

## Research Article

# Security System Analysis Using the HTTP Protocol Against Packet Sniffing Attacks

<sup>1st</sup>Inez Sri Wahyuningsi Manguling, <sup>2nd</sup>Jumadi M Parenreng 

<sup>1,2</sup>Department of Computer Engineering, Universitas Negeri Makassar, Makassar 90222, South of Sulawesi, Indonesia

\*Corresponding Author: [jparenreng@unm.ac.id](mailto:jparenreng@unm.ac.id)

## Abstract:

The security level of the SIM MBKM website information system needs to be analyzed because it is accessed by many students who provide essential data. The security testing process for the SIM MBKM system uses the software Ettercap and Wireshark to test the system's security level and network data against cybercrime attacks. The results of these experiments showed the same essential data, but Wireshark displayed more personal information. The difference lies in the system architecture using different methods or stages, namely ARP Poisoning and Filtering HTTP. The third difference is the estimated time taken during the experiment. Ettercap and Wireshark applications enable eavesdropping on confidential and essential data and information. Through security testing using Ettercap and Wireshark, more critical data is displayed. solutions and preventive actions can be implemented by encrypting confidential data and improving website security using the HTTPS (Hypertext Transfer Protocol Secure) protocol.

**Keywords:** HTTP Protocol, Sniffing, ARP Poisoning, Ettercap, Wireshark



**Citation:** I.S.W.Manguling, J.M.Parenreng "Security System Analysis Using the HTTP Protocol Against Packet Sniffing Attacks". *Iota*, 2023, ISSN 2774-4353, Vol.03, 04.  
<https://doi.org/10.31763/iota.v3i4.612>

Academic Editor : Adi, P.D.P

Received : August, 07 2023

Accepted : September, 18 2023

Published : November, 09 2023

**Publisher's Note:** ASCEE stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2023 by authors.  
Licensee ASCEE, Indonesia. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution Share Alike (CC BY SA) license(<https://creativecommons.org/licenses/by-sa/4.0/>)

## 1. INTRODUCTION

The development of technology in the present era is rapidly influencing human life and activities. The story of the technological age cannot be separated from the use of the Internet as a provider or medium for using smartphones and computers. The use of the Internet and technology facilitates fast communication and information dissemination. With the Internet, accessing applications and websites is easily obtained to search for information on the provided platforms and exchange or send important information through the Internet [1].

As internet and technology use continues to grow, it hurts cybercrime. Internet access becomes one of the threats to system and network crimes by disrupting network traffic, leading to information and data loss [2]. Various protocols commonly used in networks and the Internet include TCP/IP, HTTP, SSL, HTTPS, Ethernet, FTP, RTP, DNS, IMAP, SSH, ICMP, and UDP [3]. The HTTP protocol is commonly used to access websites or web-based platforms. The security of website-based systems is one aspect that needs to be considered because securing websites means ensuring that user administrators are more secure.

However, security levels must continuously be monitored and updated because the ease of accessing the Internet through website protocols has security vulnerabilities that make it susceptible to attacks. One of the applied attacks to test websites is the HTTP protocol [4]. Sniffing techniques become dangerous when network traffic using the HTTP (Hypertext Transfer Protocol) is intercepted [1]. One system crime that occurs through websites is eavesdropping, where important individual or company data is

obtained. This problem is commonly referred to as sniffing [5]. Sniffing is a form of computer crime where criminals steal and obtain essential data such as usernames and passwords, which are then used for fraud using the victim's identity. Sniffing is a technique used by unauthorized parties to retrieve sensitive data by reading data packets sent over the network [5].

SIM MBKM is used to provide important information regarding the policy program of "*kampus Merdeka*" (independent campus) under the auspices of the Ministry of Education, Culture, Research, and Technology (KEMENDIKBUD-RISTEK). SIM MBKM is intended for students [6] who will participate in the "*Kampus Merdeka*" program by uploading necessary documents, and, of course, accessing it requires logging in using a username and password. Therefore, the security level of the SIM MBKM website information system needs to be analyzed. The security testing process for the SIM MBKM system uses Ettercap and Wireshark software to test the system's security level and network data against cybercrime attacks.

Moreover, Ettercap is an application that includes hacking systems or well-known sniffer applications [7]. Wireshark is an application that understands the structure of protocols and monitors packets [8], allowing the data to be read and eventually identifying personal data based on the SIM MBKM website system [9]. Based on the background above, this research aims to analyze the security of a system using the HTTP protocol to test the security of data and systems in network activities to detect sensitive data being monitored and prevent it from becoming a source of packet sniffing attacks.

## 2. THEORY

System security is crucial nowadays due to the increasing incidents of data theft from internet users, which are then used for malicious purposes. Therefore, one of the solutions to prevent data theft is network protection. Data theft by hackers poses a severe problem for operators or administrators when restoring websites, as many websites lack secure system security standards, making them vulnerable to hacker attacks [10]. Website system security is essential for information system administrators to control access to websites and prevent unauthorized source misuse [11].

The HTTP protocol is a protocol that facilitates communication between clients and web servers [5]. HTTP is a text-based protocol based on a request-response system. The client sends a request to the service provider (server), which receives a response [12]. Sniffing techniques are applied to HTTP protocol traffic using Wireshark software. Wireshark collects data from wireless data transmissions, filters the data, and focuses only on HTTP protocol data. The HTTP protocol carries POST data that contains important information such as usernames and passwords [1]. The testing involves sniffing, user login, password hacking, and MAC address duplication or ARP spoofing [6].

Sniffing techniques (ARP spoofing) involve sending false or modified ARP packets with the attacker's email address to forge the victim's ARP cache table. ARP spoofing attacks are dangerous as they can track the victim's browser searches and steal social media, office, and other account login credentials. This attack supports other computer network attacks, such as denial of service, man-in-the-middle attacks, identity theft, and more [13]. Evidence of ARP spoofing attacks includes the MAC addresses of the attacker and victim, as well as the timing of the attack, obtained using Wireshark and Ettercap tools to analyze network traffic, particularly using the ARP protocol [14]. ARP spoofing attacks can lead to other attacks, such as denial of service and man-in-the-middle attacks, which prevent users from accessing the network and result in data theft [14].

