

**SINKRONISASI FILE TEXT SATU ARAH BERBASIS  
CLIENT-SERVER**

Skripsi



oleh  
**KRISTIAN ADRI HARTANTO**  
**22084531**

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
2013

# **SINKRONISASI FILE TEXT SATU ARAH BERBASIS CLIENT-SERVER**

Skripsi



Diajukan kepada Program Studi Teknik Informatika Fakultas Teknologi  
Informasi Universitas Kristen Duta Wacana  
Sebagai Salah Satu Syarat dalam Memperoleh Gelar  
Sarjana Komputer

Disusun oleh

**KRISTIAN ADRI HARTANTO**  
**22084531**

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
2013

## PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

### SINKRONISASI FILE TEXT SATU ARAH BERBASIS CLIENT-SERVER

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaan saya.

Yogyakarta, 27 Mei 2013



KRISTIAN ADRI HARTANTO  
22084531

## HALAMAN PERSETUJUAN

Judul Skripsi : SINKRONISASI FILE TEXT SATU ARAH  
BERBASIS CLIENT-SERVER  
Nama Mahasiswa : KRISTIAN ADRI HARTANTO  
N I M : 22084531  
Matakuliah : Skripsi (Tugas Akhir)  
Kode : TIW276  
Semester : Genap  
Tahun Akademik : 2012/2013

Telah diperiksa dan disetujui di  
Yogyakarta,  
Pada tanggal 27 Mei 2013

Dosen Pembimbing I



Antonius Rachmat C., SKom.,M.Cs

Dosen Pembimbing II



Willy Sudiarto Raharjo, SKom.,M.Cs

**HALAMAN PENGESAHAN**

**SINKRONISASI FILE TEXT SATU ARAH BERBASIS  
CLIENT-SERVER**

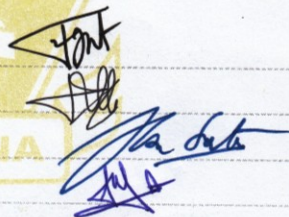
Oleh: KRISTIAN ADRI HARTANTO / 22084531

Dipertahankan di depan Dewan Penguji Skripsi  
Program Studi Teknik Informatika Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana - Yogyakarta  
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Komputer  
pada tanggal 27 Mei 2013

Yogyakarta, 27 Mei 2013  
Mengesahkan,

Dewan Penguji:

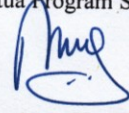
1. Antonius Rachmat C., SKom., M.Cs
2. Willy Sudiarto Raharjo, SKom., M.Cs
3. Budi Susanto, SKom., M.T.
4. Aditya Wikan Mahastama, S.Kom



Handwritten signatures of the examiners, including one that appears to be 'Antonius Rachmat C.' and another that appears to be 'Willy Sudiarto Raharjo'.



Dekan  
  
(Drs. Wimmie Handiwiidjojo, MIT.)

Ketua Program Studi  
  
(Nugroho Agus Haryono, M.Si)

## UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas berkat, rahmat, dan karunianya sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Sinkronisasi File Text Satu Arah Berbasis Client-Server”.

Penulisan laporan ini merupakan kelengkapan dan pemenuhan dari salah satu syarat dalam memperoleh gelar Sarjana Komputer. Selain itu, penulisan laporan Tugas Akhir ini juga bertujuan untuk melatih mahasiswa agar dapat menghasilkan suatu karya yang dapat dipertanggungjawabkan secara ilmiah, sehingga dapat bermanfaat bagi *user*nya.

Dalam menyelesaikan penelitian dan laporan Tugas Akhir ini, penulis telah banyak menerima bimbingan, saran, dan masukan dari berbagai pihak, baik secara langsung maupun secara tidak langsung. Untuk itu dengan segala kerendahan hati, pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada :

1. Bapak Antonius Rachmat, S.Kom., M.Cs. selaku dosen pembimbing I yang pertama yang selalu sabar dan menuntun penulis dalam membimbing dan mengerjakan penelitian serta penyusunan laporan Tugas Akhir.
2. Bapak Willy Sudiarto Raharjo, SKom.,M.Cs. selaku dosen pembimbing II yang selalu sabar membimbing dan memberikan contoh serta ide kepada penulis dalam mengerjakan penelitian dan penyusunan laporan Tugas Akhir.
3. Ayah, ibu serta adik, keluarga dr. Paulus dan keluarga Edi yang selalu memberikan doa dan dukungannya kepada penulis dalam menyelesaikan Tugas Akhir.
4. Rekan-rekan penulis yang dengan senang hati memberikan arahan, saran, dan, sharing dalam pengerjaan Tugas Akhir maupun penulisan laporan Tugas Akhir.

