

## Perbandingan pencadangan data menggunakan RSync dan SFTP

**Asep Nurhuda<sup>1</sup>, Annafi Franz<sup>2</sup>**

<sup>1,2</sup>Program Studi Teknologi Rekayasa Perangkat Lunak, Jurusan Rekayasa dan Komputer,

Politeknik Pertanian Negeri Samarinda

<sup>1</sup>acep.noor@gmail.com, <sup>2</sup>annafifranz1@gmail.com

### ABSTRAK

Pencadangan data pada server merupakan langkah kritis dalam pengelolaan sistem informasi dan teknologi. Dalam era teknologi informasi yang terus berkembang pesat, data menjadi aset berharga bagi organisasi dan perusahaan. Menghadapi berbagai ancaman seperti kegagalan perangkat keras, serangan siber, atau peristiwa tak terduga, pencadangan data menjadi prioritas utama. Manfaatnya meliputi perlindungan terhadap potensi kehilangan informasi, dukungan terhadap kelancaran operasional, dan menjaga integritas informasi. Penelitian ini fokus pada penerapan pencadangan data menggunakan rsync dan sftp, dengan tujuan membandingkan proses yang lebih cepat dan efisien. Hasil penelitian ini memberikan wawasan terkait teknik pencadangan data yang optimal menggunakan rsync dan sftp. Organisasi dan perusahaan dapat memilih metode yang sesuai dengan kebutuhan untuk memastikan kehandalan, kecepatan, dan efisiensi dalam melindungi serta memulihkan data yang krusial.

**Kata kunci:** *backup, rsync, sftp, siber.*

### ABSTRACT

*Backing up data on servers is a critical step in managing information and technology systems. In the era of information technology that continues to develop rapidly, data has become a valuable asset for organization and companies. Facing various threats such as hardware failure, cyberattacks, or unexpected events, data backup becomes a top priority. Benefits include protection against potential loss of information, support for smooth operations, and maintaining information integrity. This research focuses on implementing data backup using rsync and sftp, with the aim of comparing faster and more efficient processes. The results of this research provide insight into optimal data backup techniques using rsync and sftp. Organizations and companies can choose the method that suits their needs to ensure reliability, speed and efficiency in protecting and restoring crucial data.*

**Keywords:** *backup, rsync, sftp, cyber.*

### 1. PENDAHULUAN

Pencadangan data pada server menjadi langkah kritis yang tidak dapat diabaikan dalam konteks pengelolaan sistem informasi dan teknologi. Seiring dengan kemajuan pesat dalam teknologi informasi, data kini menjadi aset yang sangat berharga bagi berbagai entitas seperti organisasi dan perusahaan. Dalam menghadapi berbagai ancaman, seperti kegagalan perangkat keras, serangan siber, atau peristiwa tak terduga lainnya, kebutuhan untuk melaksanakan pencadangan data semakin menjadi prioritas utama.

Manfaat dari pencadangan data pada server bersifat luas dan mendalam. Pencadangan data berperan sebagai perlindungan terhadap potensi kehilangan informasi akibat berbagai kejadian yang tidak diinginkan. Selain itu, praktik pencadangan data juga mendukung kelancaran operasional sehari-hari, memungkinkan organisasi dan perusahaan untuk kembali ke kondisi normal dengan segera setelah terjadi insiden. Pencadangan data juga memiliki peran kunci dalam

menjaga integritas informasi, dengan memiliki salinan data yang dapat diandalkan, organisasi dapat mengurangi risiko kehilangan data yang sangat berharga, sekaligus mencegah dampak negatif terhadap reputasi organisasi.

