Jurnal Mantik, 7 (3) (2023), ISSN 2685-4236 (Online)

Published by:Institute of Computer Science (IOCS)

Jurnal Mantik

Journal homepage: www.iocscience.org/ejournal/index.php/mantik

Remote server telegram bot using long polling method at PT. Ngampooz Pintar Sejahtera

Irawan[1], Muhammad Ainur Rony[2], Motika Dian Anggraeni[3], Suwaebatul Aslamiyah[4]
[1,2,3]Department of Information Technology, Budi Luhur University, Indonesia
[4]Department of Computer Engineering, Universitas Indraprasta PGRI, Indonesia

A R T I C L E I N F O

Article history:

Received Oct 25, 2023
Revised Oct 29, 2023
Accepted Nov 09, 2023

Keywords:

Long Polling;
Remote Server;
Telegram Bot;
Visibility.

ABSTRACT

Visibility is an important aspect of server management. A system administrator needs to have visibility into the servers under his management. A system administrator needs to have a vision of the current conditions on the servers under his control. With this visibility, you can see whether there is a problem with the server, and action can be taken to deal with the problem. One of the problems that occurs in companies or institutions that have servers to run public services is the lack of flexibility in the supervision process to continuously monitor server performance and condition 24 hours a day. Seeing these problems, it is necessary to have a system that can carry out monitoring and management remotely to increase the flexibility of system administrators. Telegram, which is a messaging platform, can be used as a remote management medium by utilizing the available bot features. The method used to communicate with the bot is the long polling method which is more efficient than the regular polling method. Using remote management via Telegram bot media can be an alternative way that system administrators can use to interact with the server. With this research, it is hoped that more efficient and flexible media or other methods can be developed for managing and monitoring servers.

Corresponding Author:

Motika Dian Anggraeni
Departement of Information Technology
Budi Luhur University
Jl. Ciledug Raya, South Jakarta, DKI Jakarta 12260
Email: motika.diananggraeni@budiluhur.ac.id

1. INTRODUCTION

With the rapid progress of internet technology today, services are developing that can be used by users online (Arisoemaryo et al., 2022). These services certainly require a server to run and be accessed by their users. A server is a computer system that has special services in the form of data storage. Data stored via the server is in the form of information and various types of complex documents (Harsono, 2022). The server's job is to serve everything connected to the network and is the main device in the network communication system to function as a service provider (Sholihah et al., 2020). The server is required to remain fully operational for 24 hours, so that the services running on it can continue to be accessed by users. The server's inability to operate will certainly disrupt the quality of the services running on it, so that the service cannot run properly or cannot even be accessed at all (Idhom et al., 2018).

Visibility is an important aspect of server management (Sulaeman, 2022). A system administrator needs to have visibility into the servers under his management. With the rapid progress of internet technology today, services are developing that can be used by users online. These services certainly require a server to run and be accessed by their users. A system administrator needs to have a vision of the current conditions on the servers under his control. With this visibility, you can see whether there is a problem with the server, and action can be taken to deal with the problem. One of the problems that occurs in companies or institutions that have servers to run public services is the lack of flexibility in the supervision process to continuously monitor server performance and condition 24 hours a day. Seeing these problems, it is necessary to have a system that can carry out monitoring and management remotely to increase the flexibility of system administrators. The theoretical benefit of this research is to provide a scientific contribution regarding the implementation of the long polling method and also the use of Telegram in creating remote servers. Practically, the results of this research can be used by PT. Ngampooz PIntar Sejahtera to help monitor and access the server. Telegram, which is a messaging platform, can be used as a remote management medium by utilizing the available bot features (Gunawan & Anshor, 2023). The method used to communicate with the bot is the long polling method which is more efficient than the regular polling method (Ferdiyan et al., 2021). The information that can be seen through this remote management bot is CPU usage, memory usage, and uptime (Syani & Saputro, 2021). Apart from monitoring the server, system administrators can also make changes to the server, namely adjusting the firewall status, turning on and off existing services, and scaling the server memory capacity (Prabakaran et al., 2022). Using remote management via Telegram bot media can be an alternative way that system administrators can use to interact with the server (Adam & Suryadi, 2022). With this research, it is hoped that more efficient and flexible media or other methods can be developed for managing and monitoring servers.

