

# IMPLEMENTASI *VIRTUAL DATA STORAGE* MENGGUNAKAN OPENFILER DENGAN PROTOKOL iSCSI PADA KANTOR CAMAT JATINEGARA JAKARTA

Muhammad Afad Nawal<sup>1</sup>, Anton<sup>2</sup>

Teknik Informatika, STMIK Nusa Mandiri,  
Jl. Kramat Raya No. 18, Jakarta, Indonesia  
[afadnawal8@gmail.com](mailto:afadnawal8@gmail.com)

Manajemen Informatika, AMIK BSI Tangerang,  
BSD Sektor XIV Blok C1/1, Tangerang  
[anton@bsi.ac.id](mailto:anton@bsi.ac.id)

**Abstract**—With the growth of technology, information and data is the thing that never lost and will always grows up in number, this makes the existence of problems in the storage area as the place where data and information collected to. Some problem will appear due to full storage area. The presence of SAN assist in issues, each issue will lose data in the event of damage to the hard drive is the most important. Because it was developed by RAID technology that can solve the problems of data. This technology used as the duplication of data, so if there is a drive has failed, the data stored in it will still be accessible by the user. In this research, the authors tried to make virtual data storage using free open source OS Openfiler 2.3 with protocols iSCSI and do some experiment, saving data using RAID system on VMware. The results of this study is a data storage that can prevent data loss in the event of a damage to the hard drive.

**Keywords:** *Virtual Data Storage*, SAN (Storage Area Network), RAID (Redundant Array Of Independent Disks), iSCSI

**Intisari**—Seiring dengan perkembangan teknologi yang selalu berkembang, data dan informasi merupakan hal yang tidak akan pernah hilang dan akan terus meningkat jumlahnya, hal ini menjadikan adanya permasalahan pada tempat penyimpanan (*storage*) sebagai tempat bermuaranya data dan informasi. Dengan banyaknya data yang tersimpan maka akan menimbulkan beberapa permasalahan. Hadirnya SAN membantu dalam permasalahan *storage*, tetapi permasalahan akan kehilangan data apabila terjadi kerusakan pada *harddisk* adalah yang terpenting. Karena itu berkembang pula teknologi RAID yang dapat mengatasi permasalahan data. Teknologi ini digunakan sebagai duplikasi data sehingga jika terjadi *harddisk* rusak namun data yang tersimpan didalamnya akan tetap dapat diakses oleh *user*. Pada penelitian ini, penulis mencoba membuat *virtual data storage* menggunakan Openfiler 2.3

dengan protokol iSCSI dan melakukan percobaan penyimpanan data menggunakan RAID pada VMware. Hasil dari penelitian ini adalah sebuah tempat penyimpanan data yang dapat mencegah terjadinya kehilangan data apabila terjadi suatu kerusakan pada *harddisknya*.

**Kata Kunci:** *Virtual Data Storage*, SAN (Storage Area Network), RAID (Redundant Array Of Independent Disks), iSCSI.

## PENDAHULUAN

Suatu perusahaan membutuhkan penambahan media penyimpanan seiring dengan bertambahnya data karyawan maupun transaksi yang dilakukan. Jika perusahaan memutuskan untuk tidak menambah media penyimpanan tersebut, perlu dilakukan pemilihan data mana saja yang akan tetap disimpan untuk mengurangi over kapasitas ruang penyimpanan. Pemilihan teknologi yang digunakan sebagai alternatif permasalahan penyimpanan data menjadi sangat penting karena selain penerapan yang tepat juga memperhitungkan efisiensi, kebutuhan dan fleksibilitas perkembangannya di masa depan.

Pada penelitian sebelumnya Reza (2015) menyatakan permasalahan overload dan high performance media penyimpanan dapat diatasi dengan media penyimpanan terpusat (Storage Area Network) yang dapat diterapkan dengan menggunakan jaringan Fiber Channel (FC) yang disebut dengan FC-SAN atau menggunakan Internet Protocol (IP) yang disebut IP-SAN. IP-SAN adalah sebuah jaringan yang dibangun dengan menggunakan protocol iSCSI yang dapat diterapkan dalam jaringan Local Area Network (LAN). Aameek Singh (2013) dalam penelitiannya menyebutkan metode Storage Virtualisation System dengan Load Balancing yang di pasang pada Storage Area Network (SAN) juga efektif mengatasi masalah kelebihan beban data pada server, switch, dan storage.