5. Pihak lain yang tidak dapat penulis sebutkan satu per satu, sehingga Tugas Akhir ini dapat terselesaikan dengan baik.

Penulis akan menerima dengan senang hati apabila terdapat kritik dan saran yang bermanfaat serta membangun dalam program dan laporan tugas akhir ini kedepannya. Penulis menyampaikan terima kasih atas perhatian dalam membaca laporan ini. Semoga apa yang penulis tuangkan dapat bermanfaat baik bagi semua orang yang berkepentingan.

Yogyakarta, Mei 2013

Penulis

©UKDW

## INTISARI

### Sinkronisasi File Text Satu Arah Berbasis Client-Server

Kemanan data dan informasi mulai berkembang dengan pesat. Begitu juga dengan data yang semakin mahal. Salah satu cara untuk mengamankan data yaitu dengan melakukan backup data atau sinkronisasi file. Sinkronisasi data dilakukan untuk mencegah kehilangan data dapat disebabkan oleh serangan pada data, kerusakan pada *harddisk*, kerusakan pada perangkat keras, *software corruption*, virus komputer, kesalahan *user*, maupun bencana alam. Namun untuk melakukan sinkronisasi file dengan ukuran yang besar, dibutuhkan waktu yang tidak sedikit, terlebih lagi jika dilakukan antar komputer dengan *bandwith* yang terbatas

Pada penelitian ini dilakukan perancangan sistem sinkronisasi dimana sistem tidak mengirim file secara utuh, namun hanya mengirim perbedaan file, dengan kata lain, sistem dapat memanfaatkan file yang lama sehingga pertukaran *byte* menjadi efektif. Sistem dengan model *client-server* mencari *byte* yang berbeda dengan checksums MD5 menemukan *byte* berbeda, mengirim dan menyusun ulang file untuk memperoleh hasil akhir file yang sama.

Dari hasil penelitian didapatkan bahwa sistem ini dapat mencapai performa terbaik dalam hal waktu eksekusi program dengan perbandingan ukuran file dengan blok pembagi adalah 1.000:1. Berdasarkan pengujian sistem dengan melakukan perbandingan *weak signature* CRC32 dengan *strong signature* MD5, CRC32 unggul dalam kecepatan melakukan sinkronisasi. Berdasarkan pengujian sistem terhadap *speedup* atau rasio perbandingan ukuran total file dengan data literal, semakin besar nilai *speedup*, maka program akan semakin efektif dalam pengiriman perbedaan file.

Kata Kunci : Algoritma, Sinkronisasi, Rsync, *Client-Server*, Rolling checksums.



## DAFTAR ISI

PERNYATAAN KEASLIAN SKRIPSI .....	iii
HALAMAN PERSETUJUAN .....	iv
HALAMAN PENGESAHAN .....	v
UCAPAN TERIMA KASIH .....	vi
INTISARI .....	viii
DAFTAR ISI .....	ix
DAFTAR GAMBAR .....	xii
DAFTAR TABEL .....	xiii
DAFTAR GRAFIK .....	xiv
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan Penelitian .....	2
1.5 Metodologi Penelitian .....	3
1.6 Sistematika Penulisan .....	3
BAB 2 TINJAUAN PUSTAKA DAN LANDASAN TEORI .....	5
2.1 Tinjauan Pustaka .....	5
2.2 Landasan Teori .....	7
2.2.1 Model <i>Client-Server</i> .....	7
2.2.2 Arsitektur <i>File Server</i> .....	9
2.2.3 Protokol .....	10
2.2.4 Rsync .....	11
2.2.4.1 Algoritma Rsync .....	11
2.2.4.2 Dua <i>Signature</i> .....	13
2.2.5 MD5 Hash .....	15
2.2.6 CRC32 Hash .....	17
2.2.7 FTP .....	18