Penelitian berkaitan dengan *backup* dan *restore* data telah banyak dilakukan seperti yang dilakukan oleh Adhiwibowo, berdasarkan hasil penelitian yang telah dilakukan, dapat diambil kesimpulan bahwa metode *remote connection* untuk melakukan *backup & restore* dapat dilakukan menggunakan Rsync pada sistem operasi linux [1]. Sedangkan menurut penelitian Jackie, dengan menggunakan aplikasi Veeam Backup & Replication dapat melakukan *backup* data secara berkala dan berkelanjutan sesuai konfigurasi yang telah diatur sehingga dapat meringankan beban kerja admin server [3]. Dalam penelitian Huang, pemanfaatan GlusterFS dengan skema mirroring atau replicate dapat menjamin data never loss [2]. Sedangkan pada penelitian ini dilakukan penerapan pencadangan data menggunakan rsync dan sftp, untuk membandingkan proses yang lebih cepat dan efisien.

## 2. METODOLOGI PENELITIAN

### 2.1 Metode Pencadangan Data

Metode pencadangan data yang digunakan dalam penelitian ini yaitu *on-site* dan *off-site backup*.

#### 2.1.1 On-site Backup

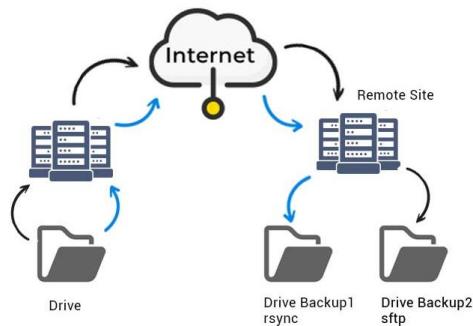
Singkatnya *on-site backup* adalah metode backup data dimana salinan data disimpan pada perangkat keras seperti HDD/SSD. Keuntungan dari metode ini salah satunya ialah tidak membutuhkan banyak biaya. Selain itu, *on-site backup* tidak memerlukan internet untuk melakukan mengakses kembali data yang telah di-*backup*. Yang artinya, data dapat diakses kapan saja secara *offline*.

#### 2.1.2 Off-site Backup

*Off-site backup* juga dikenal dengan *cloud backup*. Data yang telah disalin disimpan dalam cloud storage. *Cloud storage* adalah penyimpanan virtual yang diakses dengan jaringan internet. Hal ini memudahkan pengguna untuk mengakses kembali data mereka kapan saja dan di mana saja. [6]

### 2.2 Rancangan Pencadangan Data

Dalam satu *local site* (server) yang akan dicadangkan datanya dipilih satu direktori, kemudian direktori dibuat duplikasi dengan mengirim ke *remote site* menggunakan rsync dan sftp. Remote site telah disiapkan dua direktori penyimpanan untuk pencadangan dari rsync dan sftp, kecepatan *bandwidth* untuk *upload* dan ukuran *file* yang akan dicadangkan adalah sama.

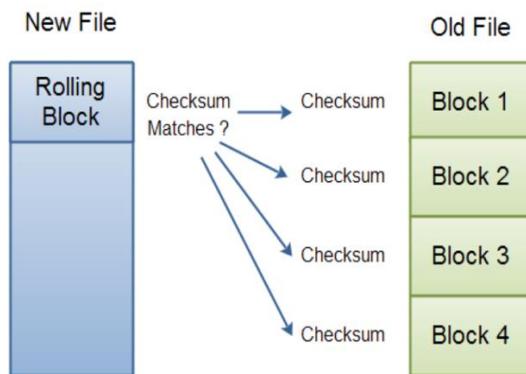


Gambar 1. Pencadangan data

*Drive backup1* akan menampung data pencadangan dari rsync dan *drive backup2* akan menampung data pencadangan dari sftp.

