TokChat: Tokenization of Text for Secured Peer-to-Peer Communication

Praneeth Vadlapati
VIT-AP University, India
praneeth.18bce7147@vitap.ac.in
ORCID: 0009-0006-2592-2564

Abstract: This paper presents TokChat, a system that utilizes a new approach designed for enhanced security in peer-to-peer communication using text messages. To improve the security of messages in a conversation, the system converts a message into tokens by leveraging existing pre-trained tokenizers that are commonly used across numerous natural language processing (NLP) tasks. The tokens represent a compact numerical form of text that captures the original message. A new private key is generated for every new conversation to ensure the confidentiality of the messages in the conversation. The tokens are encrypted using an existing AES-based cryptographic encryption approach, ensuring they can be securely transmitted to the recipient. The receiver receives the encrypted tokens, which are decrypted using the conversation-specific key. The decrypted tokens are converted to readable text using the tokenizer. The usage of the system ensures that the messages fail to be converted into readable text by successful attackers, even with the usage of the private conversation-specific key, without being aware of the tokenization process. The system offers a higher security standard for secure messaging in the use cases in which the confidentiality of the messages is crucial. The system has been tested using multiple text values and successfully tokenized the text, encrypted the tokens, decrypted the transmitted ciphertext, and reproduced the original text. The code is available at github.com/Pro-GenAI/TokChat.

Keywords: natural language processing (NLP), tokenization, cryptographic messaging, secure communication

I. INTRODUCTION

Securing digital communication is an integral part of safeguarding sensitive information that is exchanged over networks [1]. Traditional messaging applications often utilize server-based encryption procedures that might be vulnerable to security risks [2], [3]. Existing peer-to-peer communication might be vulnerable to attacks, and the successful attackers might have the ability to access the confidential text [4].

A. Proposed system and its benefits

The paper proposes a system called TokChat to mitigate the risks in confidential peer-to-peer communications without utilizing servers that involve centralized data processing. The system utilizes the process of tokenization, which is commonly utilized in natural language processing (NLP) tasks. A new private key is created for every new conversation. The tokens are encrypted using the private conversation-specific key. The encrypted tokens could be transmitted through any existing secure method to the recipient and could not be deciphered or decoded by the attackers during transmission. The recipient decodes the encrypted tokens and decodes the tokens into readable text. This method ensures secure communication, which is crucial in numerous critical cases, such as military communication during wars, confidential government communication, and secure corporate communication.

B. Related work

A substantial amount of work exists on peer-to-peer (P2P) protocols, end-to-end encryption, homomorphic encryption, allowing an encrypted transmission of text [5], [6], [7], [8], [9]. However, they do not ensure the confidentiality of messages in cases where private keys get compromised. Tokenizers have not yet been implemented

for encryption in communications. TokChat experiments on converting text into tokens that could not be easily decoded without knowledge of the tokenization process. TokChat utilizes existing encryption methods to encrypt tokens that are generated from the original text.

II. METHODS

A. Selection of a tokenizer

TokChat utilizes a tokenizer to generate tokens based on original text. The tokenizer of the language model GPT-2 [10] is used for the experiment by considering the optimal performance of the tokenizer.

B. Generating keys for a new conversation

For every new conversation, a new unique 256-bit key is generated, derived from a cryptographically secure pseudo-random generator. This key is used in steps that follow to allow a unique security for each conversation.

C. Converting messages into tokens

Sample text messages of a user are created by covering a wide range of test cases, such as alphabets, special characters, and numbers in both short and long sample values. Each user message is converted into tokens by utilizing the selected tokenizer. Tokens represent a compact numerical representation of text using the redundant patterns that exist in text. Tokens facilitate efficient encoding to ensure that each unique text fragment is translated into a token number.

TABLE I. SAMPLE TEXT VALUES

Index	Text
1	“A short text”
2	“Hello, this is a secure message.”
3	“Text with alphabets as well as numbers such as 1234567890”
4	“Text that includes special characters !@#$%^&*()_+-=[]y|;:,.<>?/”
5	“A very long message that contains numerous words and sentences, including punctuation. This message is long enough to be split into multiple lines.”

D. Encryption of tokens

The tokenized message is encrypted to transform into a ciphertext using “AES encryption [11], [12] in Cipher Feedback (CFB) mode [13]”. The process involves generating a random 128-bit “Initialization Vector (IV) [14]” value, which is used along with the shared conversation-specific key to encrypt the text. The IV value is concatenated with the ciphertext and transmitted to the recipient. The usage of the IV value ensures that each encryption process is unique and enhances the security of the data during transmission.

E. Decryption of tokens

Encrypted tokens are received by the recipient. The IV values are extracted from the received ciphertext upon receiving each encrypted message. The IV value and the shared conversation key are used to decrypt the ciphertext into tokens. This step ensures the reversal of the encryption process to get the original sequence of tokens.

