

IMPLEMENTASI DATA STORAGE SERVER PADA JARINGAN INTRANET MENGUNAKAN OWNCLOUD

Edwin Wicaksono Saputro, Dadan Irwan, Haryono
Program Studi Teknik Komputer Unisma Bekasi
Jl. Cut Mutia No. 83 Bekasi
Email : ewinz89@gmail.com

ABSTRACT

Information technology is developing so rapidly, both in terms of hardware and software. One of the utilization of technology that tends to be used today is data processing and data storage system. The increasing use of data makes the need of data storage facilities used must have a large capacity, easy to access from anywhere and anytime, has a high speed and of course good data security. The presence of cloud computing technology makes the problem can be solved well. The large number of opensource software allows users to build their own personal data storage server that can be fully controlled by the user. Making data storage server by utilizing owncloud software can be done quickly and easily because it does not require special hardware. The use of data storage server is also fairly easy for users because of its user friendly appearance, in addition to the speed, convenience, features and security available to make users more comfortable using this storage server data than using conventional storage media.

Keyword : data storage, cloud computing, owncloud, data storage server.

ABSTRAK

Teknologi informasi berkembang begitu pesat, baik dari sisi perangkat keras maupun perangkat lunak. Salah satu pemanfaatan teknologi yang cenderung digunakan saat ini adalah pengolahan data dan sistem penyimpanan *data*. Semakin meningkatnya penggunaan *data* membuat kebutuhan fasilitas penyimpanan *data* yang digunakan harus memiliki kapasitas yang besar, mudah di akses dari mana dan kapan saja, memiliki kecepatan yang tinggi dan tentu keamanan *data* yang baik. Hadirnya teknologi *cloud computing* membuat permasalahan tersebut dapat diselesaikan dengan baik. Banyaknya *software opensource* membuat pengguna dapat membangun sendiri *data storage server* pribadi yang dapat di kontrol penuh oleh pengguna. Pembuatan *data storage server* dengan memanfaatkan *software owncloud* bisa dilakukan dengan cepat dan mudah karena tidak membutuhkan perangkat keras khusus. Penggunaan *data storage server* ini juga terbilang mudah bagi pengguna karena tampilannya yang *user friendly*, selain itu kecepatan, kemudahan, fitur dan keamanan yang tersedia membuat pengguna lebih nyaman menggunakan *data storage server* ini dibandingkan menggunakan media penyimpanan konvensional.

Keyword : penyimpanan data, cloud computing, owncloud, data storage server

1. Pendahuluan

Teknologi informasi saat ini berkembang demikian pesat, baik dari sisi perangkat keras maupun dari sisi perangkat lunak atau aplikasi. Salah satu kecenderungan pemanfaatan teknologi informasi, khususnya dalam pengelolaan data berukuran besar adalah pusat data atau data center, yaitu suatu fasilitas yang

digunakan untuk penempatan kumpulan server atau sistem computer dan sistem penyimpanan data yang dilengkapi pula dengan sistem keamanan fisik (Henriyadi, 2008).

Semakin meningkatnya pengolahan *data* digital membuat semua orang membutuhkan media penyimpanan *data* yang fleksibel, berkapasitas besar dan tentu

saja aman. Namun saat ini media penyimpanan *data* yang digunakan oleh sebagian besar orang masih berupa media penyimpanan fisik, dimana perangkat ini rentan dari kerusakan, kehilangan dan sangat tidak fleksibel karena kita harus membawanya kemana saja kita pergi sehingga membuat kita repot. Meski media penyimpanan *data* fisik ini hanya digunakan di lingkungan kerja tertentu seperti kantor, sekolah, atau rumah, penggunaannya tidak efisien dan efektif (Harris, 2012).

Hadirnya teknologi *cloud* computing membuat permasalahan pengelolaan data dapat teratasi. Kita dapat memilih pusat data mana yang cocok dengan lingkungan kerja kita. Tetapi bila kita menginginkan kendali penuh terhadap *server* penyimpanan *data*, ada beberapa alternative aplikasi untuk membangun *server* penyimpanan *data* pribadi. Salah satu aplikasi tersebut adalah *OwnCloud*. Dengan memiliki *server* sendiri maka keamanan *data* dapat dipantau secara *real time* dan manajemen *data* dan user dapat dilakukan lebih leluasa (Afrianto, 2014).

2. Bahan dan Metode Penelitian

2.1. Bahan

Pada penelitian ini digunakan komputer server dan komputer client dengan masing-masing spesifikasi yang tersaji dalam tabel 1 dan tabel 2.

2.2. Metode Penelitian

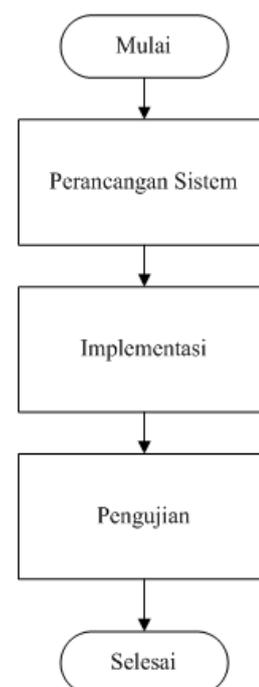
Penelitian ini terbagi menjadi 3 tahapan, yaitu tahap perancangan sistem, implementasi dan pengujian (Gambar 1.).