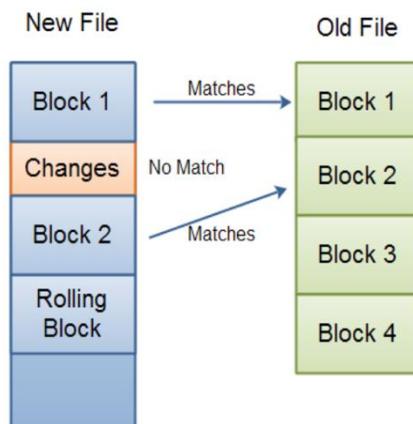
### 2.3 RSync

RSync adalah protokol sinkronisasi *file* (atau data) jarak jauh yang memungkinkan untuk menyinkronkan *file* yang disimpan di komputer lokal dengan *file* yang disimpan di komputer jarak jauh (*remote computer*) sehingga setelah sinkronisasi, *file* lokal dan jarak jauh menjadi identik. Jika ada perbedaan antara *file* lokal dan jarak jauh, RSync mendeteksi perbedaan tersebut dan hanya menukar perbedaannya (+ instruksi penggabungan) antara komputer lokal dan jarak jauh, sehingga kedua *file* dapat dibuat identik [4].

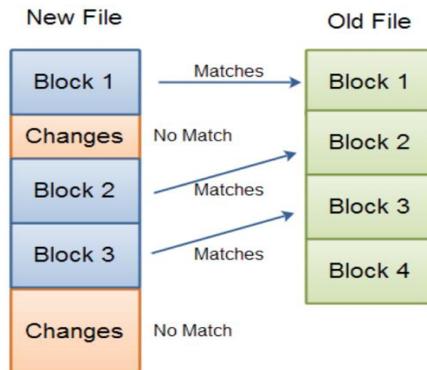


Gambar 2. Pengecekan Checksum pada File Lama yang dicocokan dengan File Baru.

Komputer yang menyimpan *file* terlama (*Old File*) membagi versi *file* terlama menjadi blok-blok, katakanlah 1024 atau 2048 byte. *File* tidak dibagi pada *disk* tapi hanya sesuatu yang dilakukan secara logis, secara internal di dalam memori. Untuk setiap *blok old file* dihitung *checksum*-nya, daftar *checksum* blok dikirim ke komputer yang menyimpan *file* terbaru (*New File*).



Gambar 3. Checksum File Lama ada yang tidak cocok dengan File Baru

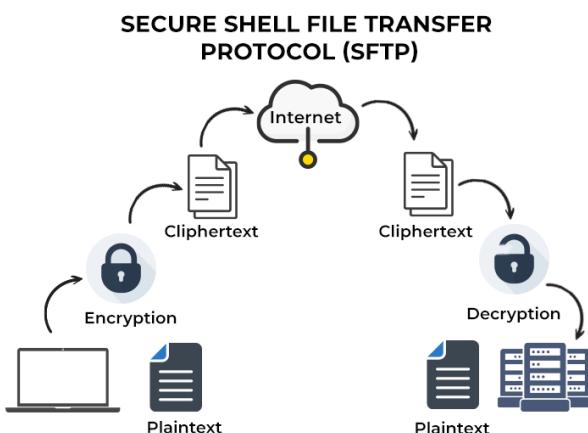


Gambar 4. Pengecekan *checksum* selesai

Pada gambar3 pengecekan *checksum* (*old file*) terus dilakukan dengan *new file* hingga ditemukan blok yang tidak cocok. Pada gambar 4 pengecekan *checksum* selesai dan ditemukan dua *block file* baru yang tidak cocok dengan *file* lama. Komputer yang menyimpan *file* baru mengirimkan kembali instruksi ke komputer yang menyimpan *file* lama tentang cara membuat salinan *file* versi terbaru. Hal ini dilakukan dengan mengirimkan daftar referensi blok di *file* lama untuk bagian *file* terbaru yang tidak berubah. Sedangkan untuk bagian yang diubah juga dikirimkan kembali data yang diubah tersebut secara lengkap. Komputer yang menyimpan *file* lama menerima daftar referensi blok dan data literal (perubahan) serta membuat versi *file* baru.

## 2.4 SFTP

SFTP (SSH File Transfer Protocol) adalah protokol *transfer file* yang aman. Ini berjalan melalui protokol SSH. Ini mendukung fungsi keamanan dan otentifikasi penuh SSH.



