REAL-TIME SMS TRANSMISSION USING WEBSOCKET WITH END-TO-END ENCRYPTION (E2EE)

M.GUNASEKAR¹, V.DHINESH², N.DHISWANTH³, R.REVANTH⁴, G.SRI VIGNESH⁵

¹Assistant professor, ²,³,⁴,⁵UG Student
Department Of Information Technology
M.Kumarasamy College of Engineering
Thalavapalayam, Karur, Tamilnadu, India

¹gunasekarm.it@mkce.ac.in, ²dhinesh5124@gmail.com, ³dhiswabai55@gmail.com, ⁴revanthramesh578@gmail.com, ⁵ssrivignesh2000@gmail.com

ABSTRACT

With the development in communication technology, real-time messaging is improved to greater levels. We transformed from sending letters to Electronic mail in two decades. And with the addition of smartphones, sending someone a message is extremely fast. These smartphones questioned privacy in the beginning stage. That’s when encryption comes into play. However, we still receive important information like One Time Password (OTP) to our cellular inbox and not in WhatsApp or Facebook messenger. When we travel to different places we usually take our cellular sim card in order to get things done. But we cannot be so sure that the country we travel to have a good network coverage for our cellular. Even if it’s present we end up paying more than we pay in our native country. Hence cellular sms must be transmitted over the internet. With the help of websockets we can send and receive cellular sms with low latency. All we need is a smartphone and access to the internet. The cellular sms is sent using one smartphone and received by another so that we can be free of additional charges charged by other countries.

Keywords: One Time Password, Cellular sms, Encryption, Smartphone, Electronic mail, Low Latency, Network, Websocket.

I. INTRODUCTION

We use smartphones to send and receive text messages. Some of the popular applications like WhatsApp, Telegram are used by billions of people. So when someone has to go to abroad country, they can still communicate with friends and family using WhatsApp. However, they can’t receive OTP’s from banks and government organizations. Developed countries like United States offer International roaming for selected cellars with plans that are expensive. What if a person lost his SIM card or accidentally damage the SIM card. Then it is a problem for them to make bank transactions, verify phone numbers, authorize social media sites etc.

We propose a software solution called “Emitter” which will end the troubles experienced by people living abroad.

A person living abroad leaves his SIM card with his neighbor and his neighbor insert this one as a secondary SIM in his phone. Whenever a cellular message is received to that sim card, it will be read, encrypted and sent to the database. This person use Emitter. It will react to changes in the database and as soon as a new message is added to the database, it will get that encrypted message and decrypt it locally and display it to the user.

II. RELATED WORK

2.1 The XMPP protocol

XMPP is an open source protocol and it is also an extensible protocol which was originally called as Jabber. It is primarily used for transmitting current information as data. This is a messaging protocol, meaning that it is used
to design softwares which are messaging platforms. It can be applied to protocols such as HTTP, WebSocket or TCP/IP. One of the popular application using XMPP is WhatsApp.

2.2 Websockets

WebSocket is a protocol used for messaging. The first step is for the client to send an HTTP request to the server asking to open a connection. If the server agrees it will send a 101 switching protocol response. At which point, the handshake is complete. The TCP/IP connection is left open allowing bi-directional messages to pass between two parties with very low latency. The connection will stay open until one of the parties drops off. And the TCP/IP resources can be unallocated. It is often referred to as full duplex connection.

2.3 Sending and receiving binary data

In an application like online score board, the users want the scores to update automatically. If we use HTTP for this application then the user has to update the web page every few seconds. This is a major drawback of HTTP connection. With WebSockets, there is no need to update the page once it is loaded. For example, if the server sends 500 KB of payload, then the client will do the processing only when the whole payload is received to the client. After receiving, it is converted to one of the two types based on what kind of data is received. For data based messages it is converted to DOM string. For binary data it is converted to blob object. XMPP has limitations in sending binary data. If the application we are targeting has a lot to do with binary data, then WebSockets would be the right protocol.

III. PROPOSED SYSTEM

A methodology using Firebase is designed to store and retrieve messages using End-to-End Encryption algorithm. End-to-end encryption means that the message is encrypted before it leaves sender’s device, and it is only decrypted when it reaches the receiver you are messaging. So while in transmit, your messages are encrypted and cannot be read.

Methodology steps:

- Create the frontend for sender and receiver
- Setup Firebase Authentication
- Setup Firebase cloud firestore
- Create collection for Users and Messages
- Listen to incoming messages on sender side
- Display the message on receiver side

IV. IMPLEMENTATION

4.1 Platform used:

We used Flutter framework and Firebase cloud firestore database. The first step is to install Flutter and then connect the flutter app to firebase.

4.2 User Authentication:

A User can Sign in / Sign up using two ways.

- Sign in using Google
- Sign in using Email and Password

Firebase authenticates users using Json Web Tokens.
4.3 Encryption and Decryption:

End-to-end encryption (E2EE) asserts encryption and decryption to take place at the clients’ devices (end destinations). This allows the ciphered texts to transmit via an insecure and public channel, such as the internet, without the messages being compromised. It requires a public and private key pair. First step is to exchange the public key between two parties. Normally, Public key is stored in Database and it will be send upon request.

The below diagram illustrates the public key exchange process. (Orange color denotes Alice and Blue denotes Bob).

The below diagram illustrates the sending and receiving of messages between two users.
4.4 System Architecture:

V. RESULTS

In this work we have ensured real time sms transmission using secure Authentication using JWT. We have used End to End encryption for message transmission so that even we, developers of the project could never be able to read user’s messages.

VI. CONCLUSION AND FUTURE WORK

Employee’s must not struggle when they lose or damage their SIM card when going for an Onsite location. Our application will help make things easier without the hustle of network issues from Internet Service Providers in abroad countries and with a minimum cost. This application can be a solution for various other situations too. It can be extended in ways we developers may not expect.
REFERENCES