PROTOTYPE OF A DESKTOP-BASED COMMUNICATION APPLICATION USING A DELAY TOLERANT NETWORK ARCHITECTURE

Slameta, Tata Supriyadi[1)], Ashari[2)], Muh Haikal Pandia Situmorang[3)], Kamal Falah Firdaus[4)], Griffani Megiyanto Rahmatullah[5)]

Department of Electrical Engineering, Politeknik Negeri Bandung

Jl. Gegerkalong Hilir, Ciwaruga, Kec. Parongpong, Kabupaten Bandung Barat, Jawa Barat 40559, Indonesia

e-mail: slameta@polban.ac.id[1)], tata.supriyadi@polban.ac.id[2)], muh.haikal.tkom418@polban.ac.id[3)], Kamal.fa-lah.tkom19@polban.ac.id[4)], griffani.megiyanto@polban.ac.id[5)]

Corresponding Author: Slameta

ABSTRACT

Currently, the process of exchanging information or data is carried out quickly, precisely, effectively and efficiently using the internet network. However, in reality, there are still some problems in areas that are difficult to get a signal, such as connection loss or unstable connection. One of the proposed solutions to anticipate this problem is the implementation of a network using the Delay Tolerant Network architecture. This architecture was chosen because of its basic capability, namely being able to send files even in a high delay state or used in areas with intermittent properties. The implementation carried out is making a chat application prototype that is able to function in intermittent areas and can be used for communication between sender and receiver in difficult signal conditions. The hardware used is a raspberry pi and supporting software using ION-DTN and traffic management designed using the python programming language. The application is tested by applying the concept of functionality testing, namely with full signal condition scenarios and limited signal conditions as a representation of the intermittent area. The test results show that the application is able to function based on the initial design with a delay with test variations of 1-10 minutes of signal dropout time. Further development can be done by adding a chat feature so that it is able to send more varied data.

***Keywords**: Chat, Delay Tolerant Network, Intermittent, ION-DTN*

I. INTRODUCTION

The commonly used data communication technology is Transmission Control Protocol (TCP) because it is reliable, especially in sending and receiving data. The protocol is included in the Internet architecture, the Transmission Control Protocol / Internet Protocol (TCP / IP) architecture. TCP technology has the nature of connection-oriented data transmission, i.e. communication can occur if the communication path has been established between the sender and the receiver. This is also assisted by the acknowledgement process and the retransmission process if there is data that is not sent. However, the problem arises when there is a disconnection due to weak signal strength as a communication medium for the sender and receiver. If this condition occurs, the communication process will be given a timeout value as a time reference for retransmission. The value of the timeout is the problem because it is often found that the timeout is short and results in failure to send. This condition is often referred to as an intermittent condition [1].

Based on this, it is proposed to implement the application using a network with a Delay Tolerant Network (DTN) architecture to minimize the occurrence of failures in data transmission so that data can be sent when the connection is reconnected. The choice of this architecture is because it is considered to have advantages, namely, data communication that can be carried out without being connection-oriented. Thus, DTN can reconnect with a predetermined time in the event of a connection drop.

II. THEORY

Research conducted by Kevin Fall states that a network with a DTN architecture can be applied using long delay time parameters and uncertain connections [2]. DTN is a network architecture to provide solutions for networks that have intermittent connectivity, long delays, different data rates and high error rates.

In general, the DTN architecture consists of all layers of the Internet architecture plus an additional layer, namely the bundle layer. The position of the bundle layer is between the Application and Transport layers, as shown in Figure 1. The bundle is the basic data unit in the form of variable sizes and signals used in DTN networks. The bundle layer is the main key used in DTN because it has the task of storing and forwarding all or parts of the bundle between nodes [3].