F. Decoding tokens into messages

The decrypted token sequence is converted back into readable text using the tokenizer. The text is compared with the original text to test whether the system has decoded the text correctly to reproduce the original text. The validation allows the testing of the accuracy of the system.

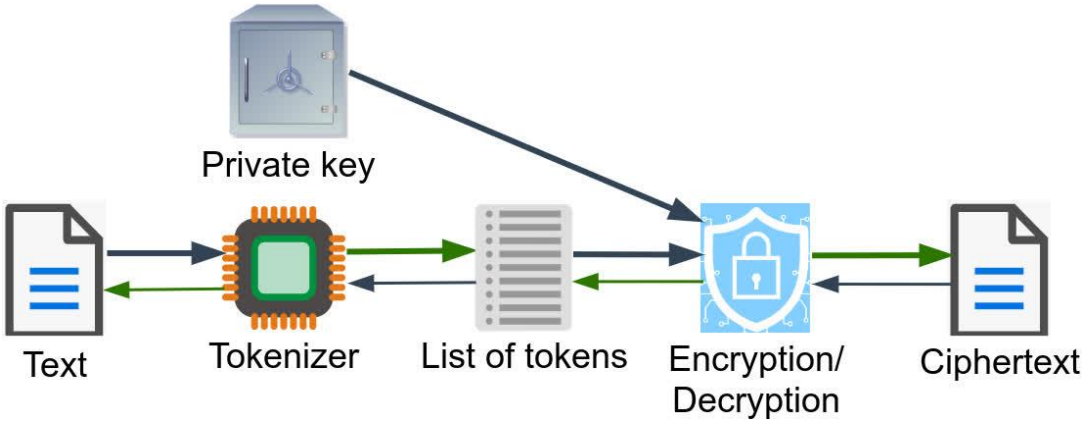

Fig. 1. Workflow of TokChat

III. RESULTS

A. Tokenization results

The text has been successfully converted to tokens using the tokenizer. The tokens generated by the tokenizer based on each sample text value are mentioned below.

TABLE II. TOKENIZATION RESULTS

Index	Tokens
1	[32, 1790, 2420, 1988]
2	[15496, 11, 428, 318, 257, 5713, 3275, 13]
3	[8206, 351, 435, 746, 397, 1039, 355, 880, 355, 3146, 884, 355, 17031, 2231, 30924, 3829]
4	[8206, 326, 3407, 2041, 3435, 5145, 31, 29953, 4, 61, 5, 9, 3419, 62, 10, 12, 28, 21737, 88, 91, 26, 45299, 29847, 29, 30, 14]
5	[32, 845, 890, 3275, 326, 4909, 6409, 2456, 290, 13439, 11, 1390, 21025, 2288, 13, 770, 3275, 318, 890, 1576, 284, 307, 6626, 656, 3294, 3951, 13]

B. Encryption results

The tokens have been successfully encrypted and converted to ciphertext. The encrypted ciphertext values generated using the tokens are mentioned below.

TABLE III. ENCRYPTION RESULTS

Index	Ciphertext
1	"0oL1gpAH1XsTCnDR_jEmLtAak4V72Zd0"
2	"D-oncVApg0h_VfO9aOUhr34nBqnDesGHTP9rvdRieFY="
3	"Nhf76dJ1SB4NSHC6PHfvQLGd52r7mKLT8V_2gyW3LkjmmCO1utTd0ZQiCs--cywR"
4	"D2HzGVz8Acsl5WWXsnkPrT0KH-UiYpFP8gByl5eGc90E7Ec59gCeT_7fMF-diamRxX1oFdl__KryqdiqZx_QLSH4q8s="
5	"Ven4Gcr6SaaKl0MN9gzu22CAt-gIsrvPLQNRaaO5wXaBLQHMSD0AsULq88CV3vCk-RmIzOB8QAHtk_NdNT-7dkTYTjBvsQ=="

C. Validation results

The ciphertext of all test cases was successfully decrypted back into tokens. The tokens were decoded into the original form of text that is in readable form. The decoded text successfully matched the original text for all test cases. The decoded text values in all test cases are mentioned below.

TABLE IV. VALIDATION RESULTS

Index	Decoded text	Validation result
1	"A short text"	Successful
2	"Hello, this is a secure message."	Successful
3	"Text with alphabets as well as numbers such as 1234567890"	Successful
4	"Text that includes special characters !@#$%^&*()_+-=[]y|;:,.<>?/"	Successful
5	"A very long message that contains numerous words and sentences, including punctuation. This message is long enough to be split into multiple lines."	Successful

IV. DISCUSSION AND LIMITATIONS

While TokChat presents a new addition to existing approaches to secure peer-to-peer communication, it relies on a unique key for each session, requiring both users to exchange the keys by utilizing an existing secure method to ensure a secured transmission. While tokenization enhances the security of communications, additional computational resources are required to execute the application efficiently. Tokenization adds an extra latency to the application. The usage of custom tokenizers might improve the efficiency and speed of the system.