BAB 3 PERANCANGAN SISTEM.....	20
3.1 Spesifikasi Kebutuhan.....	20
3.1.1. Kebutuhan Fungsional .....	20
3.1.2. Kebutuhan Non-Fungsional .....	20
3.2 Perancangan Program .....	21
3.2.1. Flowchart Sinkronisasi File .....	21
3.2.2. Flowchart Sinkronisasi File dengan Algoritma Rsync.....	23
3.2.3. Activity Diagram.....	26
3.3 Perancangan Storage .....	27
3.4 Perancangan Pengujian .....	27
3.5 Perancangan Antarmuka Pengguna .....	27
3.5.1. Form Main Menu .....	27
3.5.2. Form Files .....	28
BAB 4 ANALISA DAN IMPLEMENTASI SISTEM .....	30
4.1 Implementasi Sistem .....	30
4.1.1. Implementasi Antarmuka .....	30
4.1.1.1. Implementasi Antarmuka Client .....	30
4.1.1.2. Implementasi Antarmuka Server.....	32
4.1.2. Implementasi Input .....	33
4.1.2.1. Implementasi Input Sinkronisasi File dengan Algoritma Rsync.....	33
4.1.2.2. Implementasi Input Sinkronisasi File dengan Protokol FTP .....	33
4.1.2.3. Implementasi Input Sinkronisasi File pada Lokal Komputer.....	34
4.1.3. Implementasi Output.....	35
4.1.4. Implementasi Proses .....	36
4.2 Pengujian Sistem.....	41
4.2.1. Tujuan Pengujian .....	41
4.2.2. Data Pengujian .....	41

4.2.3. Pengujian Performa untuk Menemukan Pembagi Blok Terbaik.....	42
4.2.4. Pengujian Data Literal dan performa .....	45
4.2.5. Pengujian Perbandingan Performa Antara <i>Weak Signature</i> dengan <i>Strong Signature</i> .....	47
BAB 5 KESIMPULAN DAN SARAN .....	49
5.1 Kesimpulan.....	49
5.2 Saran.....	49
DAFTAR PUSTAKA.....	50
LAMPIRAN .....	51

©UKDW

## DAFTAR GAMBAR

Gambar 2.1. Model client-server.....	7
Gambar 2.2. Middleware client server.....	9
Gambar 2.3. Inisiasi awal pencarian blok.....	12
Gambar 2.4. Proses Pendeteksian Perbedaan antara File Baru dan yang Lama. ....	12
Gambar 2.5. Perbedaan deteksi selesai. tiga blok dalam file baru cocok. ....	13
Gambar 2.6. Algoritma pencarian signature.....	14
Gambar 2.7. Cara kerja algoritma checksum MD5 .....	16
Gambar 2.8. Cara Kerja FTP .....	18
Gambar 3.1. Flowchart sinkronisasi file .....	22
Gambar 3.2. Flowchart komputer server .....	23
Gambar 3.3. Flowchart komputer client .....	24
Gambar 3.4. Activity diagram proses sinkronisasi .....	26
Gambar 3.5. Tab Connection Log .....	27
Gambar 3.6. Tab Files .....	28
Gambar 3.7. JFileChooser untuk menentukan file server directory .....	29
Gambar 4.1. Tampilan tab Server file.....	30
Gambar 4.2. Tampilan desain antarmuka server .....	32
Gambar 4.3. Tampilan sinkronisasi lokal .....	34
Gambar 4.4. Contoh file teks client-server sebelum sinkronisasi.....	35
Gambar 4.5. Rolling block menemukan blok berbeda. ....	37
Gambar 4.6. Skema sistem sinkronisasi .....	40

## DAFTAR TABEL

Tabel 2.1. Protokol berdasarkan lapisan OSI.....	10
Tabel 4.1. Tabel perbandingan file <i>A'(client)</i> dan <i>A(server)</i> .....	41
Tabel 4.2. Tabel pengujian sinkronisasi file <i>samplepossible-passwords2.txt</i> .....	42
Tabel 4.3. Tabel pengujian sinkronisasi file <i>Oxford English Dictionary 3.0.txt</i> .....	43
Tabel 4.4. Tabel pengujian sinkronisasi file <i>Skyrim FAQWalkthrough.txt</i> .....	43
Tabel 4.5. Tabel pengujian sinkronisasi file <i>BestMovieList (Times version).txt</i> .....	44
Tabel 4.6. Tabel perbandingan data literal file <i>BestMovieList (Times version).txt</i> ...	46
Tabel 4.7. Tabel perbandingan data literal file <i>FAQWalkthrough.txt</i> .....	46

©UKDW

## DAFTAR GRAFIK

Grafik 4.1. Grafik pengujian file samplepossible-passwords2.txt .....	42
Grafik 4.2. Grafik pengujian file Oxford English Dictionary 3.0.txt.....	43
Grafik 4.3. Grafik pengujian Skyrim FAQWalkthrough.txt.....	44
Grafik 4.4. Grafik pengujian file BestMovieList (Times version).txt .....	44
Grafik 4.5. Perbandingan checksum pada file Skyrim FAQWalkthrough.txt .....	47
Grafik 4.6. Perbandingan checksum pada file samplepossible-passwords2.txt.....	47