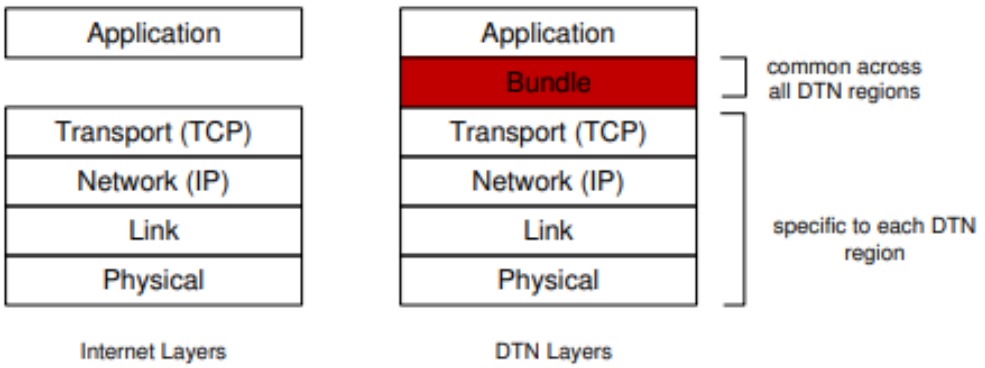


Figure. 1. Comparison of TCP/IP and DTN architecture [3]

Specifically, each DTN node in the network will send messages to other *nodes* over a route by the way messages are forwarded from source to destination. The difference between TCP/IP architecture or Internet architecture is the use of connection-oriented properties. The line of communication must be established in advance, and the sender must also know the path to reach the destination. If there is a disconnection, the possibility of packet loss will be high due to the absence of temporary storage for the packet being sent. On the other hand, the DTN architecture offers a different concept of not always assuming the availability of the network and stipulating that nodes will store bundles within a certain time interval. The process of saving and forwarding to the next node is called *store-carry-forward* [4].

There are several other pieces of literature that also discusses the use of DTN, namely using raspberry pi hardware components as sending and receiving nodes and wifi networks as transmission media, in software using IBR-DTN modules supported by python programming language. The test was carried out under full signal conditions and intermittent signals. The results showed that the communication system has successfully functioned with the farthest distance, namely 23 meters, with a 100KB bundle configuration and produced speeds ranging from 150KB/s [5].

Next, build a DTN network for public transportation using four routing protocols. The four routing protocols are Epidemic, Spray and Wait, MaxProp, and Prophet V2 using the ONE Simulator application. From the results of this study, it was found that MaxProp routing is the best routing in simulation results carried out with DTN networks but has a high latency value [6].

Other studies have also discussed applications for communication in challenging area conditions with the integration of DTN (Delay Tolerant Network) on public mobile phones. An IP network architecture that makes it easier to expand delay-tolerant applications by implementing the Bundle Protocol specification and plans to integrate social networking applications such as Twitter and Youtube [7]. The use of Delay Tolerant Network architecture network on the Internet of Things can also be used to overcome intermittent connections, which can improve data transmission ratio performance and data efficiency by providing a delay tolerance scheme [8].

III. SYSTEM DESIGN

The design is focused on the target of prototyping data communication applications using a DTN architecture equipped with a custom GUI to facilitate the sending and receiving of data. The assessment indicators measured are the functionality of sending and the integrity of the data on sending and receiving information within defined distances. The illustration of the research design system is shown in figure 2.

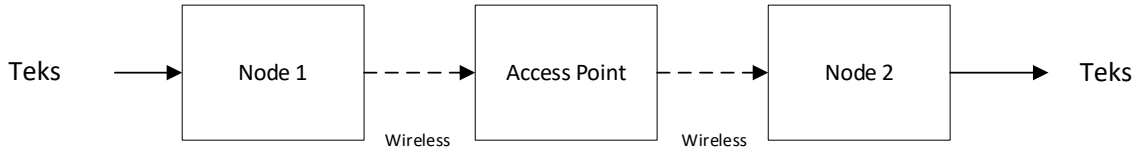
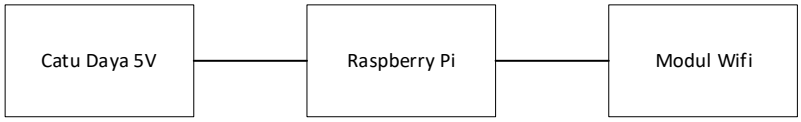


Figure. 2. Illustration of communication design

The hardware design of one of the clients includes *Wifi* and Raspberry modules, as shown in Figure 3. The system to be built will consist of the main component, namely Raspberry pi, as a component of the process of sending and receiving data. *The power* supply used is a 5v/2A power supply for raspberry pi. The wifi module is used to capture the signal from the access point on the connected raspberry network. Next is the system flowchart shown in Figure 4.



