OWASP Scanner," *Jurnal Rekayasa Teknologi Informasi (JURTI)*, vol. 2, no. 1. pp. 1–8, 2018. http://e-journals.unmul.ac.id/index.php/INF/article/view/1319

D. Sasmoko and Y. A. Wicaksono, "Internet of Things (IoT) Implementation in Infusion Monitoring Using ESP 8266 and WEB to Share Data," J. Ilm. Inform., vol. 2, no. 1, pp. 90–98, 2017, doi: 10.35316/jimi.v2i1.458. https://ejournal.amiki.ac.id/index.php/JIMI/article/view/36/21

D. Setiadi and M. N. Abdul Muhaemin, "Application of the Internet of Things (IoT) in the Irrigation Monitoring System (Smart Irrigation)," Infotronik J. Teknol. Inf. dan Elektron., vol. 3, no. 2, p. 95, 2018, doi: 10.32897/infotronik.2018.3.2.108. http://jurnal.usbypkp.ac.id/index.php/infotronik/article/download/108/93

F. Rozi, H. Amnur, F. Fitriani, and P. Primawati, "Home Security Using Arduino Based on the Internet Of Things," INVOTEK J. Inov. Vokasional dan Teknol., vol. 18, no. 2, pp. 17–24, 2018, doi: 10.24036/invotek.v18i2.287. http://invotek.ppj.unp.ac.id/index.php/invotek/article/view/287/72

I. Busthomi, I. Riadi, R. Umar, and J. P. Soepomo, "Optimization of Event Registration Information Security Using Blockchain Technology," vol. XII, no. 1, pp. 74–82, 2020. https://www.researchgate.net/publication/342200694_Optimasi_Keamanan_Autentikasi_dari_Man_in_the_Middle_Attack_MiTM_Menggunakan_Teknologi_Blockchain

I.M Suartana, H. Endah Wahanani, and A. Noor Sandy, "Web Server Security System With Application Firewall (WAF)," Scan, vol. X, no. 1, pp. 3–8, 2015. https://www.researchgate.net/publication/347328470_Implementasi_Web_Application_Firewall_Menggunakan_Modsecurity_Sebagai_Strategi_Pengamanan_Web_Server/link/5fd9dbf545851553a0bd813b/download

I. Riadi et al., "Cross-Site Scripting (XSS) Attack Vulnerability Analysis on Smart Payment Applications Using the OWASP Framework," vol. 5, no. 3, pp. 146–152, 2020. https://www.researchgate.net/publication/345830787_Analisis_Kerentanan_Serangan_Cross_Site_Scripting_XSS_pada_Aplikasi_Smart_Payment_Menggunakan_Framework_OWASP

J. Fat, H. Candra, and W. Wiliam, "Sensor Data Securitization in the Internet of Things (IoT) Applications Using the Ethereum Blockchain on the Testnet Network," TESLA J. Tek. Elektro, vol. 21, no. 1, p. 79, 2019, doi: 10.24912/tesla.v21i1.5886. https://www.neliti.com/id/publications/296798/sekuritisasi-data-sensor-pada-aplikasi-internet-of-things-iot-dengan-menggunakan

L. Arief and T. A. Sundara, "Study on the Use of Blockchain for the Internet of Things (IoT), "J. RESTI (Engineering Systems and Information Technology), vol. 1, no. 1, p. 70, 2017, doi: 10.29207/resti.v1i1.26. https://www.researchgate.net/publication/321798741_Studi_atas_Pemanfaatan_Blockchain_bagi_Internet_of_Things_IoT/link/5cded3a2299bf14d95a2b834/download

P. J. Taylor, T. Dargahi, A. Dehghantanha, R. M. Parizi, and K. K. R. Choo, "A systematic literature review of blockchain cyber security," Digit. Commun. Networks, vol. 6, no. 2, pp. 147–156, 2020, doi: 10.1016/j.dcan.2019.01.005. https://www.sciencedirect.com/science/article/pii/S2352864818301536

R. Cole, M. Stevenson, and J. Aitken, "Blockchain technology: implications for operations and supply chain management," Supply Chain Manag., vol. 24, no. 4, pp. 469–483, 2019, doi: 10.1108/SCM-09-2018-0309. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7522652/pdf/main.pdf

Rusdiana, C. Banta, and Sanusi, "Website Security Analysis Against Cross-Site Request Forgery (CSRF) Attacks," KANDIDATJurnal Ris. dan Inov. Pendidik., vol. 1, no. 1, pp. 21–29, 2019. http://jurnal.abulyatama.ac.id/index.php/kandidat/article/view/328/pdf_1

S. D. K. Hu, H. N. Palit, and A. Handojo, "Implementasi Blockchain: Studi Kasus e-Voting," J. Infra, vol. 7, no. 1, pp. 183–189, 2019. https://trijurnal.lemlit.trisakti.ac.id/pakar/article/view/4176

S. S. H. Putra, "Countermeasures for XSS Attacks, CSRF, SQL Injection Using Blackbox Methods on the IVENMU Marketplace," J. Pendidik. dan Teknol. Inf., vol. 4, no. 2, pp. 289–300, 2017. http://lppm.upiyptk.ac.id/PTI/index.php/pti/article/view/75/51

Sunardi, I. Riadi, and P. A. Raharja, "Vulnerability analysis of E-voting application using open web application security project (OWASP) framework," Int. J. Adv. Comput. Sci. Appl., vol. 10, no. 11, pp. 135–143, 2019, doi: 10.14569/IJACSA.2019.0101118. https://thesai.org/Publications/ViewPaper?Volume=10&Issue=11&Code=IJACSA&SerialNo=18

T. A. S. Lathifah Arief, "Jurnal Resti," Study on the Usability of Blockchain for the Internet of Things, vol. 1, no. 1, pp. 19–25, 2017. https://www.neliti.com/id/publications/240112/studi-atas-pemanfaatan-blockchain-bagi-internet-of-things-iot

U. Rahardja, Q. Aini, M. Yusup, and A. Edliyanti, "Application of Blockchain Technology as a Media for Securing E-Commerce Transaction Processes," CESS (Journal Comput. Eng. Syst. Sci., vol. 5, no. 1, p. 28, 2020, doi: 10.24114/cess.v5i1.14893. https://www.researchgate.net/publication/342955653_Penerapan_Teknologi_Blockchain_Sebagai_Media_Pengamanan_Proses_Transaksi_E-Commerce

Y. Yulianingsih, "Protecting Applications from Cross-Site Scripting Attacks with the Metacharacter Method," J. Nas. Teknol. dan Sist. Inf., vol. 3, no. 1, pp. 83–88, 2017, doi: 10.25077/teknosi.v3i1.2017.83-88. https://www.researchgate.net/publication/317113534_Melindungi_Aplikasi_dari_Serangan_Cross_Site_Scripting_dengan_Metode_Metacharacter

HASIL CEK_Smart Payment Application Security Optimization from Cross-Site Scripting (XSS) Attacks Based on Blockchain Technology

ORIGINALITY REPORT

9%	5%	2%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

4%

★ www.researchgate.net

Internet Source

Exclude quotes	On	Exclude matches	< 2%
Exclude bibliography	On		