Tabel 1. Spesifikasi Perangkat Keras Komputer Server

<i>Processor</i>	<i>Intel Core i3 2,3 GHz</i>
<i>Memory</i>	<i>RAM 4 GB</i>
<i>Harddisk</i>	<i>500 GB</i>
<i>NIC</i>	1 buah

Tabel 2. Spesifikasi Perangkat Keras Komputer Client

<i>Processor</i>	<i>AMD Athlon X2 Dual Core 64-bit</i>
<i>Memory</i>	<i>RAM 2 GB</i>
<i>Harddisk</i>	<i>160 GB</i>
<i>NIC</i>	1 buah



Gambar 1. Tahapan penelitian

2.2.1 Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem pada *data storage server*, perancangan dikerjakan dari topologi jaringan yang digunakan, kebutuhan perangkat yang digunakan, instalasi *software*, dan konfigurasi *software*.

2.2.2 Implementasi

Data storage server yang dirancang bekerja pada jaringan *LAN* perusahaan. *Data storage server* ini menyimpan *data client* secara permanen di *server* dan menyimpannya sementara di komputer *client*, hal ini bertujuan untuk melindungi *file* asli dari kerusakan.

2.2.3 Pengujian

Untuk mengetahui kinerja dari *data storage server* yang telah dirancang ini, maka penulis membuat skenario pengujian untuk mengetahui apakah *data storage server* ini sudah berjalan dengan baik atau belum.

Adapun skenario pengujian yang akan dilakukan adalah sebagai berikut :

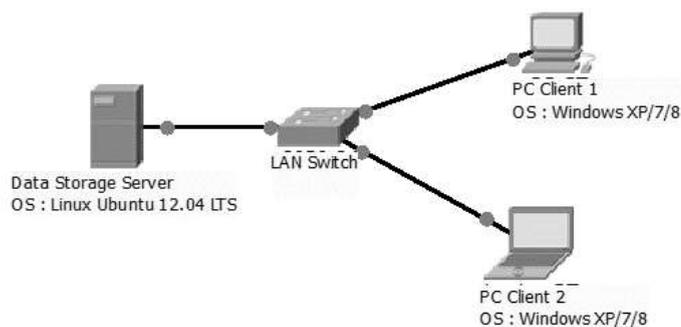
1. Apakah manajemen *user* dapat dilakukan oleh *admin server*.
2. Apakah *data user* dapat di *upload* ke *data storage server*.

3. Apakah *data user* yang telah di *upload* dapat di akses oleh *user* lain.
4. Apakah *data user* yang telah di *upload* dapat di akses oleh *user* lain yang tidak memiliki hak akses.
5. Apakah *data* yang telah di *upload* dapat dirubah secara *real time*.
6. Apakah sistem mendukung media *sharing*.
7. Apakah *user* dapat mengirim pesan singkat melalui internal message atau chat.

Sedangkan pelaksanaan pengujian meliputi pengujian manajemen *user*, pengujian *upload file*, pengujian *sharing file*, pengujian hak akses, pengujian *edit file*, pengujian *media sharing* dan pengujian *Internal Message* Atau *Chat*.

3. Hasil dan Pembahasan

3.1. Topologi Sistem



Gambar 2. Topologi Perancangan Sistem

- a. *Data storage server* adalah komputer yang berfungsi sebagai pusat penyimpanan *data*. Komputer ini menyediakan ruang penyimpanan dan juga layanan manajemen *data* milik *client*. Semua *data client* disimpan secara permanen di komputer ini.
- b. *LAN switch* berfungsi menghubungkan komputer *client* 1 dan 2 supaya terhubung dengan komputer *server*.
- c. *Client* berupa *laptop* atau komputer *desktop*. Pada komputer *client data* yang disimpan bersifat sementara

3.2. Pengalamatan IP

Peran jaringan terutama alamat *IP* pada sistem yang akan dibangun ini sangatlah penting. Bila terjadi kesalahan saat melakukan pengalamatan pada salah satu

perangkat, maka perangkat tersebut tidak akan dapat digunakan. Oleh karena itu, penulis membuat pengalamatan *IP* sebagai berikut :

Tabel 3. Pengalamatan *IP*

Nama <i>Hardware</i>	<i>IP Address</i>	<i>Subnet Mask</i>	<i>Gateway</i>
Komputer <i>Server</i>	192.168.43.8	255.255.255.0	192.168.43.1
Komputer <i>Client 1</i>	192.168.43.9	255.255.255.0	192.168.43.1
Komputer <i>Client 2</i>	192.168.43.10	255.255.255.0	192.168.43.1

