International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Enhancing Data Security: A Comprehensive Study on the Efficacy of JSON Web Token (JWT) and HMAC SHA-256 Algorithm for Web Application Security

Manish Rana
Department OF Computer Engineering,
Thakur College of Enginering And Technology,
Mumbai, India.
manish.rana@thakureducation.org,manishrana23@gmail.com

Ayush Pandey
Department of Computer Engineering,
Thakur College of Enginering And Technology,
Mumbai, India.
ayushpandey1407@gmail.com

Ankit Mishra
Department of Computer Engineering,
Thakur College of Enginering And Technology,
Mumbai, India.
ankitmishra8652@gmail.com

Vishal Kandu
Department of Computer Engineering,
Thakur College of Enginering And Technology,
Mumbai, India.
kanduvishal07@gmail.com

Abstract—In today's digital era, data security is a very important aspect in various applications and services. In order to protect the integrity, confidentiality and authentication of data, security technologies such as JSON Web Token (JWT) and HMAC SHA256 algorithm are widely used. JWT is an open standard (RFC 7519) that is used to represent information in the form of tokens that can be signed digitally. The research methodology used in this research is a descriptive research method. The descriptive method is a method that describes the purpose of the data collected and records every aspect of the situation being investigated to get a clear picture of what is needed. It was found that there were several data leaks when data security was not implemented in layers, including cases that had occurred such as loss of important data contained in the website and leaks of important data which caused identities to be spread widely. Conclusions from the use of JSON Web Token (JWT) and HMAC-SHA-256 algorithm for website security is that this combination provides a strong layer of protection against security threats that are common in the online environment.

Keywords-. JWT; HMAC; SHA-256; Security; Web Services.

I. Introduction

In the current digital era, data security is a very important aspect in various applications and services [1][2]. In order to protect the integrity, confidentiality and authentication of data, security technologies such as JSON Web Token (JWT) and the HMAC-SHA256 algorithm are widely used[3][4]. JWT is an open standard (RFC 7519) used to represent information in the form of tokens that can be digitally signed [5][6]. The HMAC-SHA256 algorithm, on the other hand, is a hash algorithm that uses a secret key to generate a digital signature. The combination of utilizing JWT with the HMAC-SHA256 algorithm allows applications to achieve a higher level of security in data exchange [7][2].

Utilization of JWT with the HMAC-SHA256 algorithm provides a high level of security in data exchange [8][9]. The main advantages include (1) data integrity. By using digital signatures generated by the HMAC-SHA256 algorithm, applications can ensure that the data in the token is not changed during transmission [10]. (2) Authentication, JWT Tokens that have been signed with the HMAC-SHA256 algorithm can be used to authenticate users or third parties and (3) Confidentiality, Even though the payload in the token can be decoded by anyone, the digital signature remains safe because only the party has a secret key that can create a valid signature[11][12].

In this journal, we will discuss the level of security that can be achieved by JSON Web Token (JWT) to secure data exchange from two sides using the HMAC SHA-256 algorithm. Data security using JWT is very important to implement in today's industrial world where almost all transactions are digital where the data is easily lost or misused by irresponsible parties if security is not added to it.

II. LITERATURE SURVEY

1. Systematic literature review: Search for journals related to the use of JSON Web Token (JWT) and the application of the HMAC SHA-256 Algorithm on various platforms [13]. Then conduct a systematic and descriptive review of related literature such as research findings, concepts and theories.
2. Case Studies: look for examples of case studies regarding security weaknesses in using JSON Web Tokens (JWT) for data security, such as in business environments, educational environments or even government environments. The aim is to provide an overview of the security gaps that can be resolved with JWT technology and the HMAC SHA-256 algorithm.

3. Data Analysis: Collecting data from sources such as books and journals to find out how current security is using JWT and SHA-256 technology [14].

4. Theoretical Study: The theoretical study aims to help understand how secure JWT and HMAC SHA 256 are when applied for data security authentication.