The final analyzed result involves finding evidence of man-in-the-middle attacks based on ARP poisoning of traffic [15] and the HTTP filtering stage in the Wireshark

application. This can be done to gather evidence and information about the perpetrators for further consideration. Attackers employ techniques to capture data frames on the local network and then modify or disrupt data traffic [16]. Wireshark is a network packet analyzer that attempts to capture data packets and display as much information as possible from those packets [17].

Wireshark is a commonly used software for viewing and capturing network packets, displaying all the detailed information contained within [18]. In the process of sniffing, Wireshark and Ettercap are used to capture traffic and obtain usernames and passwords from a station [11]. Wireshark can analyze packets in real time or display them live, meaning the application observes and displays all packet data [11]. The investigated data is temporary and can only be found in random access memory or network traffic. Therefore, the attacker's behavior and digital evidence can be identified in the form of IP addresses and target source MAC addresses [7].

### 3. METHOD

#### A. Research Flow

The research methodology applied in this study involves several stages to achieve the research objectives. The study analyzes the security of a website system by testing its security using two applications, Ettercap and Wireshark. Moreover, several steps are necessary to ensure the smooth progress of the research. The analysis begins with problem identification, which aims to determine the issues that will be addressed in the study. The problem identification has been outlined in the introduction, which focuses on testing the security level of the targeted MBKM SIM system using Ettercap and Wireshark applications. Additionally, it is identified that the leaked data resulting from the security testing will be displayed. The next step involves collecting data from previous studies based on a literature review to compare the findings and processes of previous research. The literature review stage is crucial in providing supporting data and discussions related to the explanation of each variable to be discussed in this study.

The planning and preparation stage is necessary to make plans and prepare everything needed for the security system testing. By planning and ensuring research requirements, the performance testing process can be conducted accurately, and data analysis can be performed correctly and objectively. The planning and preparation of the security system testing are designed in terms of the tools and materials used, such as hardware, software, and HTTP protocol websites. This is followed by creating a system architecture to understand the workflow of each application. The system architecture design explains the workflow of Ettercap and Wireshark applications, which have different methods of testing the system's security.

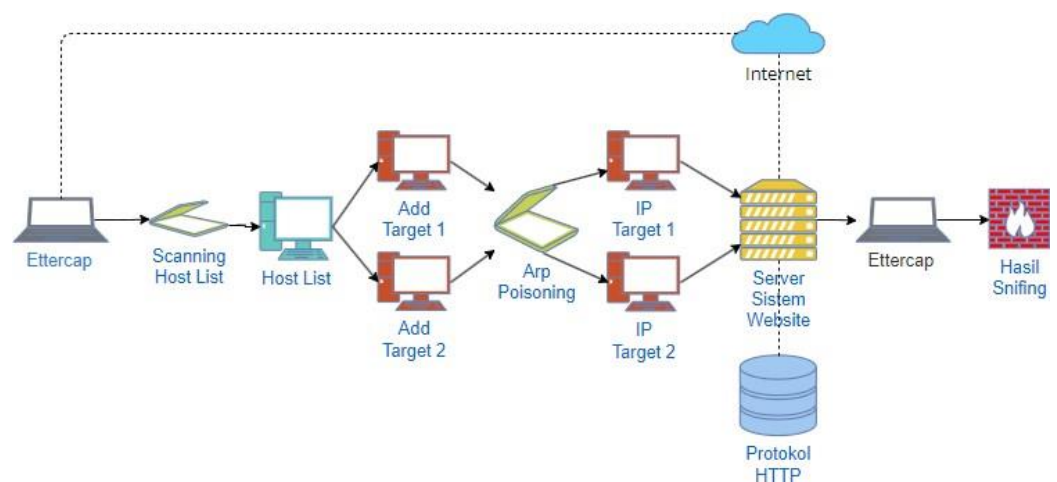
The security testing stage involves testing the system security of the selected website, "*sim.mbkm.unm.ac.id*," as the target. The designated website for security testing will be tested using Ettercap and Wireshark applications. The Ettercap testing is conducted based on Kali Linux [19] on VirtualBox, while Wireshark is based on Windows 10. The data analysis stage displays the information obtained by each application, Ettercap, and Wireshark, which includes sensitive or personal data of users accessing the system website. This data will be analyzed by comparing the performance results of both

applications. This stage concludes the data analysis by comparing the results obtained from each application and analyzing the differences effectively based on the workflow during the system testing.

### B. Ettercap System Architecture

The system security testing is conducted using the Ettercap application based on Kali Linux, installed on VirtualBox. In Figure 1, the first step is scanning the entire host list. The host list is monitored to detect the available hosts that will be used as target IPs. Once the available host list is detected, a host list is created. The host list is necessary to view the available host list or IPs that will be used as target 1 and target 2. Once all the IP addresses are available, two IP addresses are selected, each designated as Target 1 and Target 2. Afterward, the Arp Poisoning stage is conducted.

Arp poisoning is a technique where the perpetrator's MAC address is connected to a valid IP address so that they can steal or modify the victim's data. This technique is commonly used to attack a network [20]. Subsequently, a website from the system is selected for experimentation and system security testing. The system website uses the HTTP protocol and is connected through internet network access. Finally, the results of the security testing are analyzed.



**Fig.1.** Ettercap System Architecture

Furthermore, The Wireshark application performs security testing on a website system based on the HTTP protocol. Figure 2 illustrates the security testing using Wireshark by first opening the website system portal to be tested, which operates on the HTTP protocol and is accessed through the Internet. Then, a login is performed on the website system. Afterward, the security testing is conducted using two methods. The first method involves identifying the IP addresses displayed based on the name of the tested website. The second method involves filtering the HTTP protocol. Once the HTTP protocol is identified, the website system will be displayed, showing sensitive data within the system that undergoes security testing. The data displayed in Wireshark is based on

the trial of the website system using the HTTP protocol. If an IP address is identified as the victim, the data obtained from the security testing is analyzed.

