**T.I.M.**

Tiger Instant Messaging using UDP

Phuong Do  
Computer Science   
 University of Memphis  
 Memphis, TN, USA  
[kpdo@memphis.edu](mailto:kpdo@memphis.edu)

Kareem DaSilva   
Computer Science   
University of Memphis  
 Memphis, TN, USA  
 [kdasilva@memphis.edu](mailto:kdasilva@memphis.edu)

Rick DeSaussure  
Computer Science   
 University of Memphis  
 Memphis, TN, USA  
rldsssre@memphis.edu

**ABSTRACT**

One of the main benefits of the Internet is the communication it provides, being able to send data to other devices connected to the internet allows users to communicate effortlessly and more effectively than any other medium e.g. mail, phone, or mail pigeon. Tiger instant Messaging aims to utilizes the underlying protocols of the internet. T.I.M. goal is to design and implement a chatting tool with reliable data transfer using UDP.

In this paper we will discuss how we implement an instant messaging tool using UDP. Why we picked UDP as the protocol to accomplish this. The frameworks and tools we used to build out the GUI and technology implementation process.

1.0 **Introduction**

Building the Tiger Instant Messaging application is a challenge because we have to UDP to transfer the information. UDP is best used to send reliable data to the server and back. The main features of the application is to be able to send messages to a server with one to many clients. It will have support to send files through the server. This is an important feature since file transfer is one of the primary benefits of messaging platforms.

1.1**UDP**

UDP stands for User Datagram Protocol — a datagram is the same thing as a packet of information. The UDP protocol works similarly to TCP, but it throws all the error-checking stuff out. All the back-and-forth communication and deliverability guarantees slow things down.  
  
When using UDP, packets are just sent to the recipient. The sender will not wait to make sure the recipient received the packet — it will just continue sending the next packets. If you are the recipient and you miss some UDP packets, too bad — you can not ask for those packets again. There is no guarantee you are getting all the packets and there is no way to ask for a packet again if you miss it, but losing all this overhead means the computers can communicate more quickly.  
  
UDP is used when speed is desirable and error correction is not a priority.

1.2**Python and tkinter**

Python is a powerful language which allows us to its simple to read syntax and its powerful libraries to build T.I.M. Python includes all networking based functions built directly in the programming language using the frame #include protocol. In order to build the graphical user interface T.I.M uses the Python’s Defacto GUI framework called Tkinter. Tkinter allows to build really sleek looking GUI’s by programming components using Python. By using Tkinter we model T.I.M over a standard messaging platform. The layout will consist of confirm send button as well as an emoji select button. Tkinter also allows us to interact with desktop. For File Transfer we will be able to drag and drop files or select file from the host computer.

2.0**Peer 2 Peer messaging**

In order to keep a chat history, T.I.M records every message to a chat history text file. Each entry has a format of user writing the message, time the message was received by server, and the actual message written. Its is integral we have a chat history

In order to keep a chat history, T.I.M records every message to a chat history text file. Each entry has a format of user writing the message, time the message was received by server, and the actual message written. Its is integral we have a chat history so we can keep track of message send and received. This feature will allows to build more advance features like read receipts. A popular feature that is popularize in most Internet messaging application that tells the user when their message has been read by the intended recipient.

2.1**Emoji Support**

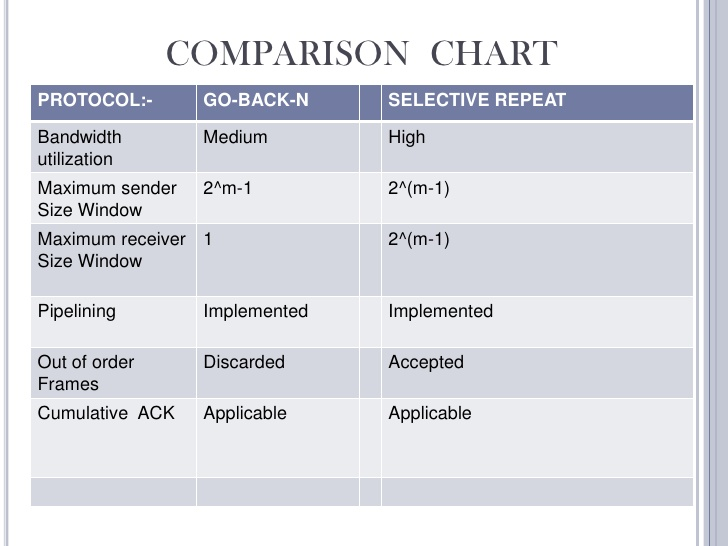
Emojis are a great way to express yourself through messaging. Since 2018 almost all Smartphones have emoji support enabled. The T.I.M. application allows the same support for emoji. By encoding emojis in unicode we are able to support up to 64 emojis on T.I.M. In terms of reliable sending emojis. After the server receives the emoji unicode. We display the image by referencing the code with image stored on the server. This allows users to instantly see the emoji without any latency. This is a better approach then downloading or streaming packets of the unique emoji image to the server, which would show the message then transform into the emoji image.

2.1**File Transfer**

File Transfer is a convenient feature in messaging platform due to its common case scenario of communicating through text and transferring relatable files using the same tool. To accomplish File Transfer with UDP/TCP T.I.M has to use a unique protocol to transfer files. The protocol is called selective repeat which accepts frames out of order but passes packets to the network layer in order. Each frame is associated with each frame consider a timer. When the timer expires.

2.2**Image Transfer**

Image transfer is very similar to File Transfer but has less use cases than the file transfer. Similarly to the file transfer we use the selective repeat protocol to ensure that all packets are kept together. The Image transfer take its a step further by adding image support. There are multiple formats of images out there but for T.I.M we are only supporting jpg and png images, since these are the most common image extensions. By simply using a Python library we can convert the bits to images and display them using Tkinter.



3.0 **Test Cases**

Tkinter is Python's de-facto standard GUI package. It is a thin object-oriented layer on top of Tcl/Tk. T.I.M.

In order to ensure packets are delivered safely and/or not corrupted we use the checksum method.  
  
Tiger Instant Messaging using multi threading in order incorporate all features intended without using the same thread. We use four threads to accomplish. The first thread is used to for file transfer. The second thread is used for sending image transfer. While the third thread is used for messaging. The main thread or the fourth thread is responsible for the ACK.  
  
UDP is used when speed is desirable and error correction is not a priority.

4.0 **Conclusions**

In conclusion, T.I.M is a standard messaging platform that use fundamental networking knowledge to accomplish modern messaging features. By using UDP T.I.M is able to perform peer 2 peer messaging, send emojis, file transfer with image support, and keep a chat history. The biggest takeaways from building Tiger Instant Messaging is the networking protocols to ensure packets were data were correctly sent and have the proper measures when we lose the packets. UDP is a powerful protocol that build large scale application however building a chat application can run into some complications. Since UDP doesn’t keep track of packets lost sending messages might to be proven difficult. However with the right procedures you can still accomplish Tiger Instant Messaging using UDP.

**REFERENCES**

[1] FNM Surname (2018). Article Title. Journal Title, 10(3), 1–10.

[2] F.N.M Surname, Article Title, https://www.acm.org/publications/proceedings-template.

[3] F.N.M Surname and F.N.M Surname, 2018 Article Title, The title of book two (2nd. ed.). Publisher Name, City, State, Country.

[4] Ian Editor (Ed.). 2018. *The title of book two* (2nd. ed.). University of XXX Press, City, Chapter 100. DOI: <http://dx.doi.org/10.1000/0-000-00000-0>.