Attached below is the literature review carried out in writing this article. There were 10 pieces of literature taken to create relevant references with the article title which can be seen in table 1:

No	Writer	Year	Topic	Methodology	Problem Gap Identified
1	Rohmat Gunawan	2019	JSON Web Token (JWT) for Authentication process Interoperability's Architecture basis RESTful Web Service	The methodology involves designing a JWT-based authentication model for a RESTful Web Service, with steps including user login, token generation, and backend authentication. Implementation in PHP, testing HTTP server responses, and validating token data ensure a secure and functional system.	The paper lacks discussion on the security of token storage, particularly in local storage (cookies), leaving uncertainties about its safety. There is a gap in identifying more suitable encryption algorithms for the implementation of JSON Web Token (JWT) in RESTful Web Service applications.
2	Heni Sulastri and Rizal Nugroho	2018	RESTful Security Web Service Use JSON Web Token (JWT) HMAC SHA-512	The study implemented and compared the standard JWT with HMAC SHA-256 and the optimized version using HMAC SHA-512. The evaluation focused on the algorithm's performance in terms of execution time and token size. Results showed a 1% improvement in execution time with SHA-512, albeit with a 2% increase in token size compared to SHA-256.	The identified security gap in the paper pertains to concerns about the adequacy of existing security measures in RESTful Web Services. The study proposes addressing this gap by implementing JSON Web Token (JWT) with the HMAC SHA-512 algorithm, aiming to enhance the security of the web services.
3	Yogiswara and Wijono Dahlan	2014	Performance Service on Process Web Integration Data	The methodology involves designing and implementing a data integration model incorporating XML-RPC, SOAP, and REST web service methods. Performance evaluation includes measuring transaction speed and user-perceived latency, with a focus on identifying the optimal configuration—highlighting the "Apache" client, "Nginx" server, and the "REST" method—for efficient data integration implementation.	The research discusses the development of a data integration model using XML-RPC, SOAP, and REST web service methods, evaluating their performance on different web servers. The results indicate that the REST method, implemented with Apache as the client and Nginx as the server, shows the best latency for data integration.
4	Adhitya Bhaviyuga, Mahendra	2017	Architectural Design of Token based Authentication of MQTT	The study implements a token-based authentication system for MQTT in constrained IoT devices, involving a	Token-based authentication on resource-limited IoT devices may compromise system efficiency,

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

	Data and Andri Warda		Protocol in Constrained IoT Device	conditional token generation process, storage, and usability/performance testing.	given constraints in processing power and memory.
5	Digvijaysinh M Rathod	2017	Performance Evaluation Restful Web Services And Soap / Wsdl Web Services	Implemented a prototype system, integrated with a mobile client, to conduct experiments comparing the performance and scalability of SOAP/WSDL and RESTful web services, revealing superior results for RESTful services in mobile environments.	The paper lacks a crucial discussion on the security implications of choosing between RESTful and SOAP/WSDL web services, focusing primarily on performance aspects.
6	Kartika Imam Santoso	2013	Factor Login Security WebUse Authentication One Time Password With Hash SHA	Secured logins with SMS-based OTP using SHA hashing, Gammu for automatic SMS delivery, and three-minute validation, validated through trials with Indonesian cellular operators	Lacks clarity on securely storing six-digit OTPs generated from the hash and addressing potential vulnerabilities in storage.
7	Gabriel Yoda Gustiegan1 and painem	2022	Web Implementation Service Restful With Authentication Json Web Token And Algonithm Aes-256 Cryptography For Application Borrowing Laboratory Based Mobile On Bud University	The application employs REST, AES-256, and JWT for lab reservation, encompassing design, architecture, database, and specifications. Testing covered various scenarios, ensuring 100% accuracy in login, data display, and encryption.	The successful implementation demonstrated the application's effectiveness with no identified gaps. It performed as expected, meeting all requirements and proving its reliability.
8	KV Kanmani, P.Sc dan S. Smitha	2013	Survey on Restful Web Services Using Authorization (Oauth),	The paper follows a literature survey approach to explore various techniques in Restful Web Services, emphasizing the integration of REST and OAuth 2.0. However, it lacks a detailed research methodology, such as experimental design or data collection methods, limiting the depth of analysis.	The identified gap lies in the absence of practical examples or empirical validation of the proposed REST and OAuth integration in real-world scenarios. The paper could benefit from showcasing the application of discussed concepts through case studies, enhancing the practical relevance and applicability of the presented techniques.
9	Rohmat Gunawan	2019	JSON Web Token (JWT) for On Open Authentication pada Interoperability Based architecture Web RESTful Service	Implemented JWT authentication in a RESTful Web Service for blood donation, ensuring interoperability across various platforms. PHP with HMAC-256 standard was employed for implementation.	Successfully achieved scalability and security with JWT in the blood donation web service. However, lacking discussions on token storage security and the impact of token modifications, requiring further research.