Telegram is an instant messaging application multiplatform (can be used on many devices) which is easy and free. Currently, there are approximately 500 million monthly active users. The Telegram application is also included in the 10 most downloaded applications in the world, as reported on the official Te-legram page (Normadhoni et al., 2021). Telegram itself has a bot feature, which allows users to create their own or run third-party applications within Telegram (Saad et al., 2021). Telegram with bot or robot features is programmed with various commands to carry out a series of instructions given by the user which are operated by software that has AI features (Shodiq & Fauzi, 2021). Users can interact with the bot via messages or commands, which the bot will respond to according to the program that has been created. This bot feature can be used as a connecting medium or middleware to the server, which can carry out monitoring and remoting (Halder et al., 2022).

There are several previous studies that used Telegram bots. Research conducted by Aisyah et al. The aim of this research is to produce e-learning based teaching material products using the Telegram bot application. After carrying out the development of online-based electronic modules with the Telegram bot program, it can be concluded that this development can answer the problems caused by the lack of teaching materials used (Aisyah et al., 2021). A research by Rahmatulloh and Firmansyah. This research will try to use technology in instant message as a means of academic service information, it is expected academic information can be delivered more quickly and up-to-date. With the making of this application, facilitate communication and delivery of academic information to lecturers, students, and the academic community (Rahmatulloh & Firmansah, 2019). A research by Parlika and Pratama, this study seeks to build a Bot telegram with database to built The Online Test application uses Telegram Bots Version 1.0. Conclusions from the results of making the chat bot is a telegram chat bot function that can provide multiple choice practice exercises where the display questions on each user will not be the same because in it there is a randomization algorithm called from the

database a set of questions that are distinguished according to their material (Parlika & Pratama, 2020). A research by Sucipto et al, this study examine the optimization of the MariaDB database on information systems in tracer studies. Database design will accommodate data from two application sources, namely webbased applications and telegram bots. Using two paths to the database to make it easier for users to register via a telegram bot, users can then access the tracer study questionnaire on web-based applications (Sucipto et al., 2019).

2. RESEARCH METHOD

There are two of the most popular bot management methods, as shown in Fig. 1 long-polling and webhooks. The bot management function is to share data efficiently between applications and application users (Sucipto et al., 2019). Both ways have advantages and disadvantages. The method used in this research is short polling. Long polling is a method of communication between the Telegram bot API and third-party software. Long polling is a technique where HTTP requests are made by the client to wait for events from the server (Candra, 2020). With the long polling method, the server will periodically check with the bot whether there are incoming messages. If there is an incoming message, the server will execute it based on the message request sent by the user (Aminudin et al., 2021). With traditional or short polling techniques, the client sends regular requests to the server and each request tries to "pull" any available events or data (Nilsson & Dunér, 2020). If no events or data are available, the server returns an empty response and the client waits for some time before sending the next request. The polling frequency depends on the latency the client can tolerate to retrieve the latest information from the server (Gorbenko, 2019). This mechanism has the disadvantage that the resources consumed (server and network processing) are highly dependent on the acceptable latency in sending updates from the server to the client. If acceptable latency is low, then the polling frequency may cause unacceptable load on the server, network, or both. In contrast to short polling, long polling minimizes latency in client-server message delivery and usage of processing and network resources (Ogundeyi, 2019).