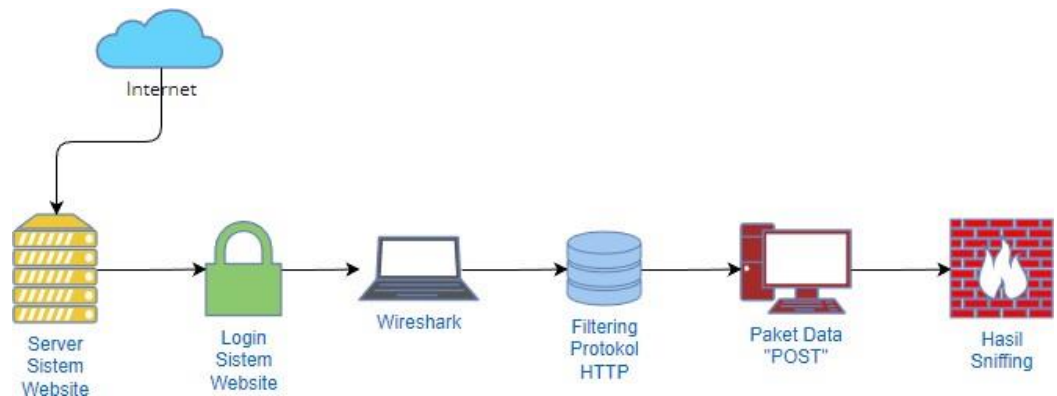


Fig.2. Wireshark System Architecture

#### 4. RESULT AND DISCUSSION

Security testing on the SIM MBKM (Independent Campus Management Information System) website is conducted to assess the level of system and network security. The captured data packets [21] are filtered to focus on capturing HTTP protocol packets that contain information about personal or sensitive data being transmitted over the Internet [22].

This analysis and testing are necessary to understand the nature of data and information displayed and distributed, which can lead to and enable packet sniffing attacks on the HTTP network protocol. Therefore, it is essential to reevaluate the website's security level against packet sniffing attacks [23]. Sniffing is highly dangerous if users unknowingly input sensitive data on websites that use the HTTP protocol for data communication. In this experiment, sniffing will be performed using two applications, Ettercap and Wireshark, which are selected for the security testing of the system.

##### A. Ettercap Testing Scenario

The testing scenario is implemented directly in the security testing applications Ettercap and Wireshark. The testing scenario in this research is divided into two stages of experimentation: the first stage is tested using the Ettercap application, and the second stage is tested using the Wireshark application. The Ettercap testing scenario consists of several stages to obtain important data from the SIM MBKM website. The testing scenario stages can be seen in the following diagram.

##### B. Host List Scanning Phase

In Figure 3, three hosts are visible after scanning in Ettercap. The host list is scanned in Ettercap to find available hosts on the system and network devices before selecting them as targets. Scanning the host list is part of the data collection process used by attackers as a shortcut to create and assess the profile of the organization that will be targeted. This stage serves as an initial step to detect displayed host lists and gather

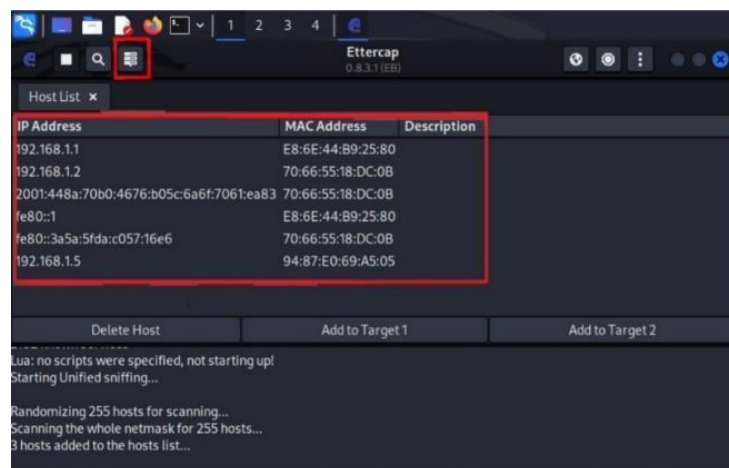
sensitive, private, or important information that can be used for further attacks. Scanning is the most common data collection technique involving discovering architecture and using information services. The Ettercap application provides various scanning methods that can be used to group multiple target IP addresses for scanning. However, the downside is that the scan results for hosts cannot be provided until the entire batch scanning process is completed [24].



**Fig.3.** Host List Scanning Process

### C. Host List Phase

The displayed host list is the result of scanning the host list. Figure 4 shows the IP addresses registered in the Ettercap application after scanning the host list. There are three IP addresses: 192.168.1.1, 192.168.1.2, and 192.168.1.5. In the next stage, two IP addresses will be selected among these three host lists: Target 1 and Target 2. This stage determines the IP addresses that will be used as targets.



**Fig.4.** Host List Display

### D. IP Address as a Target Phase

In Figure 5, the available IP addresses are displayed, and among the three scanned IP addresses, 192.168.1.2 and 192.168.1.5 are selected as the target. The host list display will be used as the target before performing ARP poisoning. Target 1 is chosen based on the IP address present in the Kali Linux operating system by checking the Kali Linux terminal

using the command "ifconfig." The selected IP address for Target 1 is 192.168.1.2, and Target 2 is 192.168.1.5, which will serve as the victim's IP address.