3.3. Konfigurasi *IP Address Server*

1. Untuk melakukan konfigurasi *Linux Ubuntu 12.04* pertama – tama buka *terminal Linux Ubuntu*. Kemudian masuk sebagai *root* dengan peritnah :
#Sudo -i
2. Kemudian buka *file network interface* dari *terminal Linux* menggunakan perintah :
#Nano /etc/network/interfaces
3. Setelah *file* konfigurasi terbuka, buat konfigurasi seperti berikut :
#interface eth0
Auto eth0
iface eth0 inet static
address 192.168.43.2
netmask 255.255.255.0
network 192.168.43.0
broadcast 192.168.43.1
setelah itu simpan dengan menekan tombol *CTRL+X*, lalu tekan *Y* dan *Enter*.
4. Untuk menerapkan konfigurasi perlu dilakukan *restart network interface* dengan perintah :
#/etc/init.d/network restart

3.4. Konfigurasi *IP Address Client*

1. Buka *control panel* yang terdapat *start menu*. Kemudian pilih *Network and*

Internet, selanjutnya pilih *Network and Sharing Center*.

2. Kemudian pilih *Change Network Adapter Setting*.
3. Pada *Local Area Connection* lakukan *double click*, akan muncul *box properties*.
4. Selanjutnya lakukan *double click* pada *Internet Protocol Version 4 (TCP/IPv4)*. Maka akan muncul *diolog box* untuk mengisi *IP Address*.

3.5. Konfigurasi *Web Server*

Agar *web server* dapat berjalan dengan baik maka perlu dilakukan konfigurasi. Berikut adalah langkah–langkah konfigurasi *web server* :

1. Aktifkan modul *rewrite* dan *headers* dengan perintah :
#Sudo a2enmod rewrite
#Sudo a2enmod headers
2. Kemudian lakukan penyesuaian konfigurasi di */etc/apache2/sites-available/default*, dengan perintah berikut :
#Sudo nano /etc/apache2/sites-available/default
3. Kemudian akan muncul *text viewer*, lalu cari baris – baris berikut :

```
<Directory /var/www/>
Options Indexes FollowSymLinks
MultiViews
AllowOverride None
Order allow,deny
allow from all
</Directory>
```

4. Lalu ganti baris :
AllowOverride None

Menjadi :

AllowOverride all

5. Selanjutnya *restart apache service* dengan perintah berikut :
#Sudo /etc/init.d/apache2 restart

3.6. Konfigurasi *OwnCloud Server*

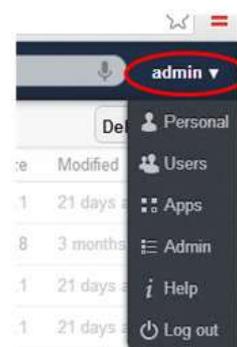
Berikut adalah langkah – langkah konfigurasi *OwnCloud* pada *server* :



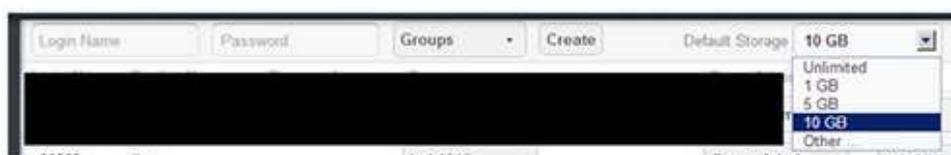
Gambar 3. Konfigurasi pembuatan akun admin

1. Sebelum dapat digunakan, setelah proses instalasi selesai *user* akan diminta untuk mengisi sebuah *form data* (gambar 3). Isi semua form seperti *Username, Password, Data Folder, Database User, Database Password, Database Name, dan Database Host*. Lalu pilih *Next* untuk melanjutkan.
2. Selanjutnya akan muncul halaman *web admin*.
3. Langkah selanjutnya adalah pembuatan akun *client*. Setelah login sebagai admin, arahkan kursor ke pojok kanan atas lalu klik *admin*. Akan muncul *dropdown menu* (gambar 4). Lalu klik *User*.

4. Selanjutnya akan muncul tampilan seperti pada gambar dibawah. Kemudian buat akun *client* dengan mengisi form yang ada pada bagian atas, seperti *Login Name, Password, Groups, dan Default Storage*. Lalu pilih *create* untuk membuat akun.



Gambar 4. *Dropdown menu*



Gambar 5. Pembuatan akun *client*

3.7. Konfigurasi OwnCloud Client

Agar *client* dapat mengakses *data* yang ada di *server*, maka perlu dilakukan beberapa konfigurasi. Berikut langkah – langkahnya :

1. Masukkan alamat *IP server*, lalu pilih *Next*(Gambar 6).
2. Bila berhasil maka akan muncul tampilan seperti gambar 7. Lalu pilih *Finish*



Gambar 6. OwnCloud Connection Wizard

Result

This page shows the status of the connection.

```
Trying to connect to ownCloud at http://192.168.0.10/nfbscloud...
Successfully connected to http://192.168.0.10/nfbscloud: ownCloud version 5.0.5 (5.0.6)

Local sync folder C:/Documents and Settings/Administrator/ownCloud already exists, setting it up
for sync.

The remote folder clientsync already exists. Connecting it for syncing.
A sync connection from C:/Documents and Settings/Administrator/ownCloud to remote directory
clientsync was set up.

Successfully connected to ownCloud!
Press Finish to permanently accept this connection.
```



Gambar 7. Status koneksi OwnCloud client

3.8. Pengujian

3.8.1. Pengujian Manajemen User

Akun *user* yang digunakan hanya dapat dibuat dan dihapus oleh *admin* dari komputer *server*. Untuk membuat akun *user*, pada *web admin* cukup memasukkan *username*, password dan grup. Untuk mengujinya maka dilakukan pembuatan akun seperti gambar 8.