One way to communicate with third-party software using the Telegram bot API is through long polling. Using a technique known as "long polling," the client sends HTTP queries to the server in order to wait for events. To ensure that the client receives updates right away, the connection is maintained. Upon receiving a message, the bot's system instantly comprehends its contents and can initiate commands based on the pre-established algorithm. Developers can customize a chatbot with the Telegram bot API to suit their needs. in order for incoming messages to be processed using the algorithm, utilized as input, and an output to be a reply message. For the lengthy polling strategy to function, a local computer is needed to manage (Ghozi et al., 2022). The long polling method is illustrated in Figure 1.

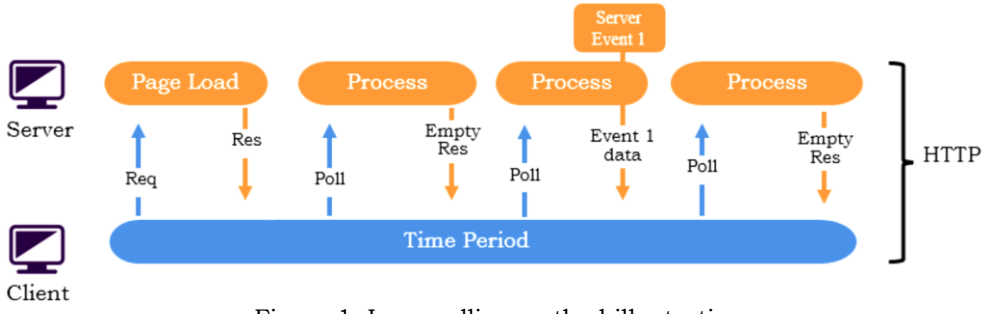

Figure 1. Long polling method illustration

3. RESULTS AND DISCUSSIONS

This research resulted in a remote server innovation that allows an administrator to easily monitor server conditions via instant messaging and also make changes to the server only via telegram messages.

3.1 Experimental Enviroment

Solution trials are carried out in a special environment to retrieve data from the trial results. The experimental environment is divided into two specifications, namely hardware specifications and software specifications.

3.2 Input Data

Input data was obtained from Telegram software on the Android platform version 5.12.1, typing was done manually.

3.3 Software Testing

The following are the solution testing steps which aim to find out whether the solution can work well and can solve the problem according to the existing problem formulation. To test this system, various messages were sent to the Telegram bot to activate the existing menus.

a. CPU Menu Testing

In this test, a message will be sent to the bot with the message "/cpu", to which the bot will reply with the percentage of CPU usage on the server. The CPU menu test results can be seen in Figure 1.

Figure 2. CPU menu testing

b. Active Firewall Access Testing

In this test, connectivity to the server will be checked using the ping command. If the firewall is active, the server cannot be successfully accessed using the ping command, as shown in Figure 3.

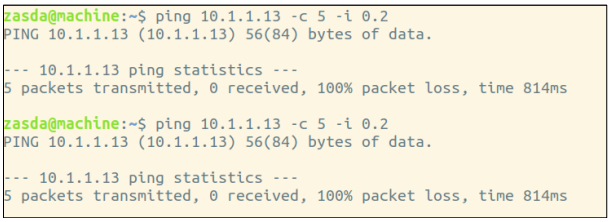

Figure 3. Active firewall access testing

c. Disabled Firewall Access Testing

In this test, connectivity to the server will be checked using the ping command. If the firewall is disabled, the server can be successfully accessed using the ping command, as shown in Figure 4.

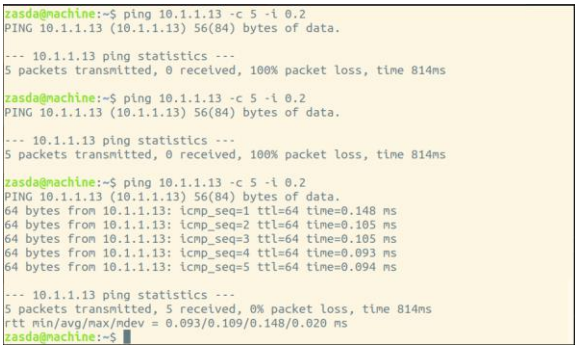

Figure 4. Disabled firewall access testing

d. Firewall Menu Test Disabled

In this test, the firewall will be deactivated via the Telegram bot. The test results can be seen in Figure 5.