V. CONCLUSION

TokChat demonstrates a token-based encryption method in peer-to-peer communication, which can provide a robust solution for securing sensitive and confidential digital communications. It reduces the reliance on centralized servers and the risk of the confidential text being readable by attackers who successfully get the conversation-specific private key. TokChat enhances the security of the existing communication methods by utilizing tokenization, which is used in natural language processing. Future research could explore the application of TokChat in multi-party conversations. Future work could involve the usage of a custom tokenizer that is not publicly available. Future research could involve encrypting using the last two messages in the conversation. TokChat offers a novel advancement in secure communication, enabling enhanced privacy and efficiency in digital messaging systems.

REFERENCES

[1] A. K. Jain, S. R. Sahoo, and J. Kaubiyal, "Online social networks security and privacy: comprehensive review and analysis," Complex & Intelligent Systems, vol. 7, no. 5, pp. 2157–2177, Jun. 2021, doi: 10.1007/s40747-021-00409-7.

[2] R. Dixit and R. Kongara, "Encryption techniques & access control models for data security: A survey," International Journal of Engineering and Technology(UAE), vol. 7, pp. 107–110, Jan. 2018, doi: 10.14419/ijet.v7i1.5.9130.

[3] F. Thabit, A. P. S. Alhomdy, A. H. A. Al-Ahdal, and P. D. S. Jagtap, "A new lightweight cryptographic algorithm for enhancing data security in cloud computing," Global Transitions Proceedings, vol. 2, no. 1, pp. 91–99, Jun. 2021, doi: 10.1016/j.gltp.2021.01.013.

[4] P. Wlazlo et al., "Man-in-the-middle attacks and defence in a power system cyber-physical testbed," IET Cyber-Physical Systems: Theory & Applications, vol. 6, no. 3, pp. 164–177, Jun. 2021, doi: https://doi.org/10.1049/cps2.12014.

[5] P. Wanda, Selo, and B. S. Hantono, "Model of secure P2P mobile instant messaging based on virtual network," in 2014 International Conference on Information Technology Systems and Innovation (ICITSI), Feb. 2015, pp. 81–85. doi: 10.1109/ICITSI.2014.7048242.

[6] M. A. Mohamed, A. Muhammed, and M. Man, "A Secure Chat Application Based on Pure Peer-to-Peer Architecture," Journal of Computer Science, vol. 11, no. 5, pp. 723–729, Jun. 2015, doi: 10.3844/jcssp.2015.723.729.

[7] B. Shaji, G. Abraham, V. E. S, and V. S. Sekhar, "Secure Peer-To-Peer Messenger and File Sharing Over IPV6," International Journal of Engineering Research & Technology (IJERT), vol. 09, no. 07, Jun. 2021, doi: 10.17577/IJERTCONV9IS07010.

[8] T. Melo, A. Barros, M. Antunes, and L. Frazão, "An end-to-end cryptography based real-time chat," in 2021 16th Iberian Conference on Information Systems and Technologies (CISTI), Jul. 2021, pp. 1–6. doi: 10.23919/CISTI52073.2021.9476399.

[9] M. L. Gaid and S. A. Salloum, "Homomorphic Encryption," in Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2021), A. E. Hassanien, A. Haqiq, P. J. Tonellato, L. Bellatreche, S. Goundar, A. T. Azar, E. Sabir, and D. Bouzidi, Eds., Cham: Springer International Publishing, May 2021, pp. 634–642.

[10] HF Canonical Model Maintainers, "GPT-2 [Language model]," Feb. 2019, Hugging Face. doi: 10.57967/hf/0039.

[11] F. J. D'souza and D. Panchal, "Advanced encryption standard (AES) security enhancement using hybrid approach," in 2017 International Conference on Computing, Communication and Automation (ICCCA), Dec. 2017, pp. 647–652. doi: 10.1109/CCAA.2017.8229881.

[12] J. Nechvatal et al., "Report on the Development of the Advanced Encryption Standard (AES).," J Res Natl Inst Stand Technol, vol. 106, no. 3, pp. 511–577, Jun. 2001, doi: 10.6028/jres.106.023.

[13] "Information Supplement: Cryptographic Key Blocks," Jun. 2017, PCI Security Standards Council. [Online]. Available: https://listings.pcisecuritystandards.org/documents/Cryptographic_Key_Blocks_Information_Supplement_June_2017.pdf

[14] E. Conrad, S. Misenar, and J. Feldman, "Chapter 5 - Domain 5: Cryptography," in Eleventh Hour CISSP (Second Edition), E. Conrad, S. Misenar, and J. Feldman, Eds., Boston: Syngress, Jan. 2014, pp. 77–93. [Online]. Available: https://doi.org/10.1016/B978-0-12-417142-8.00005-4