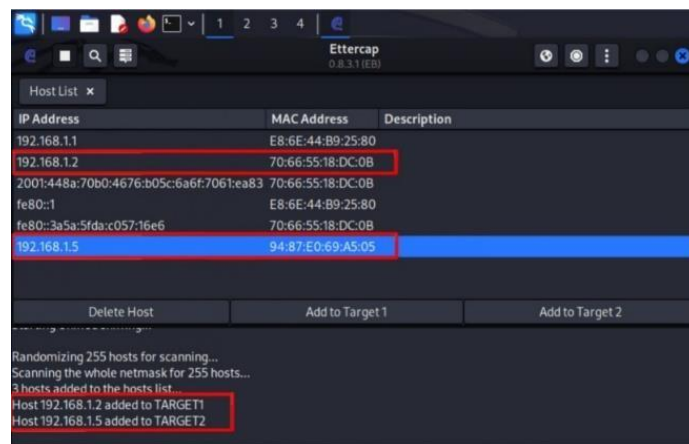


Fig.5. Adding IP as Target

### E. Arp Poisoning Phase

Figure 6 shows that ARP (Address Resolution Protocol) poisoning is performed after determining the IP addresses as Target 1 and Target 2, ARP (Address Resolution Protocol) poisoning is performed. Attackers utilize ARP poisoning through ARP spoofing techniques, where they connect the MAC address of the malicious actor to a legitimate IP address to steal or modify data from the victim. The attacker manipulates the network by sending fake ARP packets, causing the ARP table to be overwritten with the fake ARP entries sent by the attacker. After this step, the ARP poisoning phase is successfully executed, with each ARP poisoning targeting IP address Target 1, which is 192.168.1.2, in Group 1. In contrast, the subsequent ARP poisoning targets IP address Target 2, which is 192.168.1.5, in Group 2.

Furthermore, ARP spoofing works by reporting false information in the ARP message to the target computer. By sending fake ARP packets, the attacker can deceive the victim's computer when it is in front of it, thereby expediting the translation process. ARP typically maintains a simple database table that combines and links MAC addresses and IP information when a hotspot client is affected by ARP spoofing, where the client's PC's MAC address matches the intruder's MAC address [25].

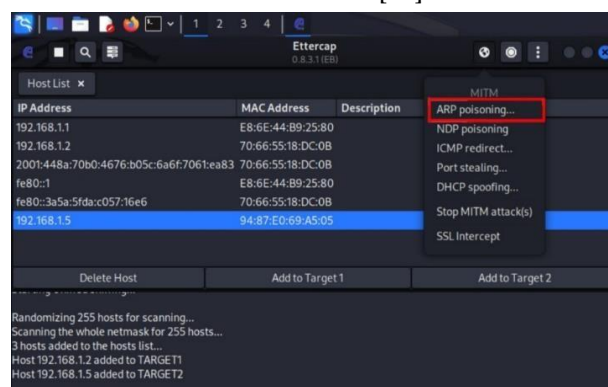
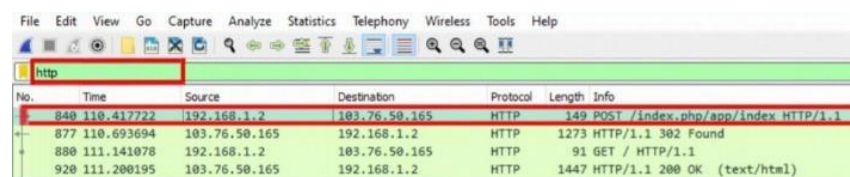


Fig.6. Arp Poisoning Process



### F. Wireshark Testing Scenario

The Wireshark testing scenario consists of several stages to obtain essential data from the SIM MBKM website. The testing scenario stages can be observed in the following diagram. The Wireshark device captures and records the communication within the system while logging into the website. Due to the large volume of data packets, filtering is performed to display only the HTTP protocol in the packet list window. Filtering is used to facilitate the viewing of protocols that specifically involve HTTP. Given the experiment's utilization of the HTTP protocol, the filtering performed in the Wireshark application is based on the HTTP protocol. Moreover, Figure 7 shows numerous IP addresses and HTTP protocol packets. Based on the experiment, data labeled "POST" is visible. The "POST" information column transmits data or values to another page for processing and observing any leaked data.



No.	Time	Source	Destination	Protocol	Length	Info
840	110.417722	192.168.1.2	103.76.50.165	HTTP	149	POST /index.php/app/index HTTP/1.1
877	110.693694	103.76.50.165	192.168.1.2	HTTP	1273	HTTP/1.1 302 Found
880	111.141078	192.168.1.2	103.76.50.165	HTTP	91	GET / HTTP/1.1
920	111.200195	103.76.50.165	192.168.1.2	HTTP	1447	HTTP/1.1 200 OK (text/html)

Fig. 7. HTTP Filtering

### G. HTTP Stream Analysis Phase

This stage is the phase where the leaked data is revealed. The Wireshark application can be used in two ways to view the leaked data. The first method is through HTTP Stream, as shown in Figure 8. To perform HTTP Stream, you need to select the "analyze" menu, then choose "follow," and finally select "HTTP stream" because the experiment conducted on the SIM MBKM website is based on the HTTP protocol. The second method involves directly double-clicking or double-clicking on the "POST" column to display the results of the security testing. After applying protocol filtering, the next step is to search for the information displayed in the "info" column. Before viewing the report, an HTTP stream is performed on the previously selected protocol. The HTTP stream function allows for a direct view of the data flow that occurs in the HTTP protocol during data transmission when logging into the SIM MBKM website.

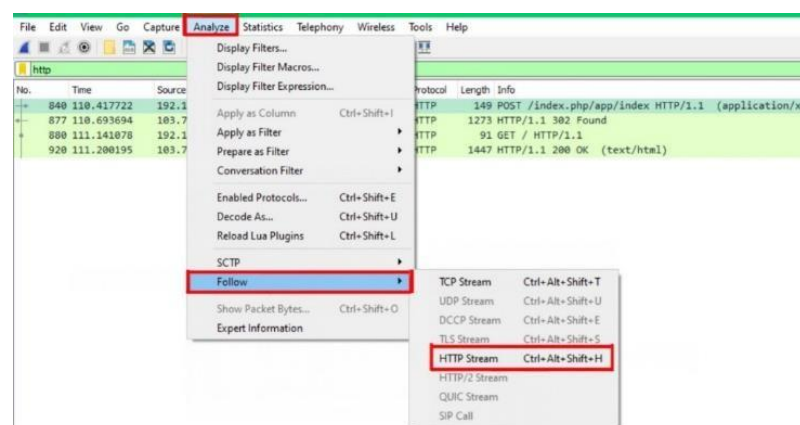


Fig.8. HTTP Stream Process



### H. Testing Environment

The security testing of SIM MBKM requires system requirements for conducting experiments with the Ettercap and Wireshark applications. Table 1 displays the hardware and software system requirements for security testing. The hardware requirement for the testing involves using an Asus laptop with an AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G 3.10 GHz processor, and 4 GB of memory. The computer runs on the Windows 10 Home Single Language operating system, version 21H2.