Figure 5. Firewall menu test disabled

e. Menu Memory Testing

In this test, a message will be sent to the bot with the message "/memory". Memory resource testing is intended to determine memory usage on the server when

sending data to clients (Hardiwansyah et al., n.d.). The results of testing the Memory menu can be seen in Figure 6.

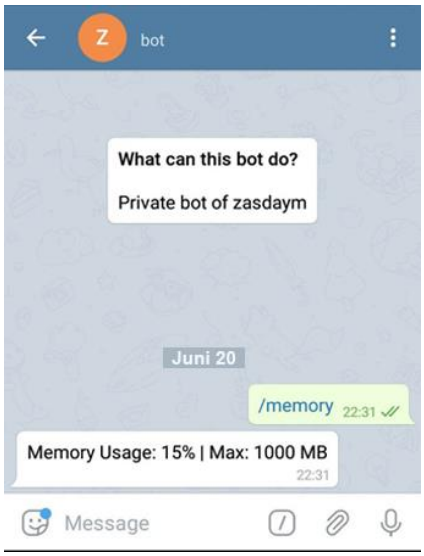

Figure 6. Menu memory testing

f. Menu Scale Testing

When the scale menu on the bot is run, the bot will provide two options to choose from. The test results can be seen in Figure 7.

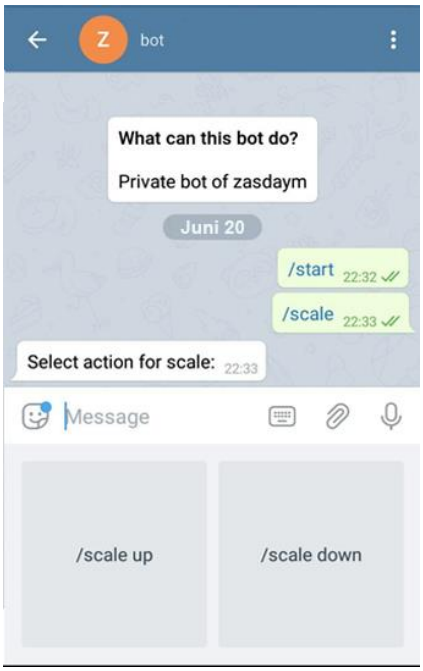

Figure 7. Menu Scale Testing

g. Scale Up Menu Testing

In this test, server capacity will be scaled up via the Telegram bot. Then a server memory check will be carried out to validate whether the scale up has run correctly. The test results can be seen in Figure 8.

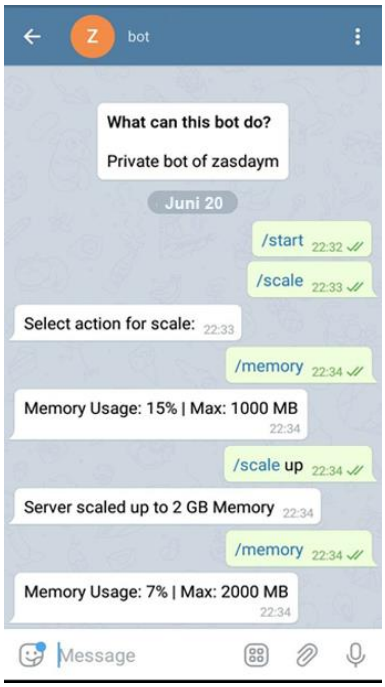

Figure 8. Scale up menu testing

h. Scale Down Menu Testing

In this test, server capacity will be scaled down via the Telegram bot. After carrying out the scale down command, the server memory will then be checked to validate whether the scale down has been carried out correctly. The test results can be seen in figure 9.

Figure 9. Scale down menu testing

i. Service Activation Testing

In this test, the service will be activated via the Telegram bot with the command "/service". The test results can be seen in Figure 10.