III. RESEARCH OBJECTIVES AND BENEFITS

Research on JSON Web Tokens (JWT) in the authentication process on websites has significant goals and benefits in securing and improving the performance of the authentication system. The main goal is to develop and validate an efficient, secure, and scalable authentication mechanism, using JWT as a reliable solution. JWT is a token format that consists of three parts, namely header, payload, and signature, which can be encrypted and decrypted to ensure data integrity and authenticity [2].

The benefit of this research is increased safety. By using JWT, user credential information can be encoded and signed, reducing the risk of hacking and data misuse. JWT also enables the use of strong encryption algorithms to protect sensitive information, thereby reducing the potential for unauthorized access. Another benefit is increased efficiency. The authentication process using JWT can be faster than traditional methods that involve repeated calls to the authentication server. The JWT issued after the first authentication contains the required information, reducing the server load in verifying credentials on each request. Apart from that, JWT's research in the authentication process on websites also has benefits in terms of system scale and scalability. With JWT, the system can be easily extended and integrated with other services, both inside and outside the application ecosystem. This allows the development of architectures that are more modular and responsive to user growth or additional services [12].

Overall, research on the use of JWT in the authentication process on websites has the main aim of increasing the security, efficiency and flexibility of the authentication system. By leveraging the power of encryption and digital signatures, as well as the ability to reduce server load and facilitate integration, JWT becomes a powerful tool in achieving this goal [15].

IV. METHODOLOGY

The research methodology used in this research is descriptive research method. The descriptive method is a method that describes the purpose of the data collected and records every aspect of the situation being investigated to get a clear picture of what is needed. Regarding flow in research. As for the research flow, it can be seen in figure 4.1:

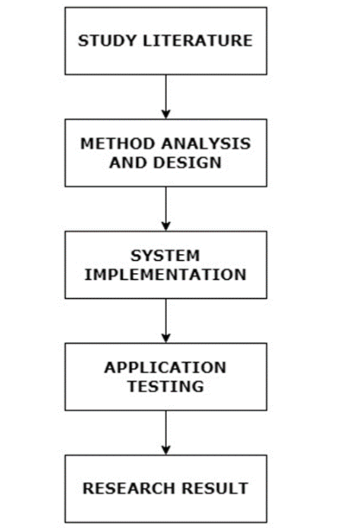


Fig. 4.1 : Research Flow

LITERATURE STUDY

Literature study is an activity related to methods of collecting library data, reading and taking notes and processing research materials for the process of investigating, analyzing and combining information that has been collected from written sources that are relevant to the topic or problem. These literature sources can be scientific journals, articles, books, theses, research reports, or documents.

A. *Method analysis and design*

Analysis and design methods are processes related to identifying, understanding, planning, and developing approaches or ways to achieve certain goals. It is a concept frequently used in various scientific disciplines and industries to optimize complex processes, systems or solutions.

B. *Implementation*

It is a stage in development where the plan or design that has been made at the analysis and design stage is realized into a system that functions in accordance with the goals and needs that have been set.

C. *Application Testing*

The process of testing and evaluating an entire software or hardware system to ensure that the system functions according to predetermined specifications and meets user needs.

D. *Results and Research*

Research results refer to findings, information, data, analysis and conclusions obtained from the research process carried out. It is the final result of a research effort that reflects new understanding and insight into the topic or problem under study.

V. RESULT AND DISCUSSION

A. *Planning and Model Analysis*

The following graphic typically depicts the suggested model for implementing token-based authentication (JWT) on the REST Web Service architecture.

