IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 3, March 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 3, March 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1003029 |

WebApp for File Encryption and Decryption

Aditya Shinde[1], Devesh Yadav[2], Abhay Vishwakarma[3], Dr. Najmuddin Aamer[4]

UG Students, Dept. of Computer Engineering, TCOE, Mumbai University, Maharashtra, India[1,2,3]

Professor, Dept. of Computer Engineering, TCOE, Mumbai University, Maharashtra, India[4]

ABSTRACT: It is imperative to protect data and information on a computer from unauthorized access.To protect data, you can use cryptography as a data encryption method. In organizations and government offices data privacy is taken seriously and a lot of resources are spent to keep the information secure but data privacy for individuals is often neglected or people are not aware of the value of their data. Our project aims to raise awareness for data privacy and to make it easily accessible and free for the regular user to encrypt their important sensitive data.

KEYWORDS: Cryptography; Data Privacy; Encryption; Decryption.

I. INTRODUCTION

File Encryption and Decryption is a web application that provides secure local file encryption/decryption in the browser. It uses modern cryptographic algorithms with chunked AEAD stream encryption/decryption while still being fast and secure. As social media, mobile devices, and miniaturized storage devices have grown, data security has become an essential requirement.The value of data has been increasingly outweighing the value of the devices underlying it in the past few years.In addition to the loss of personal information and intellectual property, laptops and USB thumb drives stolen from individual users may be used for identity theft. Hence, strong cryptographic methods are becoming increasingly necessary for protecting stored data against unauthorized access.

With the ever-increasing digitalization the risk of data leaks over the internet have increased exponentially. This calls for an immediate need for preventative measures against confidentiality breaches. Encryption of data has been looked upon as a promising technique over the previous few decades. The motivation for the development of this application came when a very well-known cloud service provider was hacked in late 2016 which resulted in the leak of sensitive information of the users. In addition, many big reputed companies have been accused of sharing sensitive information with a variety of government agencies voluntarily. Keeping all this in mind we have created a web application based on a client-server architecture that provides encryption decryption services to the users without implementing any backdoors. The whole application architecture ensures complete confidentiality of the data provided by the client even from the server itself.

II. LITERATURE SURVEY

Amrita Sahu et.al proposed a new key generation algorithm based on palm print which is used for encryption and decryption of images. Even if several people are listening in on the chat, our system allows one party to communicate a secret image to another party across an open network. The authors of this research developed a bit XOR-based image encryption and decryption system. Among the salient features of the proposed asymmetric image encryption scheme are: (a) Lossless encryption of the image. (b) Less computational complexity. (c) Convenient implementation. (d) Customizing the size of the matrix according to the size of the image. (e) The Encryption/decryption scheme uses integer arithmetic and logic operations.

Raman.S.Jamgekar et.al showed that the MREA method is used to encrypt files and transport encrypted files to the opposite end, where they are decrypted. It works well for small files, but it takes a long time for large files. Only one file can be encrypted and transmitted at a time. In the future, multiple file encryption and decryption can be accomplished. The application for the project was created with efficiency and reusability in mind. This algorithm provides a high level of protection. Where high security file communication is necessary in public forums, a modified RSA technique for file transmission can be utilized.

III. PROPOSED SYSTEM

Aim of the proposed algorithm is to encrypt files securely and rapidly using the user's system as a server, So the time of sending and receiving requests to the server will be saved. The proposed system is User friendly.

The proposed system suggests a new WebApp which encrypts and decrypts the files fast and securely and is user friendly too. Most users are not comfortable with the fact that their extremely private or confidential files can be

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 3, March 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1003029 |

accessed for various purposes. Even nowadays there is an influx of users who don't know about data theft and data piracy.

So, this Proposed system aims to spread awareness and to make the data secure by encrypting the user's file. This project uses AEAD and AES algorithms to encrypt and decrypt the data. The system is easy to use for any user; they just have to upload their file and choose the encryption method that they want to use. They have to select whether they want a password for their file or they want to use the recipient's public key to encrypt their file.

If user chooses password to encrypt the file, then he can use the same to decrypt the file and if he chooses Public and private key combination of sender and receiver's end then he will have to use the public key of the recipient to encrypt the file and while decrypting the file receiver needs to enter their own private key to decrypt the file.