Fig 10. Sevice activation testing

j. Service Deactivation Testing

In this test, the service will be deactivated via the Telegram bot with the command "/service". The test results can be seen in Figure 11.

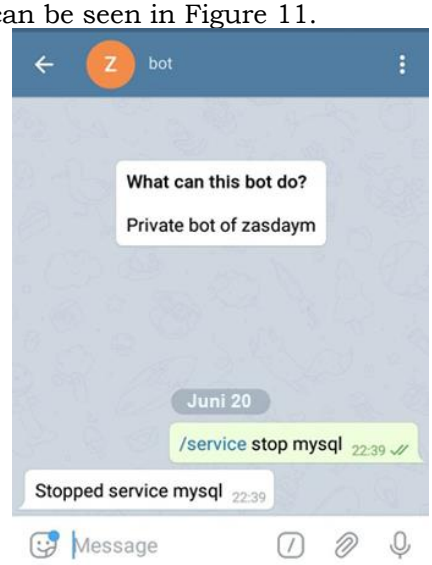

Figure 11. Service deactivation testing

3.4 Evaluation of Solutions

From several tests on the solutions that have been carried out, there are advantages and disadvantages. The advantages are Can monitor servers via Telegram media, which is instant messaging and can make changes to the server via chat. The disadvantages are the internet access is required so that solution can communicate with the telegram API, the commands that can be executed via bots are limited and the scalling feature is still done manually, it can't autimatically scale when needed.

Following are the advantages and disadvantages of the solution created: Solution Advantages(a) Can monitor servers via Telegram media, which is instant messaging. (b) Can make changes to the server via chat.

Lack of Solutions (a) Internet access is required so that the solution can communicate with the Telegram API. (b) The commands that can be executed via bots are still limited. (c) The scaling feature is still done manually, it cannot automatically scale when needed.

4. CONCLUSION

After carrying out the design and solution creation stages, which can then be continued with the testing and implementation stages, the following conclusions are : The research provides scientific contributions regarding the implementation of the long polling method and also the use of the Telegram instant messaging application in creating remote servers, With the Telegram bot solution, server monitoring can be easier because it can be accessed via chat media, server handling can be done quickly via the Telegram bot system and management can be decentralized to other team members, because server management is made easier with the Telegram bot system.

REFERENCES

Adam, S., & Suryadi, A. (2022). *BULLETIN OF COMPUTER SCIENCE RESEARCH Monitoring Notifikasi Status Services Pada Os Linux Menggunakan Bot Telegram*. *3*(1), 103–108. https://doi.org/10.47065/bulletincsr.v3i1.219

Aisyah, R. N., Istiqomah, D. M., & Muchlisin, M. (2021). *Developing E-learning Module by Using Telegram Bot on ICT for ELT Course*. *534*, 106–111.

Aminudin, M. R., Fitrani, A. S., & Rosid, M. A. (2021). *Indihome ' s New Post Activation Control Application Based On Website With Telegram Features Aplikasi Pengawalan Aktivasi Pasang Baru Indihome Berbasis Website Dengan Fitur Telegram*. *I*(I).

Arisoemaryo, B. S., Prasetio, R. T., Sanjaya, A. R., Adhirajasa, U., Sanjaya, R., User, E., Satisfaction, C., & Pengguna, K. (2022). *JAMSOSTEK MOBILE MENGGUNAKAN METODE*. *4*(1), 110–117.

Candra, H. (2020). *Designing a Chatbot Application for Student Information Centers on Telegram Messenger Using Fulltext Search Boolean Mode*. *2*(2).

Ferdiyan, B. Z., Nugroho, E. S., Informatika, T., Informasi, T., & Riau, P. C. (2021). *Sistem Informasi Rekapitulasi Pemilukada Kota Pekanbaru menggunakan Input dari Telegram API*. *1*(10), 56–63.