Gambar 5. Cara kerja SFTP

## 3. HASIL DAN PEMBAHASAN

### 3.1 Onsite Backup

Pencadangan data di *onsite backup* hanya dapat dilakukan oleh rsync karena dalam dokumentasinya rsync memiliki fitur untuk melakukan ini.

### 3.1.1 Mengecek versi rsync

Sebelum menjalankan pencadangan data, perlu dipastikan bahwa rsync telah ter-*install* dan dapat dilihat versi rsync dengan perintah `rsync --version`

```
root@172.17.0.1:~# docker# rsync --version
rsync version 3.1.2 protocol version 31
Copyright (C) 1996-2015 by Andrew Tridgell, Wayne Davison, and others.
Web site: http://rsync.samba.org/
Capabilities:
  64-bit files, 64-bit inums, 64-bit timestamps, 64-bit long ints,
  socketpairs, hardlinks, symlinks, IPv6, batchfiles, inplace,
  append, ACLs, xattrs, iconv, symtimes, prealloc

rsync comes with ABSOLUTELY NO WARRANTY. This is free software, and you
are welcome to redistribute it under certain conditions. See the GNU
General Public Licence for details.
```

Gambar 6. Melihat versi Rsync

Berdasarkan gambar 6 dapat dilihat bahwa rsync telah terinstall dengan versi 3.1.2, dari situs resmi rsync <https://rsync.samba.org/> hingga bulan April 2023 telah dirilis rsync 3.3.0 namun versi yang telah terinstall sudah cukup untuk melakukan pencadangan data.

### 3.1.2 Pencadangan ke Direktori *Onsite Backup*

Pencadangan data di *onsite backup* dilakukan pada direktori `/home/backup` sedangkan direktori yang akan dicadangkan bernama `pkmseimeriam` yang berisi file sebuah website.

Perintah untuk melakukan pencadangan data menggunakan rsync:

```
rsync -avzHP --dry-run pkmseimeriam/ /home/backup
```

Lebih jelasnya dapat dilihat pada gambar 7.

```
100-default.conf Backup30-09-23.tar.gz dbdata db_default.env docker-compose.yaml Dockerfile pkmseimeriam
root@172.17.0.1:~# docker# mkdir /home/backup
root@172.17.0.1:~# docker#
root@172.17.0.1:~# docker# rsync -avzHP --dry-run pkmseimeriam/ /home/backup/
rsync: sending incremental file list
/
DS_Store
editorconfig
env
env.example
gitattributes
gitignore
UPNP.adb
```

Gambar 7. Perintah untuk pencadangan data

Dengan menggunakan option `-P` maka *progress bar* akan terlihat, proses akan selesai ditandai dengan menampilkan *total size* dan *speedup* rsync ketika mengeksekusi *command* seperti terlihat pada gambar 8 dibawah ini.

```
sent 562,386 bytes received 60,728 bytes 1,246,228.00 bytes/sec
total size is 417,072,921 speedup is 669.34 (DRY RUN)
root@pkmseimeriam-docker#
```

Gambar 8. Proses rsync selesai

### 3.2 Off Site Backup Menggunakan SFTP

*Off Site Backup* berarti melakukan pencadangan ke perangkat lain, istilah lainnya yaitu remote site.

#### 3.2.1 Memastikan sftp berjalan

Perlu dipastikan bahwa koneksi SFTP ke *remote site* berjalan dengan baik, dapat dilihat pada gambar 9..

```
root@pkmseimeriam-docker: ~ # sftp superadmin@103.189.235.189
superadmin@103.189.235.189's password:
Connected to 103.189.235.189.
sftp> pwd
Remote working directory: /home/superadmin
sftp> ls
BackupFromRsync  BackupFromSFTP
sftp> cd BackupFromSFTP
sftp> ls
sftp> bye
```

Gambar 9. Koneksi SFTP ke Remote Site

Perintah **sftp superadmin@103.189.235.189** menunjukkan bahwa user root yang saat ini digunakan untuk mengakses komputer (server) yang akan dicadangkan datanya melakukan koneksi ke komputer dengan alamat publik 103.189.235.189 dengan user superadmin. Status berhasil mengakses superadmin ditandai dengan informasi *Connected to 103.189.235.189*.