The software requirement for the testing involves using the Ettercap application accessed through VirtualBox with the Kali Linux operating system. The Wireshark application is accessed using Windows 10, and the target for the experiment is one of the information systems owned by the State University of Makassar (UNM). The security testing is performed on the SIM MBKM website, a management information system under the State University of Makassar (UNM). SIM MBKM is a website-based information system that utilizes the HTTP protocol. SIM MBKM serves as a platform for providing necessary information related to the policies of the Kampus Merdeka program, which is under the Ministry of Education, Culture, Research, and Technology (KEMENDIKBUD-RISTEK).

**Table 1.** System Requirements

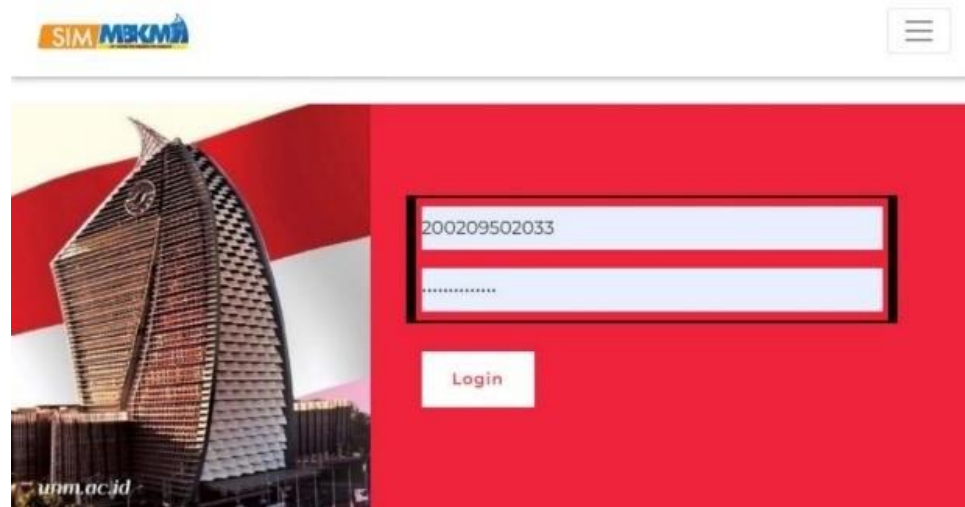
1.	Hardware	Laptop ASUS
	<ul style="list-style-type: none"> <li>AMD A9-9425 RADEON R5, 5</li> <li>COMPUTE CORES 2C+3G 3.10 GHz</li> <li>4.00 GB (3.87 GB usable)</li> <li>Windows 10 Home Single Language</li> <li>Version 21H2</li> </ul>	
2.	Software	VirtualBox Kali Linux
	<ul style="list-style-type: none"> <li>Wireshark</li> <li>Website <a href="http://sim.mbk.m.unm.ac.id/">http://sim.mbk.m.unm.ac.id/</a></li> </ul>	

### I. Ettercap Testing

The first security testing on the official SIM MBKM website system is performed using the Ettercap application on the Kali Linux operating system. This choice is made because it allows for conducting attacks on websites and other sites and provides the tools and features required for hacking techniques.

The security testing of the "*sim.mbk.m.unm.ac.id*" website system will be conducted on the *Merdeka Belajar* Management Information System, one of the information systems under the authority of Makassar State University. SIM MBKM is a management information system managed by Makassar State University, which facilitates students in accessing information and uploading files related to the *Kampus Merdeka* program under the Ministry of Education, Culture, Research, and Technology (KEMENDIKBUD-RISTEK). Therefore, security testing is needed on this system to identify potential data leaks using the Ettercap application. The security testing of the procedure involves opening a browser that is targeted as the victim. The SIM MBKM information system uses HTTP as its security protocol. The security testing of the system can be seen in Figure 9,

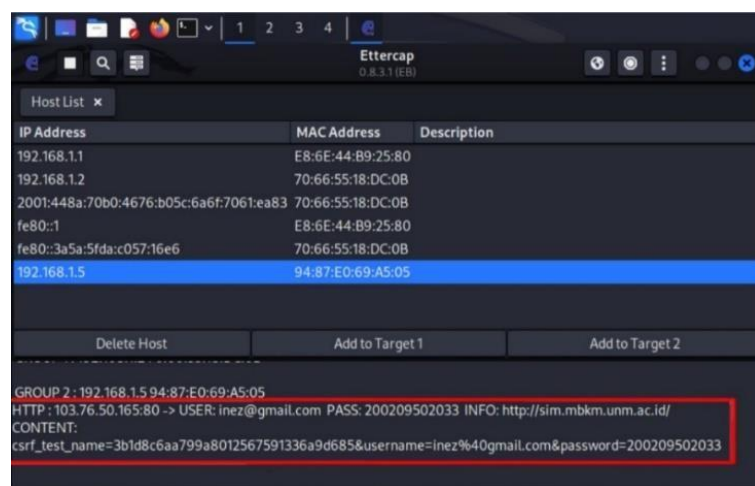
where an attempt is made to log in to the "*sim.mbkm.unm.ac.id*" website using the username "*inez@gmail.com*" and the password "*200209502033*".



**Fig.9.** Website System Testing

#### **H. Security Testing Results in Ettercap**

Essential data such as usernames and passwords used during login are among the many sensitive pieces of information held by individuals. After entering the username and password accessed through the "*sim.mbkm.unm.ac.id*" website using the Windows 10 operating system, the focus then shifts back to the Ettercap application, which is based on the Kali Linux operating system. Data leaks are displayed at the bottom of the application after entering sensitive information on the SIM MBKM website system, as shown in Figure 10. The leaked data includes the username "*inez@gmail.com*" and the password "*200209502033*".