IV. PSEUDO CODE

File Encryption (stream)

In order to use the app to encrypt a file, the user has to provide a valid file and a password. this password gets hashed and a secure key is derived from it to encrypt the file.

```
let res = sodium.crypto_secretstream_xchacha20poly1305_init_push(key);
header = res.header;
state = res.state;
let tag = last
  ? sodium.crypto_secretstream_xchacha20poly1305_TAG_FINAL
  : sodium.crypto_secretstream_xchacha20poly1305_TAG_MESSAGE;
letencryptedChunk = sodium.crypto_secretstream_xchacha20poly1305_push(
  state,
newUint8Array(chunk),
null,
  tag
);
stream.enqueue(signature, salt, header, encryptedChunk);
```

The `crypto_secretstream_xchacha20poly1305_init_push` function creates an encrypted stream, which uses an internal initialization vector, generated automatically, and the key to create a state. A 192-bit header is then created from the stream header.
It is this function that must be called before the encrypted stream is created. Thereafter, the key will not be needed.
There is a short header that is 192 bits in size for an encrypted stream. Those headers must be sent/stored before the stream of encrypted messages, as they are necessary for decryption. Using a different header would result in a failed decryption. Therefore, the header content does not need to be secret.

Each message has a tag that corresponds to the value of last, indicating whether it is the last chunk of the file or not.
It can be one of the following:

1. `crypto_secretstream_xchacha20poly1305_TAG_MESSAGE`: This doesn't impart anything about the contents of the message.
2. `crypto_secretstream_xchacha20poly1305_TAG_FINAL`: The stream ends with this message, which removes the secret key used to encrypt the previous message.

The `crypto_secretstream_xchacha20poly1305_push()` function encrypts the file chunk using the state and the tag, without any additional information (`null`).
The XChaCha20 stream cipher Poly1305 MAC authentication is used for encryption.
`stream.enqueue()` function adds the File Encryption & Decryption signature(magic bytes), salt and header followed by the encrypted chunks.

File Decryption (stream)

```
let state = sodium.crypto_secretstream_xchacha20poly1305_init_pull(header, key);
```

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 3, March 2022 ||

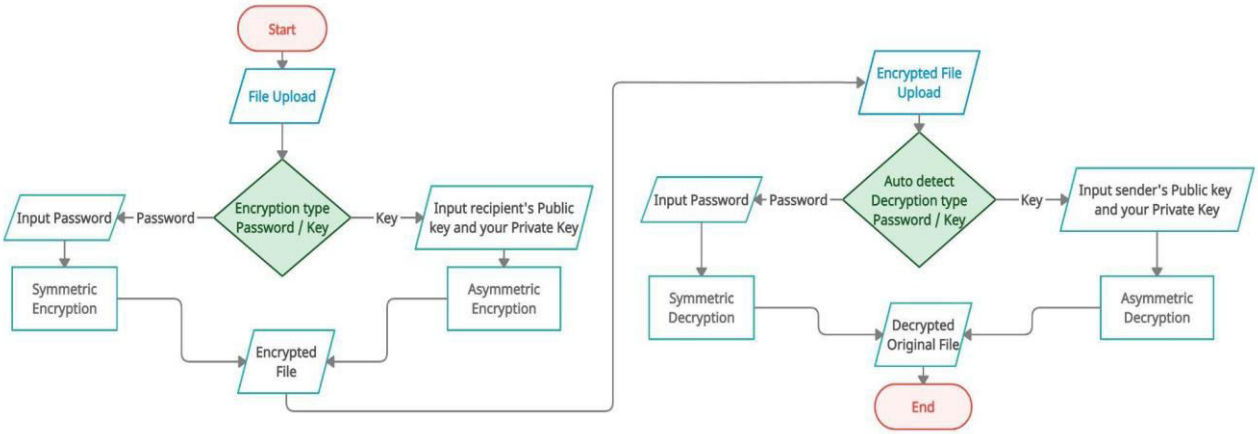
| DOI: 10.15680/IJIRCCE.2022.1003029 |

```
let result = sodium.crypto_secretstream_xchacha20poly1305_pull(
  state,
newUint8Array(chunk)
);
if (result) {
letdecryptedChunk = result.message;
stream.enqueue(decryptedChunk);
if (!last) {
// continue decryption
  }
}
```

The `crypto_secretstream_xchacha20poly1305_init_pull()` function initializes a state given a secret key and a header. The key is created by slicing the header from the file and using the password provided during decryption. For further operations, the key will no longer be required.