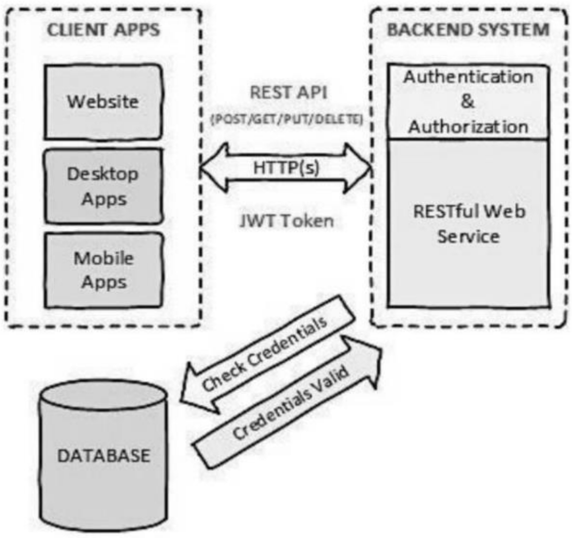


Fig. 5.1. Authentication Overview

The backend system that manages the Restful Web Service's authorization and authentication procedures is depicted in the image. Following a successful login, the server offers A JWT token, which serves as a key to access server resources, is the

Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

form of the answer. Various online, desktop, and mobile clients can access the REST API.

The following procedures are part of the suggested model:
a. Users include desktop, mobile, and web-based clients.
b. The token is checked by the system. The user is sent to the login page if the token is unavailable or has expired.
c. The user logs in and sends the API's credentials.
d. Backend system: some operations require user authentication.
e. Database: Verify user credentials f. A JWT token is returned to the client if the credentials are legitimate.

The next step is to verify that the token's access rights are present in the payload if it hasn't expired. The blood donation web service will react by providing the user with the resources they require if the token possesses access privileges to those resources. The following illustrates the token verification process and its description.

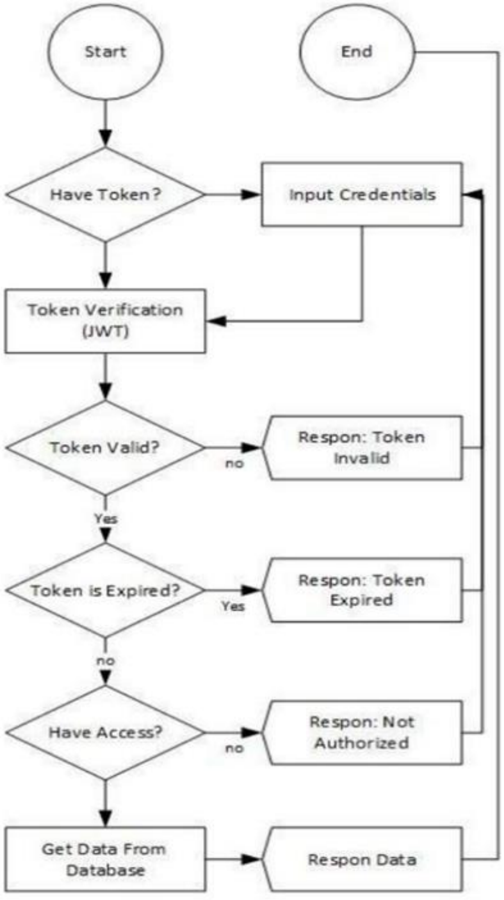
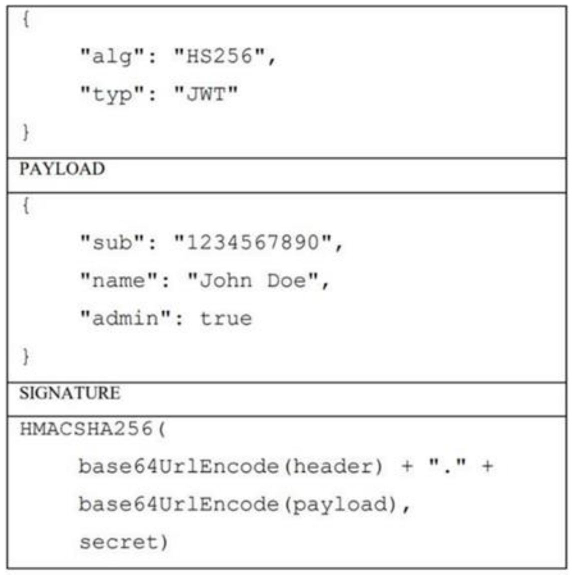