Pada gambar 8, menunjukkan akun berhasil dibuat. Selanjutnya akun dapat diakses oleh *user* yang dari komputer yang mereka gunakan dengan menggunakan nama dan *password* yang telah diberikan oleh *admin*.

3.8.2. Pengujian Upload File

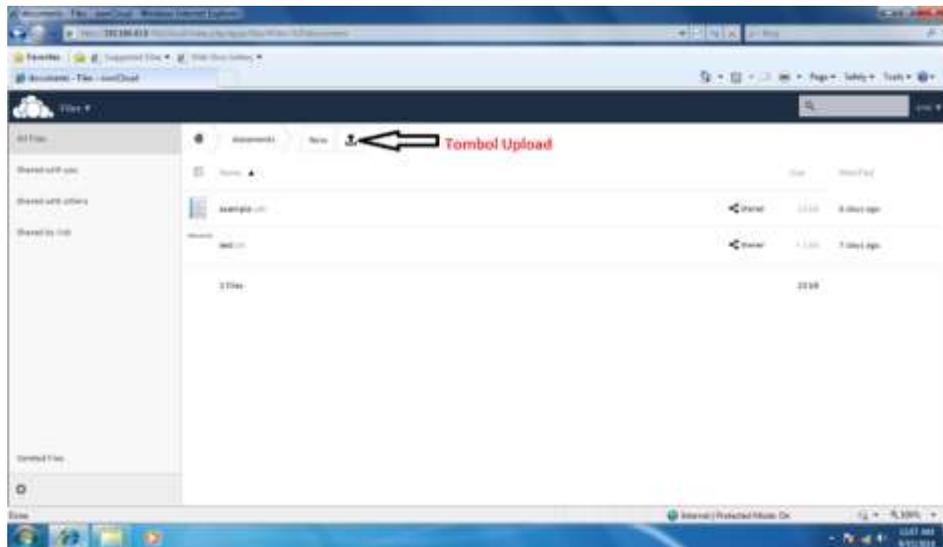
Untuk mengetahui batasan *file* yang bisa di *upload*, dilakukan *upload file* dengan

beberapa variasi ukuran dan jenis *file*. *File* – *file* tersebut di *upload* satu per satu untuk

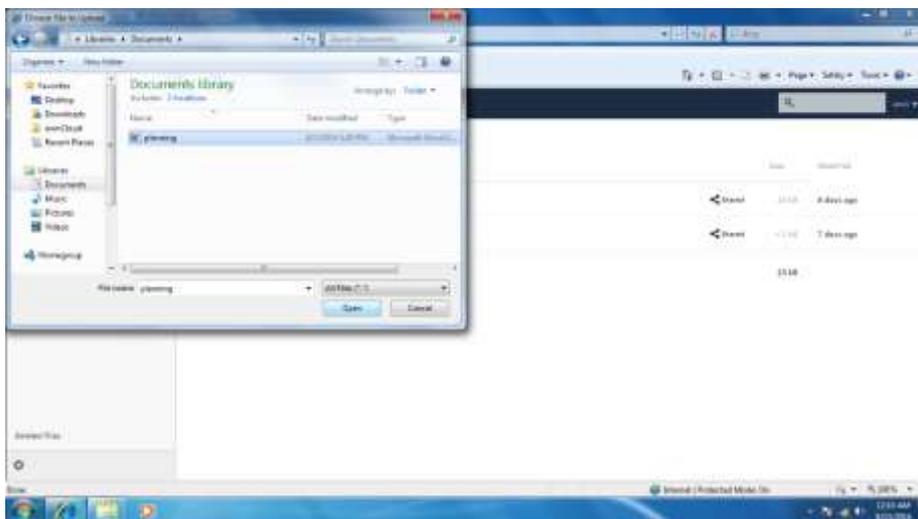
mengetahui batasan maksimum kemampuan *upload file* ke *server seperti* (gambar 9).