Dengan memperhitungkan kebutuhan dan kemampuan, pada dasarnya ruang penyimpanan data (*storage server*) dapat dibangun sendiri, melalui perhitungan yang tepat untuk kapasitas dan fleksibilitas yang sama, pengadaan perangkat keras bisa diabaikan dengan membangun sendiri. Pada perangkat keras (*hardware*), diperlukan keseimbangan antara kebutuhan, anggaran dan kehandalan mesin.

Dengan banyaknya produk *Open Source* yang dapat membantu mengatasi permasalahan penyimpanan data (*storage*) dan *high availability storage*, penulis mencoba mengimplementasikan media penyimpanan virtual dengan openfiler menggunakan protokol iSCSI sehingga penerapan tempat penyimpanan data yang dapat mencegah terjadinya overload serta kehilangan data apabila terjadi suatu kerusakan pada *harddisknya* dan meningkatkan performa dan *high availability* dari Storage Area Network (SAN) dapat diatasi dan dilakukan.

#### BAHAN DAN METODE

Menurut Wahyono (2007:106) Sebuah storage area network (SAN) adalah jenis jaringan area lokal (LAN) yang dirancang untuk menangani transfer data yang besar. Sebuah SAN biasanya mendukung penyimpanan data, pengambilan dan replikasi di jaringan dengan menggunakan server high-end, array beberapa disk dan teknologi interkoneksi Fibre Channel. Teknologi SAN serupa namun berbeda dari teknologi penyimpanan jaringan terpasang (NAS). Istilah SAN dapat merujuk ke jaringan area sistem bukan jaringan area penyimpanan.

Sebelum membangun SAN *server* terlebih dahulu dibuat sebuah rancangan awal agar dapat memudahkan proses pengerjaannya.

Perancangannya sebagai berikut:

1. Menentukan sistem operasi yang digunakan beserta *software* pendukung yang dibutuhkan dalam pelaksanaan skripsi ini, sistem operasi yang akan digunakan adalah Openfiler dan sistem operasi Windows yang akan digunakan pada setiap *workstationnya*.
2. Menentukan *hardware* yang dibutuhkan dalam pengimplementasian jaringan SAN.
3. Pengujian simulasi dilakukan dengan menggunakan 1 PC/Laptop, Openfiler akan diinstal melalui Vmware.

Sebuah *storage area network* memelihara salinan data digital melalui jaringan lokal (LAN) dengan koneksi berkecepatan tinggi. Hal ini dirancang untuk membuat file cadangan, database, dan data lainnya ke lokasi pusat yang dapat dengan mudah diakses melalui protokol jaringan standar dan alat (Mitchell, 2016).

Keuntungan SAN menurut Sugeng (2010:36)

yaitu :

1. *Availability*: Satu *copy* dari data jadi dapat diakses oleh semua *host* melalui jalur yang berbeda dan semua data lebih efisien *dimanage-nya*.
2. *Reliability*: Infrastruktur *transport* data yang dapat menjamin tingkat kesalahan yang sangat minimal, dan kemampuan dalam mengatasi kegagalan.
3. *Scalability*: *Server* maupun media penyimpanan (*storage*) dapat ditambahkan secara *independent* satu dan lainnya, dengan tanpa pembatas harus menggunakan sistem yang *proprietary*.
4. *Performance*: *Fibre Channel* (standar *enabling* teknologi untuk *interkoneksi* SAN) mempunyai *bandwidth* 100Mbps *bandwidth* dengan *overhead* yang rendah, dan SAN akan memisahkan trafik *backup* dengan trafik standar LAN/WAN.
5. *Manageability*: Berkembangnya perangkat lunak dan standar baik untuk FC-AL (*Fibre Channel Arbitrated Loop*) maupun *Fibre Channel fabric* memungkinkan manajemen dilakukan secara terpusat dan koreksi dan deteksi kesalahan yang proaktif.
6. *Return On Information Management*: Karena bertambahnya tingkat *redundansi* dan kemampuan manajemen yang baik, maupun kemampuan untuk di tambahkan *server* dan media penyimpan (*storage*) secara independen - SAN pada akhirnya memungkinkan biaya kepemilikan yang rendah pada saat yang sama menaikkan *Return On Information Management* (ROIM) di bandingkan metoda penyimpanan tradisional.