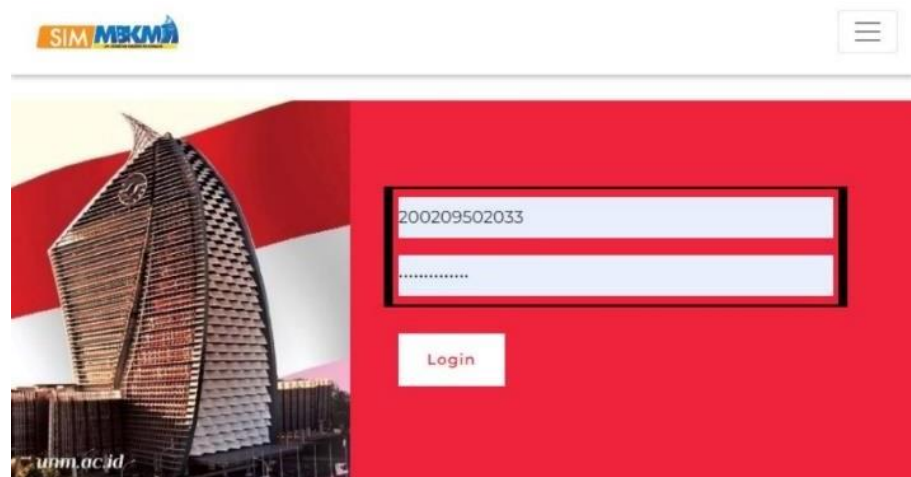


**Fig.10.** Security Testing Results in Ettercap

#### **I. Wireshark Testing**

The second security testing is conducted using the Wireshark application based on the Windows 10 operating system. The targeted research sample is the information system of SIM MBKM, which Makassar State University owns. SIM MBKM is a

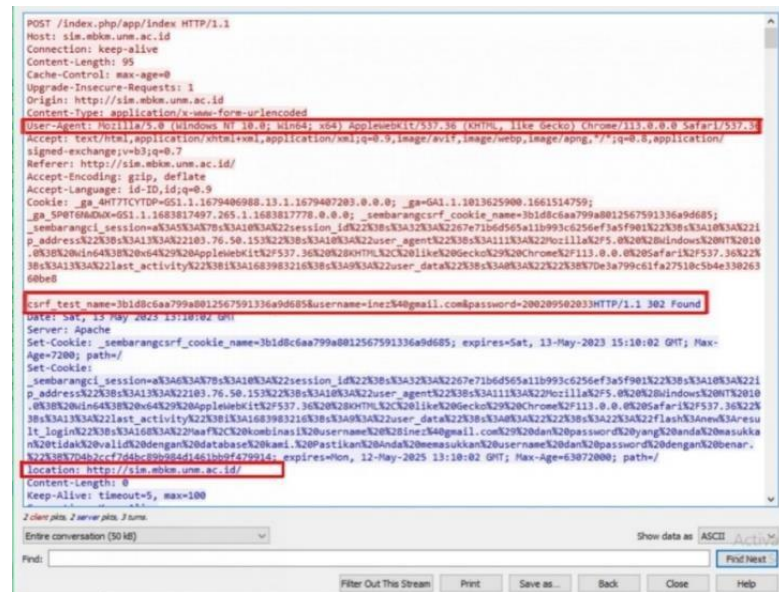
management information system for independent learning. It is chosen as the subject of security testing for its widespread use by students at Makassar State University (UNM) as part of the "*belajar diluar kampus*" (learning beyond campus) program, which falls under the Ministry of Education, Culture, Research, and Technology. Due to the high number of student access and the input of various documents into the system, security testing is conducted. Wireshark is used as a tool for security testing to detect any data leaks or essential information based on the SIM MBKM website, which uses the HTTP security protocol. Wireshark is an application and tool to analyze data packets and provide detailed information within those packets, including any potentially sensitive data [17]. In the second experiment, Wireshark was used on the SIM MBKM system accessed through a browser. Figure 11 shows the testing phase, which involves logging in with a username and entering sensitive and private data passwords. The official SIM MBKM website is accessed in the image, and the username "inez@gmail.com" and password "200209502033" are entered.



**Fig.11.** SIM MBKM System Website

#### ***J. HTTP Stream Analysis Results Phase***

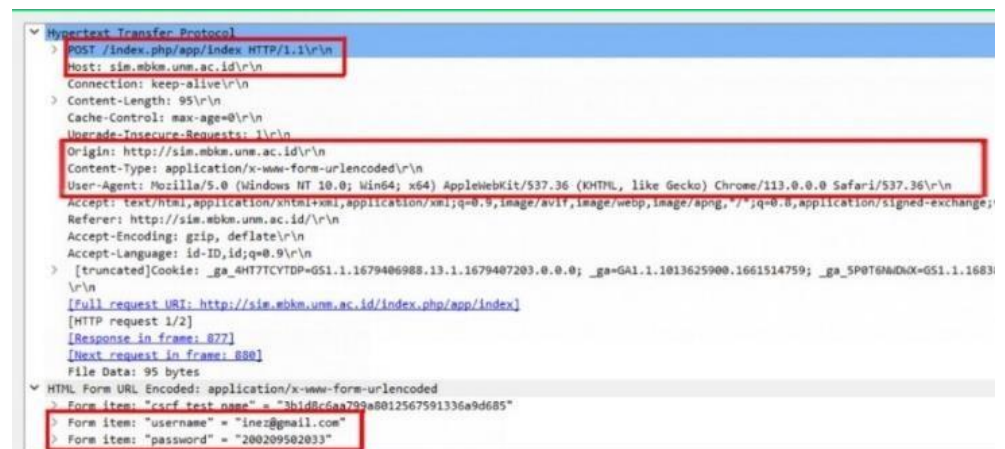
The website host and other information are displayed in this HTTP stream, as seen in Figure 12. This data is essential because it can be read and known when accessing the website using the operating system and browser application. The username and password data are the most critical information that is exposed and displayed. By observing the HTTP stream, you can see colored blocks of text in red and blue. The red blocks represent data sent from the website to the web server, while the blue blocks represent the responses from the web server displayed on the website.



**Fig.12. HTTP Stream Results**

### K. Security Testing Results in Wireshark

The results displayed in Figure 12, using the HTTP stream, are not significantly different from Figure 13, which uses the second method of double-clicking on the "POST" column. The exposure of such sensitive data, displaying the username and password, can potentially trigger data misuse and pose security threats to the website's system.