#### 3.2.2 Transfer file ke remote site

SFTP dapat melakukan transfer ke *remote site* secara *massal*, dengan menambahkan opsi **-r** maka secara rekursif perintah akan diulang di direktori didalam direktori sehingga isi didalamnya dapat ditransfer seluruhnya.

```
sftp> put -r pkmseimeriam /home/superadmin/BackupFromSFTP
Uploading pkmseimeriam/ to /home/superadmin/BackupFromSFTP/pkmseimeriam
Entering pkmseimeriam/
pkmseimeriam/.env.example                                         100% 1069    5.3KB/s  00:00
Entering pkmseimeriam/tests                                         100%  163    0.9KB/s  00:00
Entering pkmseimeriam/tests/Unit                                    100%  243    0.4KB/s  00:00
pkmseimeriam/tests/Unit/ExampleTest.php                           100%  359    2.3KB/s  00:00
Entering pkmseimeriam/tests/Feature                                100%  375    2.2KB/s  00:00
pkmseimeriam/tests/Feature/ExampleTest.php
pkmseimeriam/tests/CreateApplication.php
Entering pkmseimeriam/bootstrap
```

Gambar 10. Transfer menggunakan sftp

Proses *transfer* seperti terlihat pada gambar 10 akan memakan waktu yang cukup lama, cara paling efisien yaitu dengan melakukan kompresi pada direktori **/pkmseimeriam**, kompresi menggunakan **tar** akan lebih baik sehingga akan menghasilkan compress file berekstensi **.tar.gz**

```
root@politani:/home/politani/pkmseimeriam-docker# ls
000-default.conf dbdata db_default.env docker-compose.yaml Dockerfile pkmseimeriam
root@politani:/home/politani/pkmseimeriam-docker# tar -cvzf Backup`date +"%d-%m-%y"`.tar.gz pkmseimeriam/
pkmseimeriam/
pkmseimeriam/.env.example
pkmseimeriam/tests/
pkmseimeriam/tests/TestCase.php
pkmseimeriam/tests/Unit/
pkmseimeriam/tests/Unit/ExampleTest.php
```

Gambar 11. Compress using tar

Pada gambar 11 dapat dilihat bagaimana cara melakukan *compressing file* menggunakan tar dan mengkobinasikan dengan *command* linux yaitu **date** untuk mendapatkan tanggal saat ini.

```
root@politani:/home/politani/pkmseimeriam-docker# ls
000-default.conf Backup30-09-23.tar.gz dbdata db_default.env docker-compose.yaml Dockerfile pkmseimeriam
root@politani:/home/politani/pkmseimeriam-docker#
```

Gambar 12. Hasil Compress

*File* berekstensi tar.gz dikirim menggunakan SFTP dengan lebih cepat seperti terlihat pada gambar 13.

```
root@politani:/home/politani/pkmseimeriam-docker# sftp superadmin@103.189.235.189
superadmin@103.189.235.189's password:
Connected to 103.189.235.189.
sftp> pwd
Remote working directory: /home/superadmin
sftp> ls
BackupFromSync  BackupFromSFTP
sftp> cd BackupFromSFTP
sftp> pwd
Remote working directory: /home/superadmin/BackupFromSFTP
sftp>
sftp> ll
000-default.conf Backup30-09-23.tar.gz dbdata db_default.env docker-compose.yaml Dockerfile pkmseimeriam
sftp> put Backup30-09-23.tar.gz
Uploading Backup30-09-23.tar.gz to /home/superadmin/BackupFromSFTP/Backup30-09-23.tar.gz
Backup30-09-23.tar.gz                                100% 175MB   6.5MB/s  00:26
sftp>
```

Gambar 13. Transfer tar.gz file ke Remote Site

Dengan memperhatikan informasi pada gambar 13 direktori pkmseimeriam akan dibaca satu persatu *file*-nya dan di *upload*. Proses membaca satu persatu *file* ini membutuhkan waktu yang cukup lama karena isi direktori sebesar 183 MB. Catatan waktu yang dibutuhkan untuk menyelesaikan upload file pada direktori pkmseimeriam dituliskan pada tabel 1.