#### Analisa Penelitian

- a. Analisa Kebutuhan  
Menurut analisa dan tujuan implementasi, penulis membutuhkan dua buah PC *Highend* atau minimal spesifikasi :  
1) Intel Pentium i3  
2) RAM 8gb sebagai *initiator*  
Dan PC *lowend* dengan spesifikasi minimal:  
1) Intel pentium core2duo.  
2) Harddisk *Internal* 2 TB  
3) RAM 2 gb sebagai pc target yang berperan sebagai RAID
- b. Desain  
Desain *virtual data storage* yang akan diimplementasikan mengadopsi topologi jaringan *star* yang akan divisualisasikan menggunakan *software* cisco packet tracer dan microsoft visio.  
1) Menggunakan microsoft visio untuk mendesain topologi *virtual data storage* yang akan diimplementasikan.

- 2) Divisualisasi menggunakan cisco packet tracer.
  - 3) Menyediakan 2 buah PC, dan satu switch. 1 PC sebagai *initiator* iSCSI dan 1 PC untuk target iSCSI.
  - 4) Menghubungkan *virtual data storage/SAN virtual* melalui *Vmware*.
  - 5) Melakukan perbandingan kekurangan dan kelebihan tipe tingkatan RAID.
- c. Testing  
 Dalam skripsi ini penulis mencoba *testing* menggunakan lebih dari satu *operating system* yang berbeda dan mengkoneksikannya ke server *virtual data storage* menggunakan *virtual box*.
- d. Implementasi  
 Akan diimplementasikan pada ruang komputer di kantor camat jatinegara caranya dengan menggunakan 2 buah PC yang dikonfigurasi sebagai iSCSI *initiator* dan iSCSI target yang akan menjadi *server virtual data storage/SAN*.

Metode Pengumpulan Data

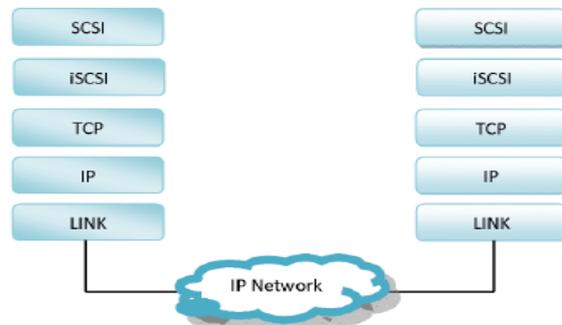
- a. Observasi  
 Dengan cara penelitian langsung ke objek penelitian, yaitu kantor camat Jatinegara.
- b. Wawancara  
 Melakukan wawancara kepada narasumber atau IT setempat untuk mendapatkan informasi dan data yang berkaitan dengan skripsi.
- c. Studi pustaka  
 Mencari literatur materi melalui artikel-artikel di *internet*, buku, dan jurnal yang berhubungan dengan materi yang akan dibahas

**HASIL DAN PEMBAHASAN**

Implementasi RAID membutuhkan minimal dua *harddisk* ketika RAID digunakan, sistem operasi akan membaca kedua *harddisk* sebagai satu *harddisk*. Jadi, meskipun ada dua *harddisk*, *drive* yang tampak pada *Windows Explorer* hanya satu C saja. Misalnya sebagai perbandingan, kalau RAID tidak digunakan, *drive* pada *Windows Explorer* muncul C dan D. Setiap *drive* untuk satu *harddisk*. Pada penelitian ini penulis mengimplementasikan *SAN storage* menggunakan protocol iSCSI pada kantor kecamatan Jatinegara.

iSCSI yang merupakan singkatan dari *Internet Small Computer System Interface*, bekerja di atas *Transport Control Protocol (TCP)* dan memungkinkan perintah SCSI untuk dikirim *end-to-end* melalui jaringan area lokal (LAN), jaringan yang luas (WAN ) atau Internet. IBM mengembangkan iSCSI sebagai bukti dari konsep

pada tahun 1998, dan disajikan draft pertama dari standar iSCSI ke *Internet Engineering Task Force (IETF)* pada tahun 2000. Protokol diratifikasi pada tahun 2003 (SearchStorage, 2014)