Fig. 5.2. Check Payload Access Rights

B. *Analysis JSON Web Token (JWT)*

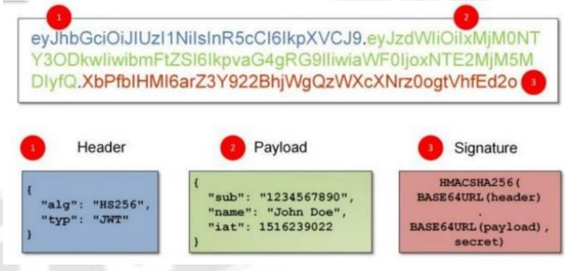
Tokens, such as JWT, are used to send data securely and verifiably between two parties. The content, signature, and header are the three primary components of the JWT token. Both the kind of token and the kind of method that was used to generate the signature are contained in the header.
Claims with information about users, roles, permissions, and other things are included in the payload. The signature is a digital signature that is produced by combining the payload, secret key, and header.
By using this signature, you can confirm that the token hasn't been altered since it was created and assure data integrity.
The three basic components—payload, signature, and header—that can be combined to create a JWT are shown in Figure 5.3. The token that can be acquired through the use of the General Structure of JSON WEB TOKEN (JWT).



```
{
    "alg": "HS256",
    "typ": "JWT"
}
PAYLOAD
{
    "sub": "1234567890",
    "name": "John Doe",
    "admin": true
}
SIGNATURE
HMACSHA256(
    base64UrlEncode(header) + "." +
    base64UrlEncode(payload),
    secret)
```

Fig .5.3. General Structure of JSON WEB TOKEN (JWT)

The three primary components (header, payload, and signature) that can be combined to create a JWT are shown in Figure 5.3. Figure 5.4 shows an example of a token that was obtained using the JWT format. It looks like this.



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NT
Y3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5M
DIyfQ.XbPfbIHMI6arZ3Y922BhjWgQzWXcXNrz0ogtVhfEd2o
```

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "iat": 1516239022
}
```

```
HMACSHA256(
BASE64URL(header)
.
BASE64URL(payload),
secret)
```

Fig. 5.4 Example of JWT Token (Securitum.Com)

The JWT token is explained in Figure 5.4, where each segment of the long string represents a different portion of the structure and is separated by dots specifically, JWT's Header, Payload, and Signature. When the code above is encoded with HMAC SHA-256.

C. *Analysis of the HMAC SHA-256 Algorithm*

A hash algorithm and a secret key are used by the cryptographic concept known as HMAC (Hash-based Message Authentication Code) to produce an authentication code or

Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

signature. One of the most used hash methods, SHA-256, yields a hash value of 256-beet. The HMAC-SHA256 technique is used in the context of JWT to convert the token into a digital signature.

This procedure entails mixing the payload with a secret key and applying the SHA-256 algorithm to produce a hash result. The digital signature that will be a part of the JWT token is represented by this hash value. The HMAC SHA-256 algorithm has the following functions, which are shown in Figure 5.5

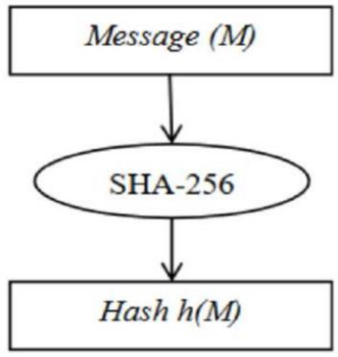


Fig. 5.5. Main functions of SHA-256

The primary purpose of SHA-256 is to take in input in the form of data or a message M, which can be any length, and then output a 256-bit hash value, h(M). Six logical functions are used by SHA-256, and they each work with 32-bit words denoted by the letters x, y, and z.