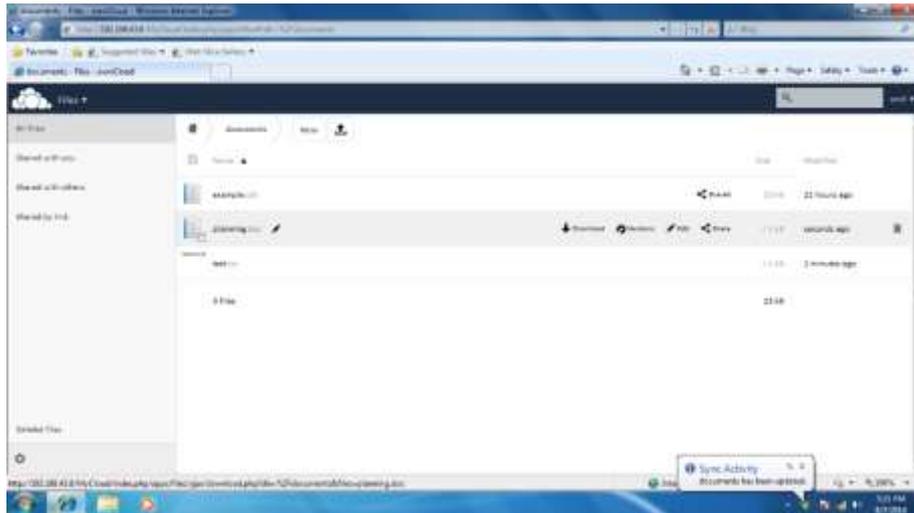
Gambar 8. Hasil pembuatan akun *user*



Gambar 9. *Upload File*



Gambar 10. Pemilihan *File Upload*

Gambar 11. Hasil Pengujian *Upload File*

Dari gambar diatas dapat dilihat bahwa proses *upload* telah berhasil. Pada menu notifikasi muncul pemberitahuan bila proses *upload* berhasil.

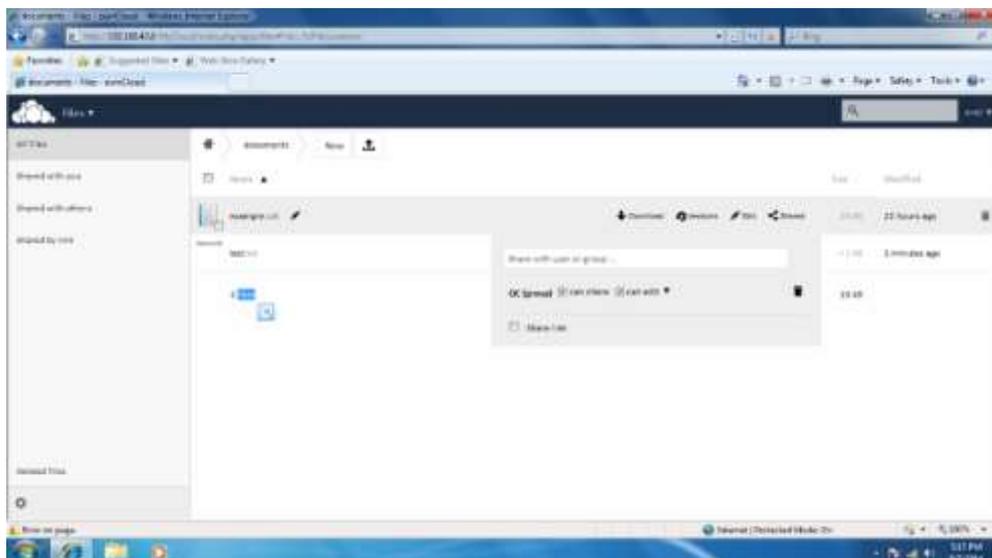
3.8.3. Pengujian *Sharing File*

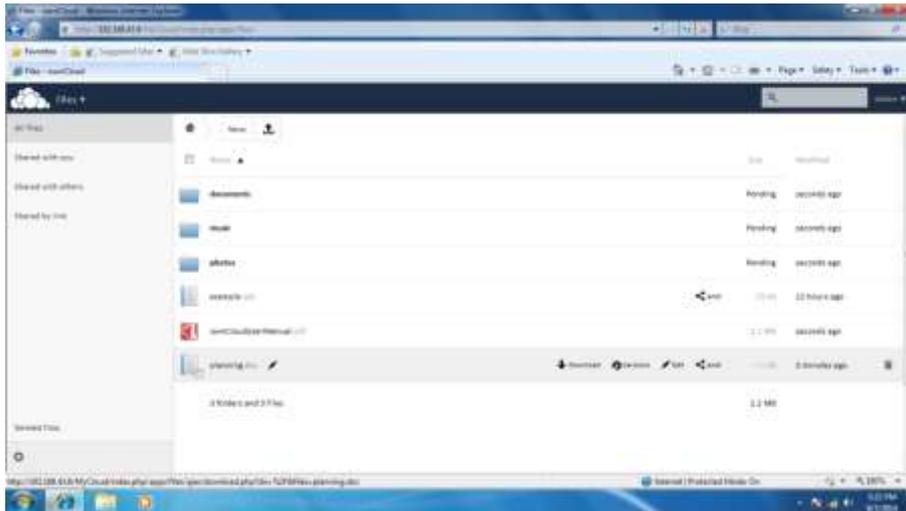
Setelah proses *upload file* berhasil, *file* tersebut dapat langsung di *sharing* ke semua *user* atau *user* tertentu saja. Proses *sharing file* ini cukup mudah karena saat kita mengarahkan kursor diatas *file* yang akan di *share* maka muncul pilihan *share*. Bila di *click* maka akan tampil pilihan lagi apakah *file* akan di *share* ke semua *user* atau *user*

tertentu saja, seperti pada gambar 12 & gambar 13.