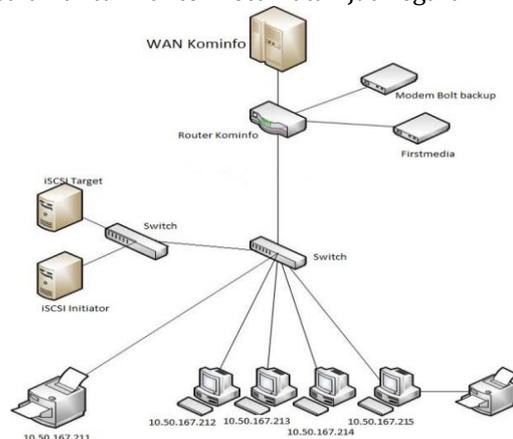


Sumber: Sabiq (2012)

**Gambar 1. Letak iSCSI**

Pada penelitian ini, penulis membatasi permasalahan untuk mengimplementasikan *SAN storage* menggunakan protocol iSCSI pada kantor camat Jatinegara.

SAN merupakan sistem media penyimpanan terpusat yang dapat diterapkan dengan menggunakan jaringan *Fiber Channel (FC)* yang disebut dengan *FC-SAN* ataupun menggunakan *Internet Protocol (IP)* yang disebut *IP-SAN*. *FCSAN* merupakan sistem SAN yang mempunyai performa sangat tinggi, tetapi *FC-SAN* membutuhkan biaya yang mahal dan bersifat kompleks. *IP-SAN* dikenal dengan sebuah jaringan yang dibangun dengan menggunakan iSCSI sebagai protokolnya (Sabiq, 2012). iSCSI merupakan protokol yang mengenkapsulasi perintah SCSI (*Small Computer System Interface*) dan data kemudian mengirimkannya melalui jaringan IP yang digunakan untuk mengirimkan data melalui *internet* ataupun *intranet* dan mengelola media penyimpanan dari jarak jauh melalui internet. Berikut adalah gambar topologi usulan untuk kantor kecamatan Jatinegara.

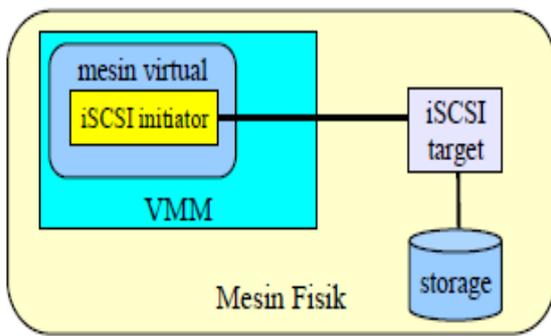


Sumber: Nawal & Anton (2017)

**Gambar 2. Topologi Usulan**

Penulis mengimplementasikan SAN dalam bentuk *virtual* menggunakan Vmware. Mesin *virtual* merupakan sebuah perangkat lunak yang dipasang pada *Virtual Machine Manager* (VMM) di suatu system operasi yang dapat menjalankan sistem operasi dan aplikasi sendiri seperti halnya komputer fisik. Mesin *virtual* mempunyai perilaku yang sama persis dengan mesin fisik, dan komponen dari mesin *virtual* tersebut terdiri dari CPU, RAM, *Harddisk* dan *Network Interface Card* (NIC) *virtual* yang sebenarnya adalah perangkat lunak.

