

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.894>

Pengembangan Sistem *Monitoring Filesystem Server Database* Dengan *Alert* Menggunakan *Bot Telegram*

Azhary Wali¹, Agus Sulistyanto², Tomi Defisa³

Program Studi Teknik Informatika¹, Teknik Informatika²,
Sekolah Tinggi Manajemen Informatika dan Komputer Jayakarta
Jalan Salemba I no.10 RT.4/RW.6 Kenari Kec. Senen Kota Jakarta Pusat 10430

Email: 20577004@stmik.jayakarta.ac.id , agus.sulistyanto@gmail.com

Abstrak

Database adalah aspek yang sangat penting dari sebuah sistem, terutama di sebuah perusahaan yang aktifitas-aktifitasnya bergantung pada sistem tersebut. Oleh karena itu, ketersediaan database sangatlah penting. Salah satu hal yang penting dalam menjaga sebuah sistem dengan database adalah file backup database. Backup file ini disimpan di dalam sebuah filesystem, yang mana hal ini membuat filesystem perlu untuk dimonitoring. Pada sebuah perusahaan yang memiliki banyak server, sangatlah sulit untuk memonitor satu persatu server-server tersebut, sehingga diperlukan fleksibilitas. Telegram merupakan sistem berkirim pesan antar platform yang gratis dan memiliki API BOT. Di era dimana semua orang menggunakan telepon genggam, memonitor sebuah server secara remote menjadi mungkin. Yaitu dengan mengintegrasikan database server dengan telegram untuk mengirimkan informasi server ke perangkat telepon genggam dari administrator menggunakan BOT telegram.

Kata kunci: database; filesystem; telegram;

Abstract: Database is an essential aspect of a system, especially in a company that its activities dependent on the system, therefore the availability of database is important. One of the most important things from maintaining a system with database is the backup file of database. Backup file is stored on a filesystem, which make a monitoring of filesystem is necessary. In a company that have so many database servers, it is difficult to monitor all the servers one by one, so flexibility is a must. Telegram is one of the cross-platform messaging system that is free and has BOT API. In the era where everyone is using a mobile phone, monitoring a server remotely is made possible. That is by integrated the database server with telegram to send information of the sever to mobile devices of administrator using telegram BOT.

Keywords: database; filesystem; telegram;

I. PENDAHULUAN

Penggunaan sistem informasi sangat dibutuhkan dalam meningkatkan produktivitas dalam suatu perusahaan. Dengan semakin banyaknya persaingan, membuat perusahaan berpikir keras dalam menjalankan proses bisnisnya. Untuk itu digunakanlah suatu sistem *database* dalam pengelolaan data. Penggunaan database dalam perusahaan merupakan salah satu hal yang sangat wajib untuk mendukung kegiatan-kegiatan yang ada di dalam perusahaan. *Database* merupakan suatu kumpulan data yang saling terhubung dan dapat diakses dengan mudah. Dengan adanya *Database Management System (DBMS)*, perusahaan dapat dengan mudah mengakses dan menyimpan data informasi [1]

Database menyimpan data penting dari perusahaan, sehingga penting untuk database agar selalu dapat diakses kapan saja. Semakin besar transaksi dari suatu perusahaan semakin besar juga data yang disimpan di dalam *database* nya. Oleh karena itu monitoring *database* merupakan hal yang sangat dibutuhkan dan harus dilakukan untuk dapat memastikan ketersediaan *database*. Karena pentingnya *database* ini, semua masalah yang mungkin mengganggu berjalannya *database* sebagaimana mestinya harus dapat dengan cepat untuk ditangani.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.894>

Dengan ini pengembangan sistem monitoring *database* merupakan hal yang perlu untuk memberikan peringatan dini sebelum suatu masalah terjadi di *database*.

Salah satu bagian penting yang perlu diperhatikan dalam menjaga *database* adalah ketersediaan *backup database*. Setiap *database* dibackup dengan tujuan untuk menyediakan cadangan data-data dari resiko kerusakan, data yang hilang atau tidak dapat diakses kembali. *Backup* dianggap sangat kritikal karena menyangkut keamanan data. Untuk memastikan proses *backup* berhasil dilakukan, maka *admin database* harus memastikan ketersediaan *filesystem* dari *server database* dalam kondisi cukup. Masalah yang sering muncul pada perusahaan atau institusi yang mempunyai *server database* adalah kurangnya fleksibilitas dalam proses supervisi yang mana hal ini terjadi karna *human error*, terlebih untuk proses supervisi yang dilakukan 24 jam.