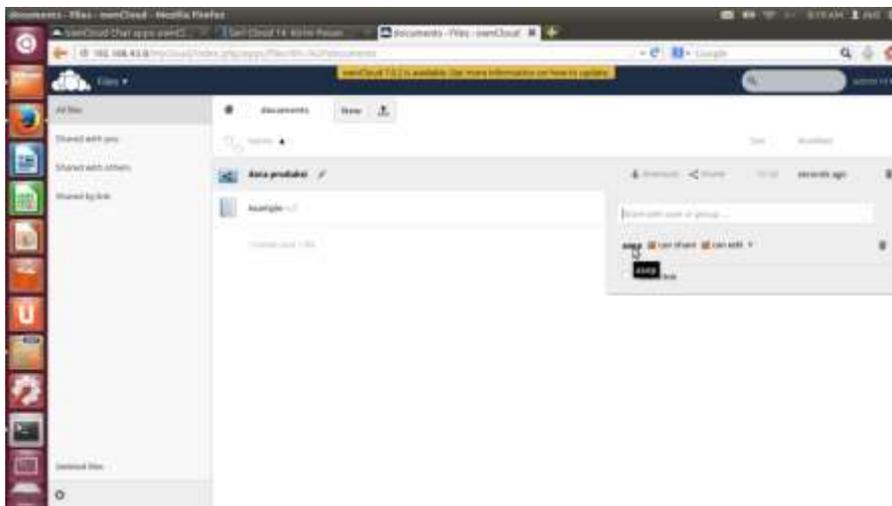
3.8.4. Pengujian Hak Akses

Setelah melakukan *upload data* ke *server*, *user* dapat melakukan *share data* ke semua *user* atau *user* tertentu yang di inginkan. Untuk menguji hal tersebut dilakukan pengujian dengan melakukan *sharing data* pada salah satu *user*, lalu *user* lain yang tidak atau belum diberi hak akses mencoba untuk mengakses *data* tersebut dari *server* (gambar 14).

Gambar 12.. Pengujian *Sharing File*



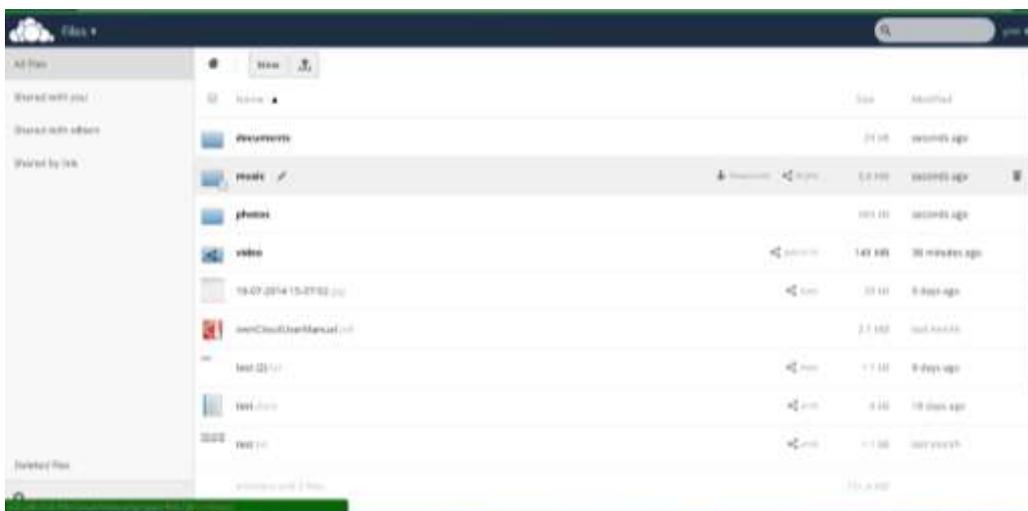
Gambar 13. Hasil *Sharing File*



Gambar 14. *User* yang diberi hak akses

Setelah dilakukan pengujian, dapat dilihat bahwa *user* yang tidak memiliki hak akses tidak dapat mengakses *data user* lain

karena *data* yang dicari tidak muncul pada halaman *web* dan juga *folder share user* tersebut.