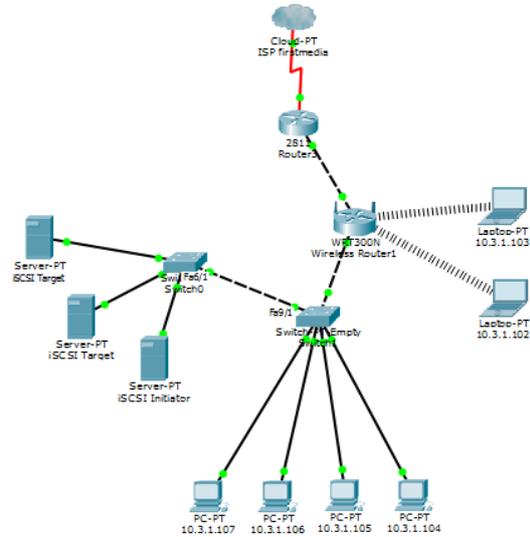
Sistem iSCSI yang akan dibangun terdiri dari dua buah PC yang berperan sebagai iSCSI *target* dan iSCSI *initiator*. PC untuk iSCSI *initiator* merupakan mesin *virtual* yang berjalan di atas *Virtual Machine Monitor* yang berada pada PC untuk iSCSI *target* seperti yang ditunjukkan pada Gambar 3. Kedua mesin ini juga dihubungkan secara *virtual* melalui jaringan menggunakan *Gigabit Ethernet Local Area Network* (GELAN).



Sumber: Sabiq (2012)  
**Gambar 3. Penerapan iSCSI**

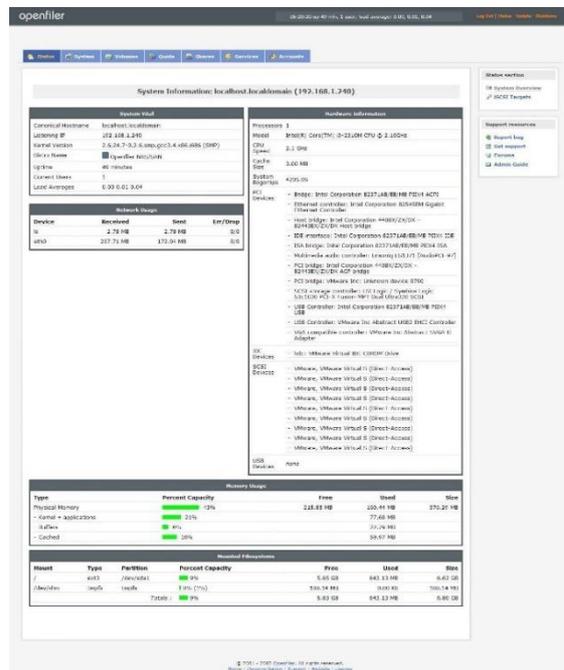
Usulan perubahan topologi jaringan, dengan diimplementasikannya SAN (*Storage area network*) akan mengakibatkan perubahan infrastruktur teknologi informasi, penerapan dan optimalisasi keamanan jaringan seperti melakukan monitoring pada komponen atau elemen-elemen jaringan pada kantor camat Jatinegara. Juga untuk penanganan masalah usulan untuk mengimplemetasikan solusi *disaster recovery system* dan peningkatan (upgrade) serta persiapan dalam pengembangan jaringan.

Berikut adalah skema jaringan usulan menggunakan cisco packet tracer.



Sumber: Nawal & Anton (2017)  
**Gambar 4. Skema menggunakan Packet Tracer**

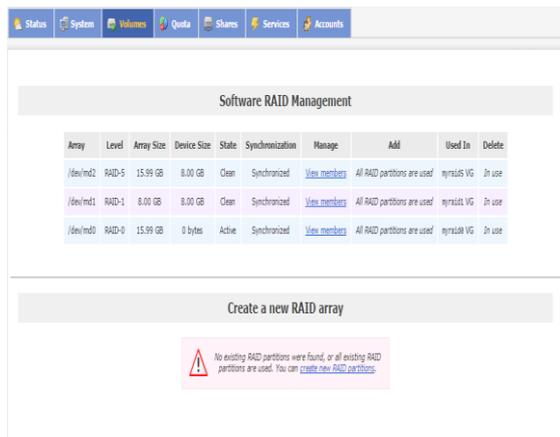
Implementasi menggunakan openfiler. Openfiler telah menyediakan antarmuka berbasis *web* (*openfiler web administration tool*) untuk administrasi sistem openfiler. *Interface* berbasis *web* ini memberikan banyak kemudahan dalam pengoperasian dan administrasi sistem openfiler.