**Fig.13.** Security Testing Results in Wireshark

The security testing of the system was targeted at the SIM MBKM (Merdeka Learning Management Information System) website, which falls under the authority of Makassar State University. SIM MBKM is one of the programs implemented to comply with the Kampus Merdeka (Independent Campus) policy initiated by KEMENDIKBUD-RISTEK (Ministry of Education, Culture, Research, and Technology). SIM MBKM is an information system that provides updates and allows students to upload files. Users must log in and provide personal data such as usernames and passwords to access the system. The testing of SIM MBKM using the Ettercap and Wireshark applications revealed that the website's security is compromised and deemed insecure.

**Table 2.** Security Testing Results

Application	Type Data	Description
<b>Ettercap</b>	Username	inez@gmail.com
	Password	200209502033
	Info	http://sim.mbkunm.ac.id/
	Host	sim.mbkunm.ac.id\r\n
	Connection	keep-alive\r\n
	Content-type	text/html; charset=UTF-8
	User-agent	Mozilla/5.0 (WindowsNT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36\r\n
<b>Wireshark</b>	Accept-language	id-ID,id;q=0.9
	Username	inez@gmail.com
	Password	200209502033
	Date	Sat, 13 May 2023 13:10:02 GMT
	Server	Apache
	location	<a href="http://sim.mbkunm.ac.id">http://sim.mbkunm.ac.id</a>

The first difference between security testing using Ettercap and Wireshark is the type of data displayed in the test results, as shown in Table 2. Both applications display personal data such as usernames and passwords, but the information is different. In the Ettercap application, only the website link, username, and password are displayed. On the other hand, Wireshark shows a more extensive set of personal data. The data displayed in the security testing using Wireshark includes the host, connection details, content type, user agent, accept language, username, password, date, server, and website location.

**Table 3.** Comparison of Security Testing Results

	Ettercap	Wireshark	Description
<b>SIM MBKM</b>	ARP	Filtering	Architecture
	Poisoning	HTTP	System
	Longer Estimated Testing Time (VirtualBox)	Faster Estimated Testing Time	Time

Table 3 shows the second difference in the system architecture between ettercap and Wireshark. The Ettercap application performs ARP poisoning, while the Wireshark application requires protocol filtering. ARP poisoning in Ettercap is done before entering data on the SIM MBKM website, whereas Wireshark requires protocol filtering after entering data on the SIM MBKM website. ARP poisoning is performed to connect the perpetrator's MAC address to a legitimate IP address, allowing them to steal or modify the victim's data. Protocol filtering is essential because of the high volume of data packets, so filtering is necessary to display only HTTP protocol data and facilitate the analysis of leaked data through Wireshark identification.

Table 3 also shows the third difference in terms of the time taken from preparation to analyzing the scattered or displayed data. The preparation for security testing involves using Windows 10 as the hardware and Wireshark and VirtualBox as the software. The Wireshark application is accessed directly using Windows 10, while Ettercap needs to be accessed through VirtualBox with the Kali Linux operating system. Therefore, performing system security testing with Ettercap.

**Table 4.** Testing Time Estimation

	Ettercap	Wireshark
<b>Time</b>	03:59	01:05

Table 4 shows the estimated time required to perform security testing using the Ettercap application. Initially, it needs to be accessed through VirtualBox, and it takes a relatively long time to start the testing from the beginning to the end, which requires 03 minutes and 59 seconds. On the other hand, the Wireshark application is accessed directly on Windows 10, and the testing can be completed in 01 minute and 05 seconds. Therefore, it can be concluded that the estimated time for security testing using the Wireshark application is faster than the Ettercap application.

## 5. CONCLUSION

Based on the experiment results and discussions regarding the analysis of system security using the HTTP protocol against packet sniffing attacks, the following conclusions can be drawn: [1] The HTTP protocol becomes highly dangerous when used and accessed in a public manner to transmit or input sensitive and confidential data and information such as usernames and passwords. [2] Ettercap and Wireshark applications enable the interception of sensitive and essential data and communication by capturing data through the Wireshark and Ettercap applications. Sniffing actions can lead to data leakage that is difficult to prevent. The exposure of important data is a source of security threats, triggering packet attacks that malicious actors can exploit. [3] The security of the SIM MBKM website system can be considered not entirely secure, as the experiment results reveal the exposure of essential data and information. [4] Regarding security testing comparison between the Ettercap and Wireshark applications, Wireshark displays more detailed and important victim data, requiring less testing time than Ettercap.



Solutions and preventive measures can be implemented by encrypting sensitive data. In addition, the SIM MBKM website should provide appropriate security measures, considering that it is accessed by many students from UNM (Makassar State University) whose data can be compromised and misused by unauthorized parties. The suggested solution is to enhance security by using the Hypertext Transport Protocol Secure (HTTPS) as the protocol for the website. The advice is to be cautious and pay close attention when accessing web pages, internet banking, and social media platforms that require essential data such as usernames and passwords.

## 6. ACKNOWLEDGMENTS

Thanks are given to organizations or institutions that assist in research, directly or indirectly, in thinking and funding.

## AUTHOR CONTRIBUTIONS

All Author is responsible for building Conceptualization, Methodology, analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, supervision of project administration, funding acquisition, and have read and agreed to the published version of the manuscript.

## CONFLICTS OF INTEREST

The authors declare no conflict of interest.