Gambar 3. Perancangan Perangkat Keras

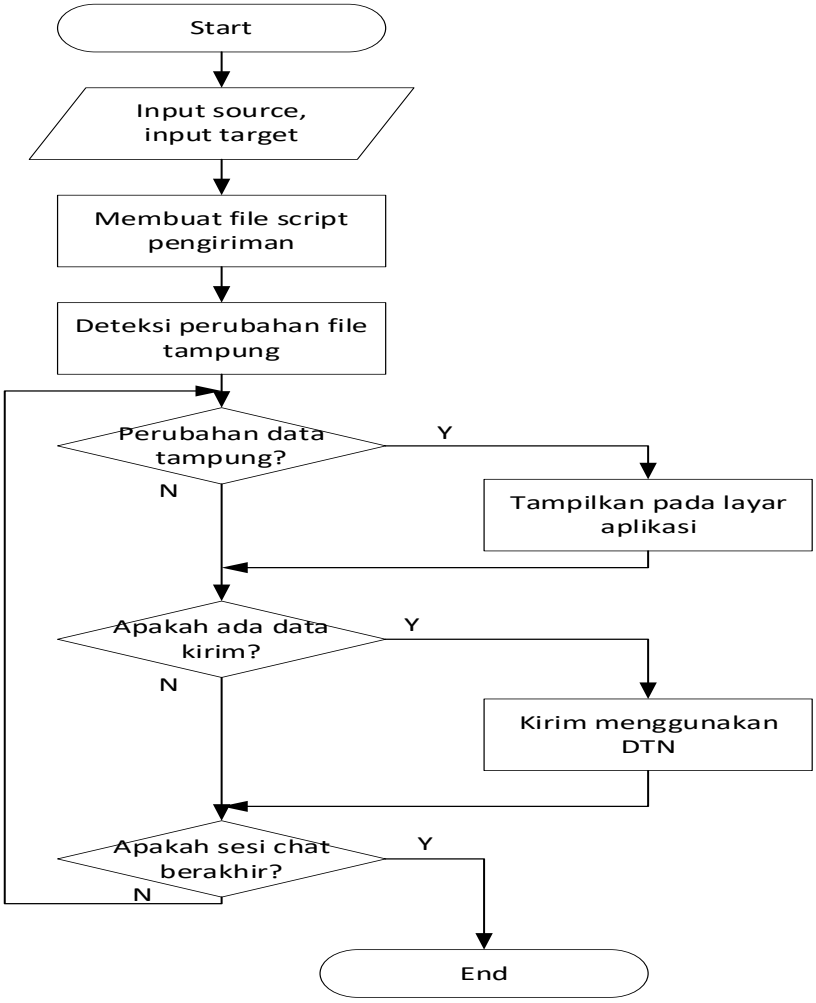


Figure. 4. Illustration of communication design

The process that is carried out starts by asking for the source and target inputs as parameters to be included in the DTN shell script. The source and target will be considered as IDs so that sending and receiving addresses can be more flexible and not limited to IP address names. Next is to carry out the process of detecting the container file, which acts as temporary storage of the data received from other clients. If there is a change in the tube data storage, it will be assumed to be data received from other clients and displayed on the application screen.

Next, the system will always detect whether there is data to be sent to other clients on the display of the chat box column with the "Enter" button trigger. If data is detected that is ready to be sent, the application will use the DTN shell script to send it to the destination client. Both processes (detection of receiving and sending) will always be carried out repeatedly until the session condition is ended by one of the clients.

IV. TEST AND ANALYSIS RESULTS

Testing was carried out to determine the functionality and performance of the *system* from the chat application prototype using the *Delay Tolerant Network* (DTN) architecture, whether or not it was in accordance with the design that had been made. Testing the functionality and performance of the system is carried out by sending and receiving data in the form of text. Next, system reliability testing is carried out by communicating when the signal condition is full, and the signal condition is broken. The full signal condition means that both devices are connected to each other in the network. While the signal break is a condition when one of the *nodes* or raspberries is in the condition that one of the devices is disconnected from the network. In more detail, such testing is carried out by disconnecting the Raspberry Pi from the network. All devices are connected to the same network by utilizing the wireless network emitted by the 2.4 GHz mobile hotspot. The test results are shown in figure 5 and figure 6.