Sumber: Nawal & Anton (2017)  
**Gambar 5. Konfigurasi Openfiler**

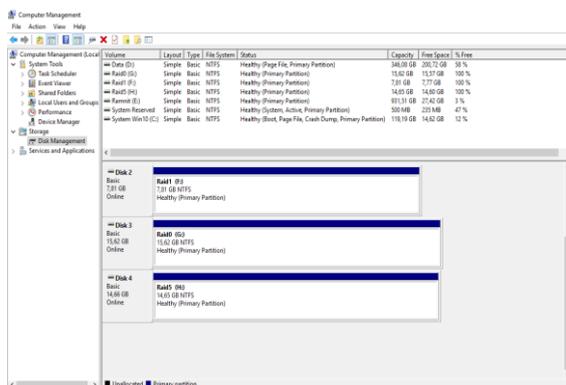
Openfiler mendukung tingkat RAID 0, 1, 5, 6 dan 10. Setiap tingkatan RAID memiliki syarat tersendiri dalam pembuatannya dan setiap tingkatan RAID memiliki sifat tersendiri dalam mengelola *harddisk*. Dalam penulisan ini penulis

membuat tiga tingkatan RAID untuk menguji fitur dari masing-masing tingkatan RAID



Sumber: Nawal & Anton (2017)  
**Gambar 6. RAID di openfiler**

Pada gambar 6, dapat dilihat bahwa RAID sudah berhasil dibuat dan terlihat seperti harddisk lokal

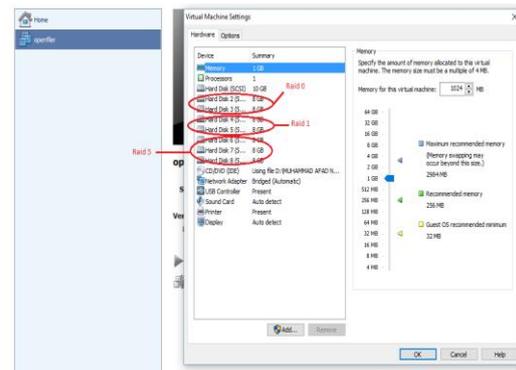


Sumber: Nawal & Anton (2017)  
**Gambar 7. RAID seperti lokal disk**

Komputer *client* yang sudah terkoneksi dengan openfiler pada sistem operasi *virtual* dapat digunakan sebagai *disk* lokal.

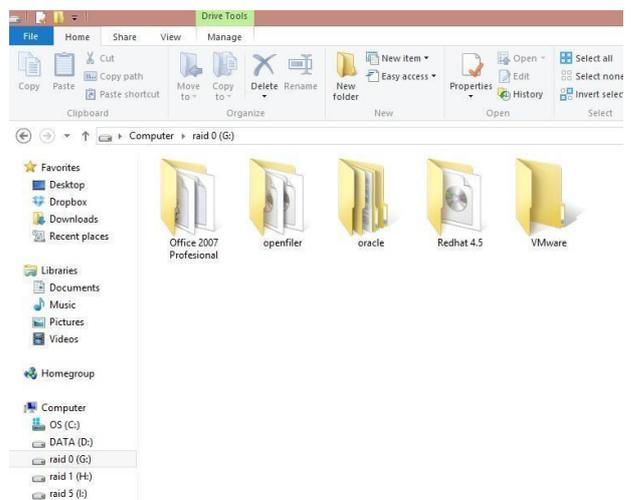
### 3. Hasil dan Pembahasan

Dalam tahap ini penulis melakukan pengujian penghapusan terhadap salah satu *harddisk* pada tipe-tipe RAID sesuai dengan fiturnya.



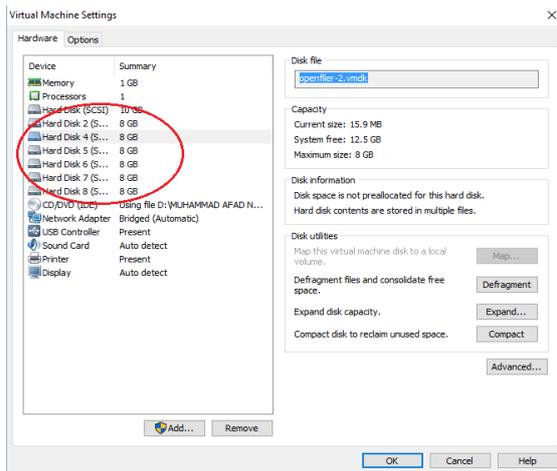
Sumber: Nawal & Anton (2017)  
**Gambar 8. Harddisk virtual untuk RAID**