Gambar 15. *Data* yang di *share* tidak muncul pada *user* yang tidak diberi hak akses

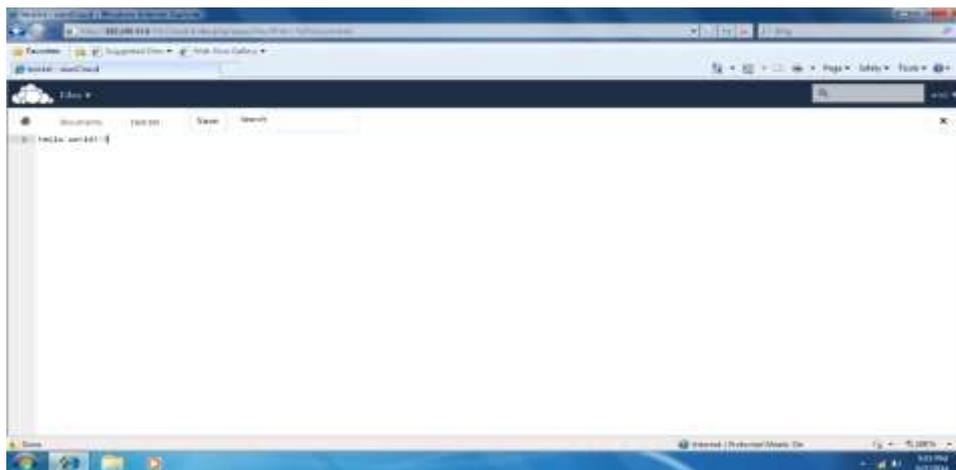


Gambar 16. Data yang di *share* muncul pada *user* yang diberi hak akses

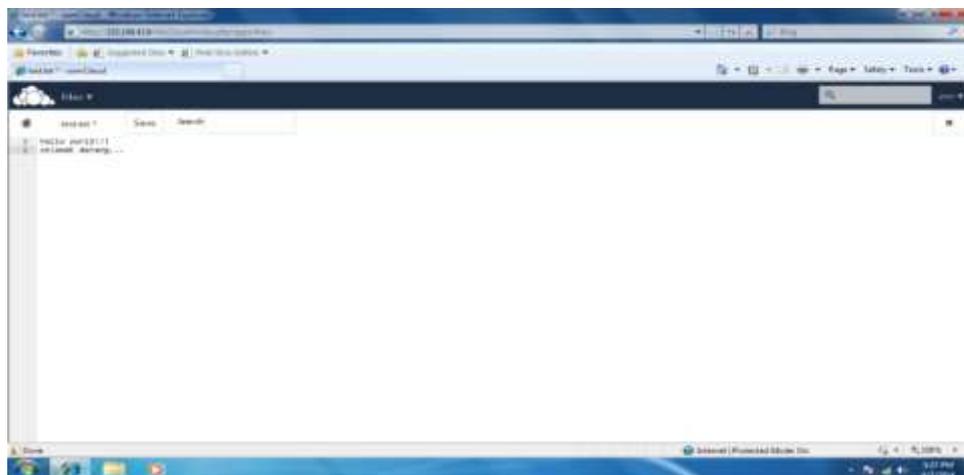
3.8.4. Pengujian Edit File

File yang di *upload* dan telah di *sharing* seharusnya dapat dirubah sewaktu – waktu oleh *user* yang meng-*upload* dan juga

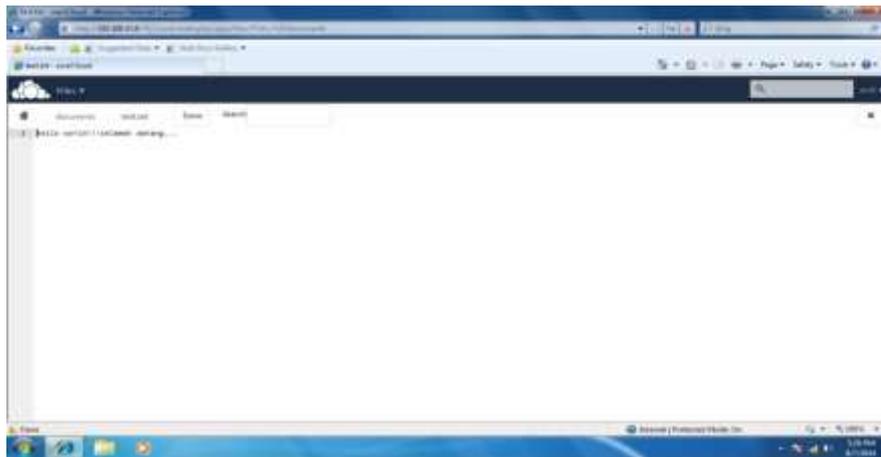
user lain yang memiliki hak akses. Untuk mengujinya maka dilakukan pengujian sebagai berikut:



Gambar 17. Isi awal *File*



Gambar 18. Perubahan Isi *file* oleh *user* pertama



Gambar 19. Tampilan isi *file* pada *user* kedua

Dari hasil pengujian yang dilakukan terlihat bahwa proses perubahan *file* dapat dilakukan dengan mudah dan *server* dapat langsung melakukan update isi *file*.

3.8.5. Pengujian Media *Sharing*

Pada pengujian ini dilakukan uji coba *streaming video* dan juga musik. Dengan

melakukan pengujian ini akan dapat diketahui seberapa cepat dan stabil kinerja *server*.

Pada pengujian pertama dilakukan *streaming file* musik dengan format *mp3* untuk menguji kemampuan musik *streaming*.



Gambar 20. Hasil uji coba musik *streaming*

Pada pengujian musik *streaming*, musik yang dipilih dapat dimainkan dengan baik dan lancar tanpa ada jeda, suara yang dihasilkan juga baik.