Berkembangnya teknologi saat ini, monitoring *database* dapat dilakukan dengan maksimal. Sistem alert dapat digunakan untuk dapat memberikan peringatan ke pada pengurus *database* dengan cepat. Salah satu sistem alert yang bisa diberdayakan adalah melalui *Telegram*. *Telegram* merupakan aplikasi *cloud based* dan sistem enkripsi yang menyediakan enkripsi *end-to-end*, *self destruction messages*, dan infrastruktur *multi- data center*. *Cloud based* pada *telegram* memungkinkan proses pengiriman jauh lebih cepat serta media penyimpanan yang besar [2]. Dengan pemanfaatan teknologi bot *telegram* maka alert dapat dikirimkan ke perangkat *mobile* dari pengurus *database* sehingga jika terdapat kemungkinan terjadi problem di *database* dapat segera disolusikan.

II. METODE DAN MATERI

2.1. Metode

Pada penelitian ini metode yang dipakai adalah *research & development*. Desain dari sistem ini dijelaskan dari tahap desain arsitektur, *server database* yang di-*monitoring* serta pemanfaatan aplikasi *Telegram* untuk mengirim *alert*.

2.2. Database

Menurut Ramez [3] sebuah database memiliki arti tersirat sebagai berikut :

- ✓ Sebuah database mewakili beberapa aspek dari dunia nyata, kadang-kadang disebut dunia kecil. Perubahan-perubahan pada dunia kecil tercermin pada database.
- ✓ Sebuah database adalah koleksi data yang koheren secara logika dengan makna bawaan. Sebuah kumpulan data yang bermacam-macam secara acak tidak tepat untuk disebut database.
- ✓ Sebuah database didesain, dibangun dan diisi dengan data untuk tujuan yang spesifik. Database memiliki kumpulan pengguna yang disasar dan beberapa aplikasi yang terbentuk sebelumnya yang mana diminati para pengguna ini.

2.3. Telegram Bot

Aplikasi *Telegram* dipilih karena aplikasi ini tidak berbayar, ringan dan *multiplatform*. *Telegram* juga memiliki sebuah *Bot API* yang cukup lengkap dan berkembang, sehingga memungkinkan untuk dijadikan bot pintar yang dapat merespon pesan-pesan dari *public*.

Ada dua cara untuk membuat sebuah *Telegram Bot*, dengan menggunakan metode *long polling* dan *webhook*. Dengan metode *long-polling*, *server* secara berkala kepada bot untuk melihat apakah ada pesan masuk. Jika ada pesan masuk, *server* akan mengeksekusi berdasarkan permintaan dari pesan masuk yang berasal dari pengguna. Jika tidak ada pesan masuk, kondisi *server* ada pada posisi *idle*. Pada metode *long-polling* Bot akan berjalan secara langsung melalui *server*, semetara pada metode *webhook bot* akan di-*hosting* dan harus menggunakan <https> [4] .

2.4. Linux



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

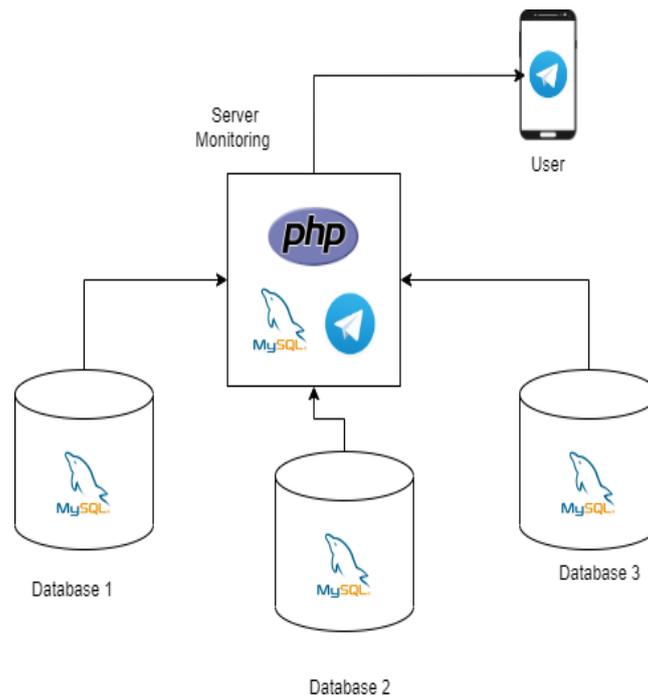
DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.894>

Linux adalah sebuah sistem operasi *open source* dan *load* (gratis) dibawah lisesnsi *GPL (General Public License)*. *Open source* berarti sumber kode telah termasuk di dalam *linux* sehingga dapat dilihat siapapun tanpa harus menandatangani perjanjian seperti *NDA (Non-Disclosure Agreement)* [5].