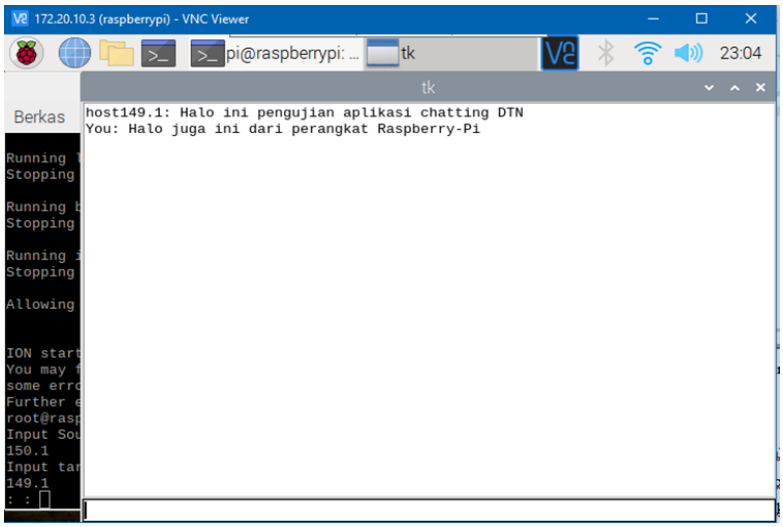


Figure. 5. Client test results 1

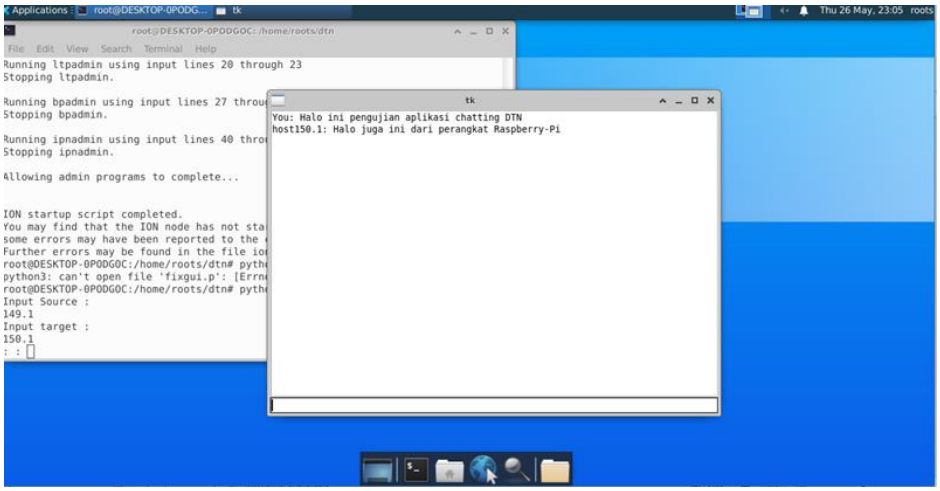


Figure. 6. Client test results 2

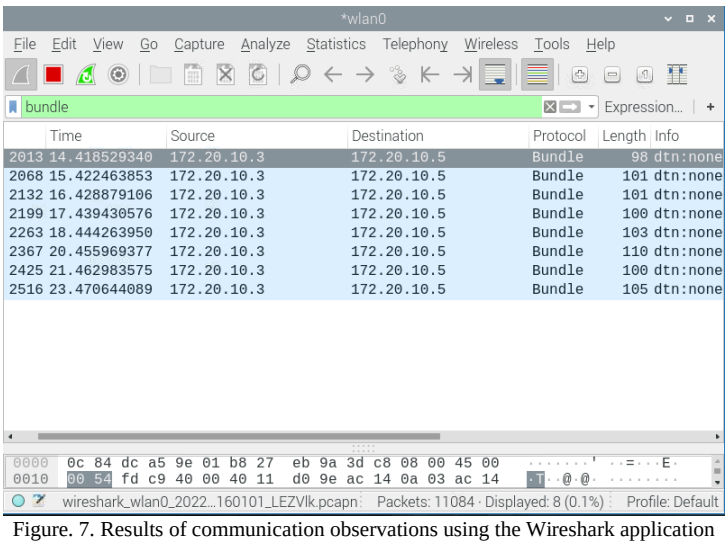


Figure. 7. Results of communication observations using the Wireshark application

The designed application is successfully implemented using the DTN architecture based on verification performed using the Wireshark application. Figure 7 shows that communication is carried out using the bundle protocol included in the bundle layer of the DTN architecture. The bundle works just like a buffer as temporary storage on each node to be passed on to the next node.