The `crypto_secretstream_xchacha20poly1305_pull()` function verifies that the chunk contains a valid ciphertext and authentication tag for the given state.
This function will keep looping until it finds a message that has the `crypto_secretstream_xchacha20poly1305_TAG_FINAL` tag.
The function produces an error if the decryption key is wrong.
If the ciphertext or the authentication tag appear to be invalid it returns an error.

V. FLOW DIAGRAM



VI. CONCLUSION AND FUTURE WORK

The project was carried out to develop a webapp for secure file encryption and decryption that can be easily accessible and convenient for the general user. The goal of this project was to create a webapp which uses the user system as a server to encrypt/decrypt the data. since the whole process of encryption and decryption is done locally on the user system no data is ever uploaded to any server so the data is private.

The proposed system is much more efficient and secure compared to the existing one. This webapp is user friendly, easier and lighter to use. The project's goals were met by observing the software development process as well as software design and implementation concepts. Two primary sections were planned and implemented to achieve the project's purpose. Firstly, the design of the UI is attractive, intuitive, responsive, and with a good user experience in mind. This was fulfilled and implemented by following responsive design guidelines for all devices and mobile-first approach to make the webapp easily accessible to the general user. Secondly, the implementation of the actual project with all functionalities.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 3, March 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1003029 |

In conclusion, it is important to know that this application could still be improved upon by integrating this web app's functionality with other social media and file sharing services. For example, we can also add a feature that makes this a plugin to browsers so the user can easily encrypt the files on the go.

REFERENCES

1. H. Mohan, and R. Raji. "Performance Analysis of AES and MARS Encryption Algorithms". International Journal of Computer Science Issues (IJCSI), Vol. 8, issue 4. 2011.
2. Rajan.S.Jamgekar, Geeta Shantanu Joshi, "File Encryption and Decryption Using Secure RSA", International Journal of Emerging Science and Engineering (IJESE), Vol-1, Issue-4, February 2013.
3. Akanksha Mathur, "A Research paper: An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms", International Journal on Computer Science and Engineering (IJSCE), Vol. 4 No. 09 sep 2012.
4. Singh S Preet, Mani Raman, "Comparison of Data Encryption Algorithms", International Journal of Computer science and Communications, Vol. 2, No.1, January-June 2011, pp. 125-127.
5. Thakur Jawahar, Kumar Nagesh. "DES, AES and Blowfish Symmetric Key Cryptography algorithm Simulation Based Performance Analysis", IJETAE, vol. 1, Issue 2, DEC. 2011, pp. 6-12.
6. M. Chase and S. S. M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," in ACM Conference on Computer and Communications Security, 2009, pp. 121–130.
7. F. Guo, Y. Mu, and Z. Chen, "Identity-Based Encryption: How to Decrypt Multiple Ciphertexts Using a Single Decryption Key," in Proceedings of Pairing-Based Cryptography (Pairing '07), ser. LNCS, vol. 4575. Springer, 2007, pp. 392–406.
8. Amrita Sahu, Yogesh Bahendwar, Swati Verma, Prateek Verma, "Proposed Method of Cryptographic Key Generation for Securing Digital Image", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, October 2012.
9. H.Abdulzahra, R. AHMAD, and N. M. NOOR, "Security enhancement; Combining cryptograhy and steganography for data hiding in images," ACACOS, Applied Computational Science,pp.978-960,2014.
10. M. H. Rajyaguru, "Cryptography-combination of cryptography and steganography with rapidly changing keys ,"Interntional Journal of Emerging Technology and Advanced Engineering,ISSN ,pp.2250-2459,2012.
11. D. Seth. L. Ramanathan, and A.Pandey, "Security enhancement; Combining cryptography and steganography," International Journal of Computer Applications(0975-8887)Volume,2010.
12. Kranthi Kumar K, Devi T,(2018). Secured Data Transmission in Cloud Using Hybrid Cryptography. International Journal of Pure and Applied Mathematics, 119(16), 3257-3262.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.165

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details