Tabel 1. Kecepatan Upload SFTP

File Sebelum Kompresi	File Setelah Kompresi (tar.gz)
Waktu	> 60 Menit
Size	183MB

### 3.3 Offsite Backup Menggunakan Rsync

*File* yang sama yaitu Backup30-09-23.tar.gz dikirim ke remote site menggunakan rsync seperti terlihat pada gambar 13.



Gambar 14. Transfer *file* menggunakan rsync

### 3.4 Hasil Perbandingan

Dengan memperhatikan secara seksama gambar 13 dan gambar 14, *file Backup 30-09-23.tar.gz* dikirim ke *remote site* yang sama dengan *bandwidth* internet yang sama, kecepatan *transfer* yang dibutuhkan oleh rsync yaitu 00:12 sedangkan yang dibutuhkan oleh SFTP 00:26.

Beberapa perbandingan lainnya yang perlu diperhatikan untuk proses transfer yaitu dengan mempertimbangkan aspek seperti pada tabel 2.

Tabel 2. Perbandingan Rsync dan SFTP

	Rsync	SFTP
Local File Transfer	Yes	No
Remote File Transfer	Yes	Yes
Incremental File Transfer	Yes	No
File Compression	Yes	Yes
Data Encryption	Yes	Yes
Time Execution	Faster	
Massal File Transfer		Yes

## 4. KESIMPULAN

Berdasarkan hasil yang didapatkan pada penerapan kedua cara pencadangan yaitu rsync dan sftp, maka diperoleh hasil yaitu keduanya dapat melakukan pencadangan ke *remote site*, namun rsync memiliki kelebihan lain yaitu dapat melakukan pencadangan pada *onsite*. Untuk kehandalan pencadangan data, kecepatan, dan efisiensi maka rsync lebih baik karena fiturnya yang mampu hanya menyimpan data baru atau "titik data delta" yang telah diubah sejak pencadangan sebelumnya dijalankan, bukan pencadangan penuh. Hal ini berpengaruh pada kecepatan proses pencadangan data, ukuran tempat penyimpanan, dan pemulihan data. Sedangkan sftp hanya melakukan pencadangan data secara penuh ke *remote site*, yang berarti ukuran penyimpanan yang dibutuhkan di *remote site* harus lebih besar karena duplikasi data tidak mungkin dapat diatasi dengan cara ini.

## DAFTAR PUSTAKA

- [1] Adhiwibowo, W., Suprayogi, M. S., & Nugroho, A. (2019). Pengamanan Data Pada Aplikasi Sijalu Universitas Semarang Dengan Metode Remote Backup & Restore. *Jurnal Pengembangan Rekayasa dan Teknologi*, 14(1), 24-27.
- [2] Huang, J., Fitrisia, Y., Ridha, F., & Arif, M. (2018). Implementasi Multi Server Data Storage Pada Cloud Computing. *Jurnal Komputer Terapan*, 4(2), 470947.
- [3] Jackie, J. (2022). Analisa dan Penerapan Pencadangan Pusat Data Antar Site dengan Teknologi VPN. *Journal of Information System and Technology (JOINT)*, 3(2), 257-269.
- [4] Jenkov, Jacob. (2022). RSsync - Remote Synchronization Protocol. <https://jenkov.com/tutorials/rsync/index.html>
- [5] Ssh.com, SSH File Transfer Protocol (SFTP): Get SFTP client & server. <https://www.ssh.com/academy/ssh/sftp-ssh-file-transfer-protocol>

- [6] Wowcrack.co.id, (2021). Mengenali backup data dan metodenya  
<https://blog.wowrack.co.id/2021/02/08/mengenali-backup-data-dan-metodenya/>