Untuk pengujian kedua (gambar 21) dilakukan uji coba *video streaming, file* yang digunakan adalah *file* dengan format *mp4*.

Pada pengujian *video streaming, video* dapat dimainkan dengan baik tetapi gambar

yang ditampilkan tidak terlalu bagus karena *video player* bawaan *owncloud* tidak terlalu bagus.

3.8.6. Pengujian *Internal Message* Atau *Chat*

Selain fungsi utama sebagai pusat penyimpanan *data, server* ini juga menyediakan fitur *internal message* sebagai sarana komunikasi antar *user* melalui pesan

singkat. Untuk menguji fitur ini dilakukan komunikasi antara dua *user* dengan

mengirim beberapa pesan singkat melalui *owncloud chat*.



Gambar 21. Hasil uji coba *video streaming*



Gambar 22. Tampilan fitur *messegging*

Dengan menggunakan fitur ini *user* cukup memilih nama *user* yang ingin di *chat*, kemudian tulis pesan di *box messege*.

cukup cepat, dan pada fitur ini juga pengguna diberi fasilitas *emoticon* yang membuat fitur ini terasa seperti aplikasi *messegger* yang ada di sosial media seperti *facebook*, *whatsapp*, dll.

Pada gambar 23 dapat dilihat hasil pengujian fitur ini. Pesan yang dikirim



Gambar 23. Hasil Pengiriman Pesan

4. Kesimpulan dan Saran

4.1. Kesimpulan

Adapun kesimpulan dari hasil penelitian ini adalah :

1. *Data storage server* merupakan alternatif media penyimpanan yang sangat baik karena jika dibandingkan dengan media penyimpanan fisik (flashdisk , CD, dll) data storage server ini memudahkan pengguna untuk mengakses data dari mana saja dan kapan saja, proses pengelolaan data juga lebih cepat walaupun tidak terlalu jauh perbedaannya dengan flashdisk, dan tentu lebih praktis karena pengguna tidak perlu repot membawanya ke mana-mana sehingga resiko kehilangan data lebih kecil.
2. Dengan memiliki *data storage server* pertukaran data akan lebih mudah dibandingkan dengan media penyimpanan fisik (flashdisk, CD, dll) karena bisa dilakukan di mana dan kapan saja, dan untuk berbagi data bisa dilakukan dengan cepat tanpa perlu bertemu user yang meminta data.

4.2. Saran

Dari implementasi *data storage server* ini, penulis masih belum bisa memanfaatkan fitur LDAP enhancements, mobile syncing, dan external storage access yang ada pada aplikasi *ownCloud*. Untuk perancangan *data storage server* diwaktu yang akan datang diharapkan dapat lebih baik dan dapat memanfaatkan fitur – fitur yang ada pada *ownCloud*.

Daftar Pustaka

- Afrianto, Dedy S. 2014. *The Power Of Owncloud*. Yogyakarta: Andi
- Babu, M. 2014. Implementation of Privacy-Preserving Public Auditing Methods For Secure Cloud Storage. *International Journal of Engineering Research*. 2(5): 82 – 98
- Elyyani. 2012. Metode Manajemen Backup Data Sebagai Upaya Penyelamatan Data Online Web Lapan Bandung. *Jurnal Sistem Informasi*. 13(1): 22 – 27
- Fajrin, T. 2012. Analisis Sistem Penyimpanan Data Menggunakan Sistem Cloud Computing Studi Kasus SMKN 2 Karanganyar. *Indonesia Journal of Network And Security*. 1(1): 31 – 35
- Harris, Robin. 2012. *Cloud Storage*. Arizona: TechnoQWANLLC
- Henriyadi. 2008. Data Center Dan Implementasinya Pada Perpustakaan. *Jurnal Perpustakaan Pertanian*. 17(2): 41 – 47
- Kulkarni, G. 2012. Cloud Computing as Service. *International Journal of Engineering Research And Application*. 2(1): 945 – 950
- Nugroho, Adi. 2014. Implementasi Komputasi Awan Menggunakan Teknologi Google App Engine Dan Amazon Web Service. *Jurnal Teknik Informatika*. 1(1): 1 – 12
- Purbo, Onno W. 2012. *Membuat Sendiri Cloud Computing Server Menggunakan Open Source*. Yogyakarta: Andi
- Silbershatz, Abraham. 2005. *Operating System Concepts Seventh Edition*. Danvers: Jhon Wiley & Sons.inc
- Sofana, Iwan. 2010. *Cisco CCNA & Jaringan Komputer*. Bandung: Informatika

Subrata, L. 2013. Data Storage Engine Sebagai Media Penyimpanan Dalam Jaringan Lokal. *Jurnal Teknik Informatika*. 1(1): 1 – 11

Zefriansyah. 2013. Pemanfaatan Cloud Storage Sebagai Media Penyimpanan Data Terstruktur Pada Opengoverment. *Jurnal Nasional Teknologi Informasi*. 1(1): 1 – 35