©UKDWN

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Seiring dengan pesatnya perkembangan teknologi informasi dewasa ini, kebutuhan *user* terhadap data dan informasi semakin dibutuhkan. Data yang kini semakin penting dan mahal berimbas terhadap peningkatan upaya *user* untuk menjaga keamanan data. *User* kini menyadari pentingnya menjaga data dari kehilangan data akibat kecerobohan maupun dari kerusakan data yang dapat menyebabkan kerugian waktu dan biaya. Kehilangan data dapat disebabkan oleh serangan pada data, kerusakan pada *harddisk*, kerusakan pada perangkat keras, *software corruption*, virus komputer, kesalahan *user*, maupun bencana alam. Terdapat beberapa upaya menjaga data seperti menggunakan perangkat lunak anti-malware sebagai perlindungan awal pada komputer. Solusi pengamanan lain yang dapat dilakukan adalah membuat salinan data tersebut, dengan melakukan sinkronisasi file.

Sinkronisasi file dalam komputasi merupakan salah satu dari berbagai cara pertukaran data, dimana sinkronisasi file merupakan sebuah proses untuk memastikan bahwa file di beberapa komputer selalu diperbaharui dengan aturan-aturan tertentu, dengan kata lain sinkronisasi file dilakukan untuk menyamakan file dari suatu media penyimpanan file ke media penyimpanan file lainnya. Namun untuk melakukan sinkronisasi dengan file dengan ukuran yang besar, dibutuhkan waktu yang tidak sedikit, terlebih lagi jika dilakukan antar komputer dengan *bandwith* yang terbatas. Untuk itu diperlukan sebuah solusi ataupun alternatif untuk meminimalkan pemborosan *bandwith*.

Sistem yang akan dibangun oleh penulis akan mendeteksi perbedaan file atau penambahan *byte* file pada file *client*, dan mengirim perbedaan *byte* untuk disusun menjadi file yang sama. Sistem ini diharapkan dapat menjawab keterbatasan *bandwith* selama proses sinkronisasi.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dikemukakan diatas, maka rumusan masalah didalam penelitian ini adalah :

1. Bagaimana membangun sistem berbasis *client-server* yang dapat memanfaatkan file lama untuk proses sinkronisasi *plain text* ?
2. Bagaimana menentukan ukuran blok untuk memperoleh waktu eksekusi program tercepat, serta menentukan nilai *speedup* dan pengaruhnya terhadap sistem.
3. Bagaimana pengaruh penggunaan *weak signature* dan *strong signature* terhadap sistem ? *Signature* diperlukan untuk menentukan *byte* berbeda.

## 1.3 Batasan Masalah

Demi menghindari pembahasan yang meluas mengenai penerapan *client-server* dan sinkronisasi, diperlukan adanya pembatasan sistem. Berikut ini merupakan batasan sistem pada perangkat lunak sinkronisasi file text berbasis *client-server*:

1. Melakukan proses validasi untuk membandingkan file antar komputer menggunakan checksum MD5.
2. Penelitian backup data ini lebih berfokus kepada update perubahan file, di mana versi lama dari file sudah ada di *server*.
3. Folder *depth* pada penelitian ini adalah 0.
4. Diasumsikan, jam(penunjuk waktu) di tiap komputer benar.

## 1.4 Tujuan Penelitian

Tujuan :

1. Untuk menerapkan algoritma Rsync untuk aplikasi sinkronisasi file text berbasis *client-server*.
2. Meneliti dan menganalisa rentang waktu eksekusi program serta hasil pengiriman data literal.



3. Untuk mengimplementasikan teori yang diperoleh ke dalam suatu karya ilmiah yang bernilai akademis dan dapat dipertanggung jawabkan.

Manfaat:

1. Seorang *user* dapat melakukan sinkronisasi file text dengan memanfaatkan file lama sehingga proses pertukaran data menjadi lebih *efisien* dan menghemat penggunaan *bandwith*.

### **1.5 Metodologi Penelitian**

Metodologi yang digunakan dalam penelitian ini terdiri dari tahapan sebagai berikut :

1. Studi pustaka dengan mempelajari buku-buku yang berkaitan dengan penerapan algoritma Rsync pada teknologi *client-server*.
2. Melakukan konsultasi dengan dosen pembimbing terkait perancangan dan pelaporan tugas akhir penelitian.
3. Melakukan pembuatan aplikasi sinkronisasi meliputi perancangan, pemrograman, pengujian dan perbaikan kesalahan.