III. PEMBAHASAN DAN HASIL

3.1 Arsitektur Sistem

Penelitian ini berfungsi untuk merancang sistem *alert bot telegram* yang akan menggunakan data *server database staging/development*, dimana akan terdapat beberapa *sampel server database* yang akan di-*monitoring*. Nantinya *filesystem* yang merupakan direktori dari *backup database* akan di-*monitoring* dan jika *usage* sudah melebihi batas maka sistem akan mengirimkan *alert* melalui *telegram* ke user.



Gambar 3.1 Arsitektur sistem

Sistem *alert telegram* pada penelitian ini menggunakan 4 *server sampel* yang menggunakan sistem operasi *centos 7*. Pembagian utilisasi dari setiap *server* tersebut antara lain 1 *server* merupakan *server monitoring* dan 3 diantaranya adalah *server database*. Setiap *server* nantinya akan bisa terhubung melalui port 1818 untuk pengiriman *file monitoring*. Gambaran arsitektur dari *sistem monitoring* dan *alerting bot telegram* pada penelitian ini dapat dilihat pada gambar 3.1.

Pada penelitian ini setiap *server* akan memiliki *database MySQL/mariaDB* di dalamnya. Untuk ketiga *server database MySQL/mariaDB* akan digunakan sebagai simulasi *backup* dan *monitoring filesystem* pada direktori *file backup* tersebut. Untuk *server monitoring* akan diinstal juga *database MySQL* yang akan digunakan untuk menampung informasi *filesystem* yang ada pada ketiga *database* tersebut. Nantinya setiap harinya ketiga *server database* akan mengirimkan sebuah *file .txt* ke *server monitoring* dan *file* tersebut akan di *insert* ke dalam *database* pada *server monitoring* tersebut.

3.2 Flowchart Sistem

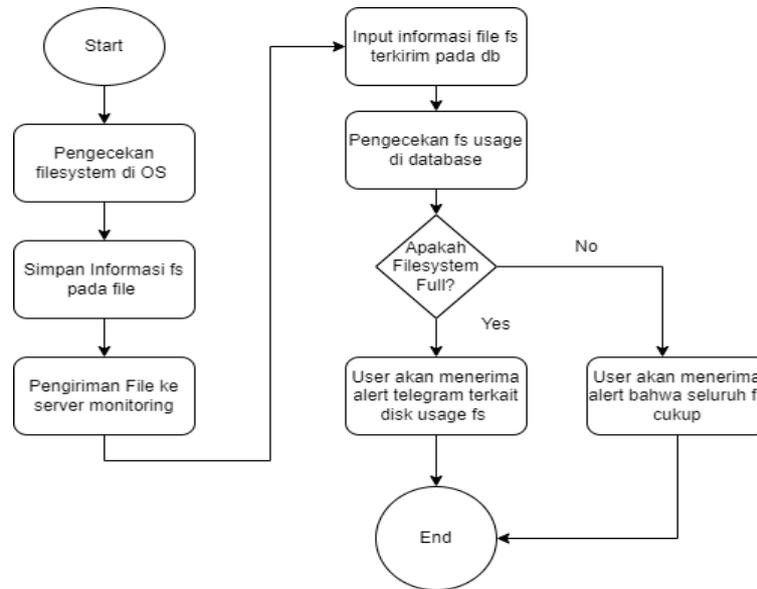


This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.894>

Terdapat dua kondisi alert yang akan dikirimkan ke perangkat user. Kondisi pertama adalah jika *filesystem usage* di *server* tidak melebihi batas, maka *alert* yang akan dikirimkan berisi informasi bahwa *filesystem* cukup. Sedangkan jika *usage filesystem* melebihi batas yang telah ditentukan maka *alert* yang dikirimkan adalah informasi bahwa *filesystem* tidak cukup seperti *flowchart* yang terlihat pada gambar 3.2