The result of each function is a new 32-bit word. Following are the boolean functions in SHA-256:

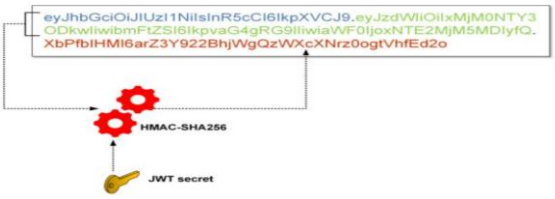
$$
\begin{aligned}
Ch(x,y,z) &= (x \wedge y) \oplus (\neg x \wedge z) \\
Maj(x,y,z) &= (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z) \\
\Sigma_0^{\{256\}}(x) &= \mathrm{ROTR}^2(x) \oplus \mathrm{ROTR}^{13}(x) \oplus \mathrm{ROTR}^{22}(x) \\
\Sigma_1^{\{256\}}(x) &= \mathrm{ROTR}^6(x) \oplus \mathrm{ROTR}^{11}(x) \oplus \mathrm{ROTR}^{25}(x) \\
\sigma_0^{\{256\}}(x) &= \mathrm{ROTR}^7(x) \oplus \mathrm{ROTR}^{18}(x) \oplus \mathrm{SHR}^3(x) \\
\sigma_1^{\{256\}}(x) &= \mathrm{ROTR}^{17}(x) \oplus \mathrm{ROTR}^{19}(x) \oplus \mathrm{SHR}^{10}(x)
\end{aligned}
$$

Fig. 5.6. Logic SHA-256 functions

Information:
ROTR (Right Rotate)
SHR (Right Shift)

Subsequently, the JWT token is encrypted using JSON Web Encryption (JWE) and the HMAC SHA-256 method. The message received is assigned a secondary key that corresponds to the outcome.

The image is shown in picture 5.7



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3
ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.
XbPfbIHMI6arZ3Y922BhjWgQzWXcXNrz0ogtVhfEd2o
```

Fig. 5.7. JWT and HMAC SHA-256 merger

The image above aims to prevent payload from creating new signatures because it requires a key to do so which is only in the API configuration.

D. Utilization of JWT and HMAC SHA-256

When JWT is used in conjunction with the HMAC-SHA256 technique, data communication is highly secure. Principal benefits consist of:

1. Data Integrity: Applications can be sure that the data in the token hasn't changed during transmission by employing digital signatures produced by the HMAC-SHA256 algorithm.

2. Authentication: Users or other parties can be authenticated by using JWT tokens that have been signed using the HMAC-SHA256 technique.

3. Confidentiality: Although everyone can decode the payload included in the token, the digital signature is still safe because only the owner of the secret key is able to produce a legitimate signature.

E. Implementation of JWT and HMAC SHA-256

The encryption method for web tokens including personal information will be encoded into a JSON format and constitute a payload for the JSON web signature (JWS).

Next, a digital signature with a MAC (Message Authentication Code) is used to encrypt the claim. The algorithm and type of token are contained in the token header part of JWT. User claims and metadata are contained in the token payload. Lastly, a digital signature is produced by merging the payload, secret, and header encodes with the signature token.

The process involves utilizing the 'echo. Context' structure in Golang to represent an HTTP request context, encapsulating details like the request URL, headers, and body. Additionally, a 'jwt.Token' structure is employed to handle JSON web tokens securely, facilitating the secure transmission of information between parties. To organize and store token claims, a custom structure named 'JwtCustomClaims' is created. The primary function revolves around extracting claims from tokens within the context and accessing user information, resulting in the retrieval and return of these claims. This process ensures a secure and organized way of managing user-related information within the specified HTTP request context.

F. JSON Web Token Testing

The testing to retrieve data which is then encoded with JWT. The Login () function will be examined in this test. In the event that the login is successful, this function generates a JWT token and verifies the user's email address and password. Valid mock data, specifically the email address and password entered into the database, will be provided in the first test.

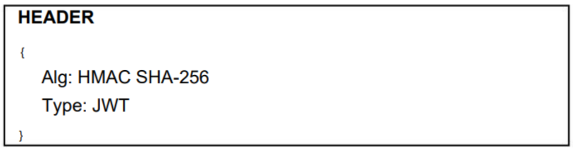
This test will produce a JWT token and be successful.

An erroneous email address or password will be the invalid mock data provided in the second test. There will be an error and the test will fail.

G. JSON web token testing result

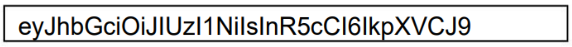
The JSON web token can be divided into three pieces, separated by commas and recognized by separators with dots, like (aaaa.bbbbbb.ccc), according to the results of testing it on Laravel 9 using Thunder Client. First, there are two sections in the Header section: one declares the token type (JWT) and the other hashing algorithm (HMAC256), which is encoded into (eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV
CJ9). Next, the user entity and other metadata made up of the following are contained in the payload web token that carries the claim.

1. Header