Next, system reliability testing is carried out, namely communicating in a disconnected network condition or conducting *an intermittent area* simulation. Testing of disconnection conditions is carried out with a *delay* time range of up to 10 minutes, with data sent in the form of text on the *chat* application. The disconnection situation is

shown in Figure 8, and the test results are shown in Table I.

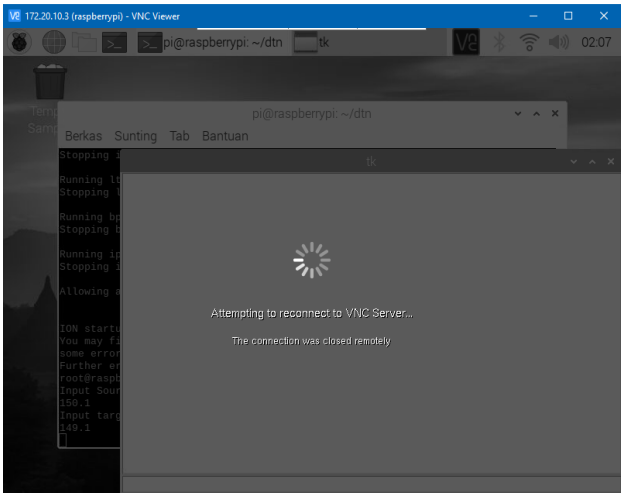


Figure. 8. Disconnection situation on one of the clients

TABLE I
TEST RESULTS WITH VARIATIONS OF DISCONNECTION TIME

Termination time connection (minutes)	Status
1	Successful delivery
2	Successful delivery
3	Successful delivery
4	Successful delivery
5	Successful delivery
6	Successful delivery
7	Successful delivery
8	Successful delivery
9	Successful delivery
10	Successful delivery

Table I. shows the reliability of the Delay Tolerant Network (DTN) architecture to communicate between devices in a disconnected state, such as in the intermittent area. With testing in the form of disconnection in 10 tests with a delay range of 1-10 minutes, communication can still be carried out with proof that sending text data for the chat sent can be received properly by the recipient. This indicates the reliability of the Delay Tolerant Network (DTN) architecture in overcoming intermittent connections and can be said to tolerate long delays.

V. CONCLUSION

The results of the study resulted in the conclusion that the application of the DTN network architecture to the designed chat application can be by utilizing temporary file storage to hold data communicated between sender and receiver. The reliability of the chat application prototype using the DTN architecture has been successfully tested with a test scenario of variations in the value of disconnection times of up to 10 minutes. Next, the chat application can be developed by sending a variety of data types and other complementary features.

ACKNOWLEDGMENTS

Acknowledgments to Unit Penelitian dan Pengabdian Masyarakat Politeknik Negeri Bandung with the help of research funding as stated in the letter number: B/114.129/PL1.R7/PG.00.03/2022.

REFERENCES

[1] L. Gao, S. Yu, T. H. Luan, and W. Zhou, Delay Tolerant Networks. Cham: Springer, 2015.
[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," Intel Research Technical Report, vol. I, 2003.
[3] S. D. Siswanti, “Pengembangan Sistem Aplikasi Pengiriman Data Daerah Terpencil Berbasis Delay Tolerant Network,” vol. 8, no. 2, pp. 240-242, 2013.
[4] A. Suharsono, “Pengertian dan Latar Belakang Delay Tolerant Network,” 2012. http://aswinsuharsono.lecture.ub.ac.id/2012/07/pengertian-dan-latar-belakang-delay-tolerant-network/
[5] G. M. Rahmatullah dan I. Fadillah, “Sistem Komunikasi Jaringan Wireless Menggunakan Raspberry Pi,” dalam Industrial Research Workshop and National Seminar, Bandung, 2020.
[6] H. E. Wahanani, I. M. Suartana dan D. Adityawati, “ANALISA KINERJA PROTOKOL ROUTING DELAY TOLERANT,” UPN “Veteran” Jatim, Jawa Timur, 2015.
[7] H. Ntareme, M. Zennaro dan B. Pehrson, “Delay Tolerant Network on smartphones: Applications for,” dalam Association for Computing Machinery, New York, NY, United States, New York, 2011.
[8] F. Z. Benhamida, A. Bouabdellah dan Y. Challal, “Using Delay Tolerant Network for the Internet of Things,” International Conference on Information and Communication Systems (ICICS), 2017.