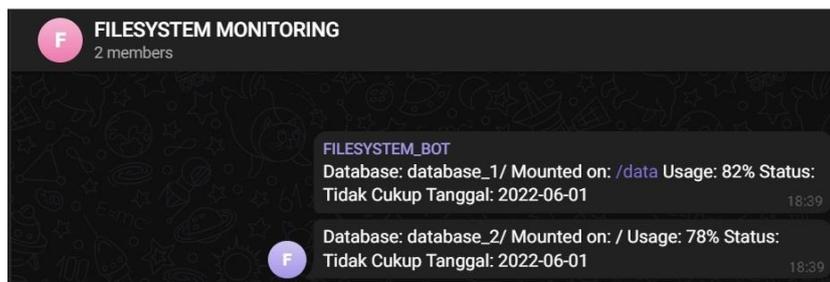
Gambar 3.2 Flowchart Sistem

3.3 Hasil

Dalam melakukan *monitoring* dan *alerting*, sistem yang dibuat akan melakukan pengecekan melalui *database*. Maka dari itu sebelum menguji sistem, harus dipastikan bahwa seluruh data informasi *filesystem* harus sudah berhasil di-*import*. Setelah *database* sudah tervalidasi maka selanjutnya akan dilakukan pengujian pada sistem *monitoring* dan *alerting* untuk menguji apakah *alert* yang dikirim sudah sesuai dengan isi dari *database*. Terdapat 2 skenario pengujian hasil:

1. Skenario Filesystem Tidak Cukup

Untuk skenario pertama ini *filesystem server database* dalam kondisi tidak cukup. *Filesystem* yang tidak cukup bisa berasal dari satu atau lebih *database*. Akan dilakukan *monitoring* apakah *bot* dapat mengirimkan pesan *alert* yang mengatakan bahwa terdapat *filesystem* yang tidak cukup. Ini terlihat pada gambar 3.3.



Gambar 3.3 Alert Telegram Filesystem Tidak Cukup

2. Skenario Filesystem Cukup

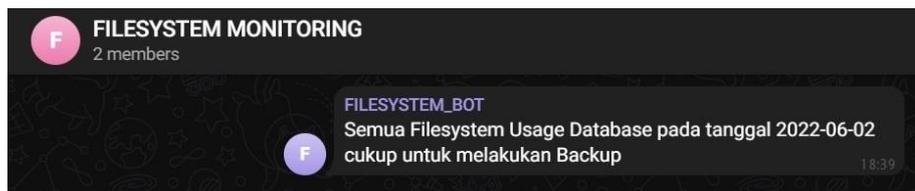


This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>

DOI: <https://doi.org/10.52362/jmijayakarta.v2i3.894>

Skenario kedua yang akan dimonitoring adalah bagaimana sistem *monitoring* dan *alerting* akan mengirimkan pesan ke *telegram* ketika seluruh *filesystem* pada ketiga *server database* cukup untuk melakukan *backup*.



Gambar 3.4 *Alert Telegram Filesystem Cukup*

IV. KESIMPULAN

Berdasarkan hasil dan analisis yang telah dilakukan maka dapat disimpulkan bahwa *alert bot telegram* yang dihasilkan mampu memangkas waktu pengecekan *filesystem* secara manual. Hal ini tentunya dapat digunakan sebagai langkah *preventive* untuk mencegah jika terjadi kegagalan *backup* ketika backup *filesystem* tidak mencukupi.

REFERENSI

- [1] M. Karsen. [Online]. Available: <https://sis.binus.ac.id/2017/09/27/manfaat-database-dalam-perusahaan/>.
- [2] J. Fahana, R. Umar and F. Ridho, "Pemanfaatan Telegram Sebagai Notifikasi Serangan," *Jurnal Sistem Informasi*, 2017.
- [3] R. Elmasri, *Fundamentals of database systems seventh edition*, 2021.
- [4] M. B. Chaniago and A. Junaidi, "Student Presence Using RFID and" Telegram" Messenger Application: A Study in SMK Unggulan Terpadu Pgi Bandung, Indonesia," *International Journal of Higher Education*, pp. 94-102, 2019.
- [5] S. Haris, *ADMINISTRASI SERVER BERBASIS LINUX# 1*, 2021.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

<http://journal.stmikjayakarta.ac.id/index.php/JMIJayakarta>