```
HEADER
{
   Alg: HMAC SHA-256
   Type: JWT
}
```

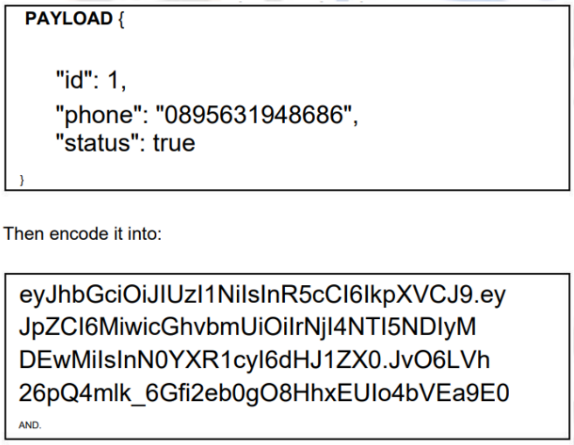
Encoded into:



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
```

Table. 5.2. Header JWT

2. Payload



```
PAYLOAD {

    "id": 1,
    "phone": "0895631948686",
    "status": true
}
```

Then encode it into:

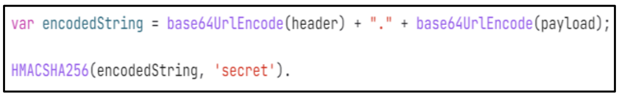
```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.ey
JpZCI6MiwicGhvbmUiOiIrNjI4NTI5NDIyM
DEwMiIsInN0YXR1cyI6dHJ1ZX0.JvO6LVh
26pQ4mlk_6Gfi2eb0gO8HhxEUIo4bVEa9E0
AND.
```

Table. 5.2. Payload JWT

3. Signature

Taken from the header and payload which have been encoded as secret
(3zTz7C4SWOd4QIwwlBLk2U2BBzShA1GMRE2vvdQvM8)
.

To get the third part of the signature in use:



```
var encodedString = base64UrlEncode(header) + "." + base64UrlEncode(payload);

HMACSHA256(encodedString, 'secret').
```

VI. ACKNOWLEDGMENT

This manuscript talk about Enhancing Data Security: A Comprehensive Study on the Efficacy of JSON Web Token (JWT) and HMAC SHA-256 Algorithm for Web Application Security.
There is full contribution of Ayush, Aniket and Vishal would be helpful to our Organization for giving us time to complete our research.

VII. CONCLUSION

The conclusion of using JSON Web Token (JWT) and the HMAC-SHA-256 algorithm for website security is that this combination provides a strong layer of protection against security threats that are common in the online environment.
JWT is very effective and popular for securing communications between users and servers. With JWT, sensitive information such as user identity and access permissions can be encrypted in the form of digitally signed tokens. This helps prevent hacking and data manipulation by unauthorized parties. By relying on JWT, websites can provide an additional layer of security to ensure that only authorized users have appropriate access to various parts of the site.