### **1.6 Sistematika Penulisan**

Pada bab 1 atau bab PENDAHULUAN berisikan tentang latar belakang masalah yang akan diteliti, rancangan penelitian yang akan dilakukan, metode penelitian, dan sistematika penulisan. Penelitian ini mengenai pembuatan sistem sinkronisasi file text satu arah berbasis *client-server*.

Pada bab 2 atau bab TINJAUAN PUSTAKA berisikan landasan teori yang memuat penjelasan tentang konsep dan prinsip utama yang diperlukan untuk memecahkan masalah. Pada bab ini, akan berisikan tentang konsep-konsep *client-server* dan algoritma Rsync yang akan digunakan.

Pada bab 3 atau bab ANALISIS DAN PERANCANGAN SISTEM berisikan tentang analisis permasalahan yang akan diatasi dengan membangun backup data dengan algoritma Rsync. Pada bab ini juga memuat perencanaan sistem yang akan dibangun.

Pada bab 4 atau bab IMPLEMENTASI DAN ANALISIS SISTEM berisikan tentang hasil implementasi, dan analisis dari perangkat lunak yang dibuat.

Pada bab 5 atau bab KESIMPULAN DAN SARAN memuat pernyataan singkat dari hasil analisis kegiatan penelitian. Selain itu pada bab ini turut memuat saran untuk kegiatan pengembangan penelitian ini dimasa mendatang.

Selain berisi bab-bab utama tersebut, skripsi ini juga dilengkapi dengan Intisari, Daftar Isi, Daftar Gambar, Daftar Tabel, Daftar Pustaka dan Lampiran.

©UKDW

## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah dilakukan penelitian dan pengujian sistem yang telah dibuat, dapat disimpulkan bahwa :

1. Sistem *client-server* yang telah dibuat dapat diimplementasikan algoritma Rsync. Rsync dapat membantu proses sinkronisasi file *plain text* untuk mengirim perbedaan *byte* sehingga dapat memanfaatkan file lama yang ditemukan sama.
2. Untuk memperoleh waktu eksekusi program terbaik, skala perbandingan antara total ukuran file dengan ukuran blok adalah 1000:1.
3. Nilai *speedup* berpengaruh terhadap nilai data literal, semakin tinggi nilai *speedup*, maka semakin tinggi tingkat efektifitas pengiriman *byte* berbeda.
4. Jika dilihat dari rentang waktu eksekusi program, sistem yang menggunakan *weak signature* akan menghasilkan waktu eksekusi program yang lebih cepat jika dibandingkan menggunakan *strong signature*.

#### 5.2 Saran

Untuk pengembangan lebih lanjut, saran yang dapat diberikan adalah sebagai berikut :

1. Mencari solusi penurunan satu *byte* setiap ditemukan blok berbeda, penurunan satu *byte* kebawah memerlukan banyak iterasi yang berimbas kepada waktu eksekusi program .
2. Mencatat tiap indeks *byte* file yang berbeda sehingga dapat digunakan oleh sistem jika kembali melakukan sinkronisasi dengan file yang sama.

## DAFTAR PUSTAKA

- Gao, Wei.(2001). Performance Analysis of Rsync+.North Carolina: Chapel Hill.
- Hal, Bryan (2011). Beej's Guide to Network Programming. Jorgensen Publishing.
- How Rsync Works. <http://rsync.samba.org/how-rsync-works.html> di akses tanggal 19 Desember 2012.
- Jenkov Tutorial. <http://tutorials.jenkov.com/rsync/index.html> diakses tanggal 12 november 2013.
- Muthitacharoen, A., Chen, B., & Mazières, D.(n.d). A Low-bandwidth Network File System. MIT Laboratory for Computer Science.
- Niemeyer, P & Knudsen k.(2000). Learning Java. O'Reilly publisher.
- Reilly, D & Reilly, M (2002) Java™ Network Programming and Distributed Computing, Boston : Pearson Education, Inc.
- Tridgell, A (1999).Efficient Algorithms for Sorting and Synchronization. Australia : The Australian National University.
- Tridgell, A.(2000).Rsync Presentation.  
<http://olstrans.sourceforge.net/release/OLS2000-rsync/OLS2000-rsync.html> diakses tanggal 2 februari 2013.