Ghozi, W., P, E. R., & Setiono, O. (2022). *Jurnal Mantik Design of a Telegram Chatbot to Control Internet Connection on Computer Laboratory*. *6*(3).

Gunawan, S., & Anshor, A. H. (2023). *BULLETIN OF COMPUTER SCIENCE RESEARCH Sistem Monitoring dan Kontrol Taman Pintar Berbasis IoT (Internet of Things) dengan NodeMCU ESP8266*. *3*(4), 283–288. https://doi.org/10.47065/bulletincsr.v3i4.270

Halder, S., Afsari, K., Serdakowski, J., Devito, S., Ensafi, M., & Thabet, W. (2022). *Real-Time and Remote Construction Progress Monitoring with a Quadruped Robot Using Augmented Reality*.

Hardiwansyah, M., Elektronika, P., Surabaya, N., Provider, C., Push, L., & Messaging, M. (n.d.). *IMPLEMENTASI PUSH DATA SERVER PADA JARINGAN*. 1–5.

Harsono, H. (2022). *FAKTOR-FAKTOR YANG MEMPENGARUHI SISTEM INFORMASI BERBASIS KOMPUTER : SISTEM OPERASI , SERVER , DAN PROGRAMMER (LITERATURE REVIEW EXECUTIVE SUPPORT SISTEM FOR BUSINESS)*. *3*(2), 583–593.

Idhom, M., Alit, R., & Endah, H. (2018). *Implementation System Telegram Bot for Monitoring Linux Server*. *1*(Icst), 1089–1093.

Nilsson, M., & Dunér, D. (2020). *Scalability of push and pull based event notification*.

Normadhoni, R., Dewanti, S. P., Namaskara, W. C., Konseling, B., Pendidikan, F. I., & Semarang, U. N. (2021). *Journal of Education and Technology*. *1*(1), 12–21.

Ogundeyi, K. E. (2019). *WEBSOCKET IN REAL TIME APPLICATION*. *38*(4), 1010–1020.

Parlika, R., & Pratama, A. (2020). *The Online Test application uses Telegram Bots The Online Test application uses Telegram Bots Version 1 . 0*. 0–7. https://doi.org/10.1088/1742-6596/1569/2/022042

Prabakaran, S., Ramar, R., Hussain, I., Kavin, B. P., Alshamrani, S. S., Alghamdi, A. S., & Alshehri, A. (2022). *Predicting Attack Pattern via Machine Learning by Exploiting Stateful Firewall as Virtual Network Function in an SDN Network*.

Rahmatulloh, A., & Firmansah, T. A. (2019). *Telegram Bot for Automation of Academic Information Services with The Forward Chaining Method*. *3*(2), 73–78.

Saad, S., Firdaus, A., Fadzil, A., & Kushan, A. L. (2021). *REMOTE INTRUSION RESPONSE SYSTEM USING*. 220–234.

Shodiq, W., & Fauzi, F. (2021). *Pengembangan Aplikasi Telegram Pada Kegiatan Pramuka di Masa Pandemi Covid-19*. *1*(2).

Sholihah, W., Pripambudi, S., & Mardiyono, A. (2020). *Log Event Management Server Menggunakan Elastic Search Logstash Kibana (ELK Stack)*. *2*(1), 12–20.

Sucipto, S., Resti, N. C., Andriyanto, T., Karaman, J., & Qamaria, R. S. (2019). *Transactional database design information system web-based tracer study integrated telegram bot Transactional database design information system web-based tracer study integrated telegram bot*. https://doi.org/10.1088/1742-6596/1381/1/012008

Sulaeman, Y. J. (2022). *Implementasi Network Monitoring dan Notifikasi Sistem di PT XYZ Menggunakan Zabbix*. *4*(1), 1–7.

Syani, M., & Saputro, B. (2021). *Pada Virtual Private Server Berbasis Telegram Bot API (Studi Kasus Politeknik TEDC Bandung)*.