REFERENCES

[1] T. D. PUTRI, W. SUGENG, and R. KATRI, "Sistem Otentikasi Login Dengan Single Sign-On Untuk Mengakses Banyak Sistem," MIND J., vol. 4, no. 2, pp. 96–110, 2019, doi: 10.26760/mindjournal.v4i2.17-31.

[2] F. C. Ramdani, A. Rahmatulloh, R. N. Shofa, J. S. Nomor, J. Tasikmalaya, and I. Barat, "Implementasi JSON Web Tokenpada Authenticationdengan Algoritma HMAC SHA-256," *Sist. Sist. Inf. Vol. 12, Nomor 1, Januari 2023 194-205*, vol. 12, no. 1, pp. 2540–9719, 2023, [Online].

[3] M. Y. Arif, M. Marsono, and A. Azanuddin, "Implementasi Web Service Enterprise Resource Planning (ERP) Menggunakan Metode Representational State Transfer (REST)," *J. Cyber Tech*, no. x, 2019, [Online].

[4] I. Afrianto and E. B. Setiawan, "Kajian virtual private network (vpn) sebagai sistem pengamanan data pada jaringan komputer (studi kasus jaringan komputer unikom),"*Maj. Ilm. UNIKOM*, vol. 12, no. 1, pp. 43–52, 2015, doi: 10.34010/miu.v12i1.34.

[5] A. Hibsy and A. Wibowo, "Implementasi Fitur Keamanan dengan JSON Web Token dan Fitur Geo-tagging pada Aplikasi Web Service Training From Home," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 3, pp. 591–600, 2020.

[6] I. Taofik and I. Afrianto, "Analisis Keamanan dan Perlindungan Data pada Komputasi Awan dalam Ruang Lingkup Pendidikan."

[7] T. Ariyadi, T. L. Widodo, N. Apriyanti, and F. S. Kirana, "Analisis Kerentanan Keamanan Sistem Informasi Akademik Universitas Bina Darma Menggunakan OWASP,"*Techno.Com*, vol. 22, no. 2, pp. 418–429, 2023, doi: 10.33633/tc.v22i2.7562.

[8] M. H. Maula and D. Kusumaningsih, "Implementasi Web Service Pada Aplikasi Pesan Pakaian implementation of Web Service in Clothing Message," no. September, pp. 1201–1209, 2022.

[9] I. Afrianto, A. Heryandi, and S. Atin, "Blockchain-based Trust, Transparent, Traceable Modeling on Learning Recognition System Kampus Merdeka," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 22, no. 2, pp. 339–352, 2023, doi: 10.30812/matrik.v22i2.2780.

[10] R. Laipaka, "Penerapan JWT untuk Authentication dan Authorization pada Laravel 9 menggunakan Thunder Client," *Semin. Nas. Corisindo,* pp. 450–455, 2022, [Online].

[11] M. I. Perkasa and E. B. Setiawan, "Pembangunan Web Service Data Masyarakat Menggunakan REST API dengan Access Token," *J. Ultim. Comput.*, vol. 10, no. 1, pp. 19–26, 2018, doi: 10.31937/sk.v10i1.838.

[12] Sopingi, F. E. Nastiti, and A. S. Majid, "Implementasi JSON Web Token Authentication pada Aplikasi Pembayaran Berbasis Mobile," *Semin. Nas. Call 8 Pap. Hubisintek,* pp. 343–351, 2021.

[13] I. Afrianto, A. Heryandi, A. Finandhita, and S. Atin, "E-document autentification with digital signature model for smart city in Indonesia," *J. Eng. Sci. Technol*., vol. 15, pp. 28–35, 2020.

[14] I. G. Anugrah and M. A. R. I. Fakhruddin, "Development Authentication and Authorization Systems of Multi Information Systems Based REst API and Auth Token,"*Innov*. Res. J., vol. 1, no. 2, p. 127, 2020, doi: 10.30587/innovation.v1i2.1927.

[15] R. Priyatna and S. Waluyo, "Implementasi Restful Dengan Jwt Untuk Booking Barang Di Primajaya Multisindo," *Semin. Nas. Mhs. Fak. Teknol. Inf. Jakarta-Indonesia,* no. September, pp. 1040–1047, 2022.