## REFERENCES

1. Z. M. Luthfansa and U. D. Rosiani, "Pemanfaatan Wireshark untuk Sniffing Komunikasi Data Berprotokol HTTP pada Jaringan Internet," vol. 05, pp. 34–39, 2021.
2. F. Naim, U. Yunan, and K. Septo, "Analysis of wireless and cable network quality-of-service performance at Telkom University landmark tower using network development life cycle (ndlc) method," vol. 07, pp. 1034–1044, 2022.
3. I. P. Agus, E. Pratama, P. A. Dharmesta, and T. Informasi, "IMPLEMENTASI WIRESHARK DALAM MELAKUKAN PEMANTAUAN PROTOCOL JARINGAN ( Studi Kasus: Intranet Jurusan Teknologi Informasi Universitas Udayana)," vol. 3, no. 1, pp. 94–99, 2019.
4. Y. Hae and W. Sulisty, "Analisis Keamanan Jaringan Pada Web Dari Serangan Sniffing Dengan Metode Eksperimen," vol. 8, no. 4, pp. 2095–2105, 2021.
5. I. M. S. R. Putu Adhika Dharmesta, I Made Agus Dwi Suarjaya, "Efektivitas Sniffer Menggunakan Natural Language dalam Pembelajaran," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 4, no. 3, pp. 392–403, 2020.
6. M. K. Anam, D. Sudyana, A. Noviciatie, and N. Lizarti, "Optimalisasi Penggunaan VirtualBox Sebagai Virtual Computer Laboratory untuk Simulasi Jaringan dan Praktikum pada SMK Taruna Mandiri Pekanbaru J-PEMAS STMIK Amik Riau," <http://jurnal.sar.ac.id/index.php/J-PEMAS Optim.>, vol. vol 1, no. 2, pp. 37–44, 2020.
7. S. Syaifuddin, D. Regata Akbi, and A. Gholib Tammami, "Analisis Address Resolution Protocol Poisoning Attack Pada Router Wlan Menggunakan Metode Live Forensics," J. Komput. Terap., vol. 7, no. Vol. 7 No. 1 (2021), pp. 62–73, 2021, doi: 10.35143/jkt.v7i1.4575.
8. L. F. Sikos, "Packey Analysis For Network Forensics: A Comprehensive Survey," Forensic Sci. Int. Digit. Investig., vol. 32, p. 200892, 2020, doi: 10.1016/j.fsidi.2019.200892.
9. A. R. Maulana, H. Walidainy, and M. Irhamsyah, "Analisis Quality of Service ( QoS ) Jaringan Internet Pada Website e-Learning Universitas Syiah Kuala Berbasis Wireshark," vol. 6, no. 2, pp. 27–30, 2021.
10. W. Agustiar, A. Pratama, S. Junaidi, K. Padang, and S. Barat, "ANALISIS KEAMANAN PROTOKOL SECURE SOCKET LAYER TERHADAP SERANGAN PACKET SNIFFING PADA WEBSITE PORTAL," vol. 6, no. 1, 2022.
11. A. F. Muhamad Aznar Abdillah, Anton Yudhana, "Sniffing Pada Jaringan WiFi Berbasis Protokol 8021x Menggunakan Aplikasi Wireshark," vol. 4, pp. 1–8, 2020.

12. I. G. N. A. Kusuma, "Perancangan Simple Stateless Autentikasi dan Otorisasi Layanan REST-API Berbasis Protokol HTTP," vol. 4, no. 1, 2021.
13. I. Firdaus, J. Al Amien, T. Informatika, F. I. Komputer, U. M. Riau, and S. Matching, "String Matching untuk Mendeteksi Serangan Sniffing (ARP Spoofing) Pada IDS Snort," vol. 1, no. 2, pp. 44–49, 2020.
14. I. Riadi, A. Fadlil, and M. N. Hafizh, "Analisis Bukti Serangan Address Resolution Protocol Spoofing menggunakan Metode National Institute of Standard Technology," Edumatic J. Pendidik. Inform., vol. 4, no. 1, pp. 21–29, 2020, doi: 10.29408/edumatic.v4i1.2046.
15. B. Prabadevi and N. Jeyanthi, "A Review on Various Sniffing Attacks and its Mitigation Techniques," vol. 12, no. 3, pp. 1117–1125, 2018, doi: 10.11591/ijeecs.v12.i3.pp1117-1125.
16. G. E. A. Kamajaya, I. Riadi, Y. Prayudi, and U. A. Dahlan, "ANALISA INVESTIGASI STATIC FORENSICS SERANGAN MAN IN THE ARP POISONING BASED ON MAN IN THE MIDDLE ATTACK IN STATIC," vol. 3, no. 1, pp. 6–12, 2020, doi: 10.33387/jiko.
17. I. H. Santoso and A. I. Irawan, "Analisis Perbandingan Kinerja Sensor Jarak HC-SR04 dan GP2Y0A21YK Dengan Menggunakan Thingspeak dan Wireshark," J. Rekayasa Elektr., vol. 18, no. 1, pp. 43–52, 2022, doi: 10.17529/jre.v18i1.23359.
18. N. K. Hamzidah et al., "Studi Komparatif QoS pada Aplikasi Video Meeting Tool dalam Jaringan 4G LTE Menggunakan Wireshark Comparative Study of QoS on Video Meeting Tool Application in 4G LTE Network Using Wireshark," vol. 12, no. index 1, pp. 31–40, 2023.
19. Y. Li and G. Mogos, "Digital forensics on Tencent QQ-instant messaging service in China," vol. 29, no. 1, pp. 412–420, 2023, doi: 10.11591/ijeecs.v29.i1.pp412-420.
20. D. Glăvan et al., "Sniffing attacks on computer networks," vol. XXIII, no. 1, 2020, doi: 10.21279/1454-864X-20-I1-027.
21. N. Kurniawati and S. Agoes, "Analysis of Voice Captured Packet using Wireshark," vol. 17, no. 2, pp. 205–216, 2020.
22. P. Asrodia and V. Sharma, "Network Monitoring and Analysis by Packet Sniffing Method," vol. 4, no. May, pp. 2133–2135, 2013.
23. B. Patel and P. Shah, "Simulation, modeling, and packet sniffing facilities for IoT : A systematic analysis," 2020, doi: 10.11591/ijece.v10i3.pp2755- 2762.
24. C. Yuan, J. Du, M. Yue, and T. Ma, "The design of large scale IP address and port scanning tool," Sensors (Switzerland), vol. 20, no. 16, pp. 1–12, 2020, doi: 10.3390/s20164423.
25. D. H. Syaiful Anam, "Implementasi Sistem Keamanan Hotspot Jaringan Menggunakan Metode OpenSSL ( Secure Socket Layer )," vol. 6, no. 1, pp. 57–64, 2020.