Pengujian pertama penulis menguji RAID 0, RAID 0 adalah RAID yang menggabungkan kapasitas dari beberapa *harddisk* sehingga secara logikal hanya “terlihat” sebuah *harddisk* dengan kapasitas yang besar (jumlah kapasitas keseluruhan *harddisk*). RAID 0 tidak melakukan duplikasi, secara logikal apabila tidak memiliki duplikasi jika salah satu *harddisk* yang ada dalam RAID 0 penulis hapus maka data yang ada di dalamnya pun akan hilang dan tidak dapat dibaca.



Sumber: Nawal & Anton (2017)  
**Gambar 9. Isi RAID 0**

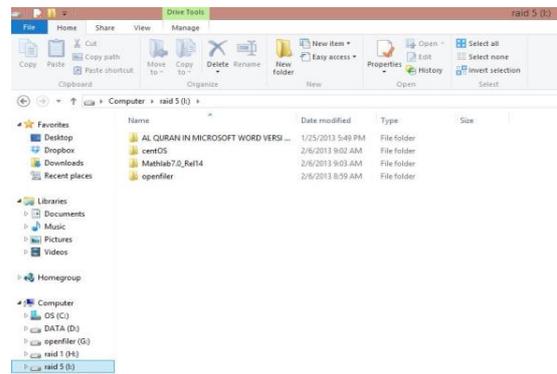
Dapat terlihat pada gambar IV.21, bahwa penulis telah menghilangkan 1 *harddisk* (*harddisk* SCSI 3) sebagai uji coba untuk RAID 0.



Sumber: Nawal & Anton (2017)

**Gambar 10. Harddisk SCSI 3 dihapus**

Penulis lalu melakukan pengecekan pada *drive* RAID 0 untuk melihat apakah setelah melakukan penghapusan 1 *harddisk* yang digunakan pada RAID 0, data yang tersimpan di dalam *drive* RAID 0 akan tetap dapat diakses. Hasil yang didapat adalah *drive* RAID 0 tidak ada lagi dalam kumpulan *drive* sehingga data yang tersimpan didalamnya tidak mungkin lagi dapat diakses dapat terlihat pada gambar 10.



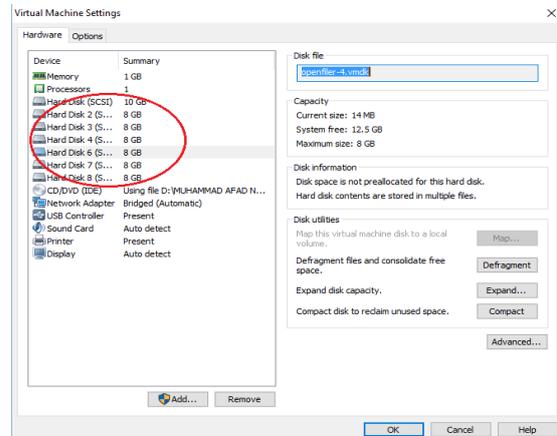
Sumber: Nawal & Anton (2017)

**Gambar 11. Drive RAID 0 tidak ada**

Untuk pengujian yang kedua, penulis menguji RAID 1 dimana RAID 1 adalah RAID yang menyalin isi sebuah *harddisk* ke *harddisk* lain dengan tujuan jika salah satu *harddisk* rusak secara fisik, maka data tetap dapat diakses dari *harddisk* lainnya. Keuntungan dari menggunakan RAID 1 adalah karena duplikasinya sehingga tidak perlu takut data akan hilang apabila hanya 1 *harddisk* yang rusak atau dihapus.

Untuk pengujian yang kedua, penulis menguji RAID 1 dimana RAID 1 adalah RAID yang menyalin isi sebuah *harddisk* ke *harddisk* lain dengan tujuan jika salah satu *harddisk* rusak secara fisik, maka data tetap dapat diakses dari *harddisk* lainnya. Keuntungan dari menggunakan

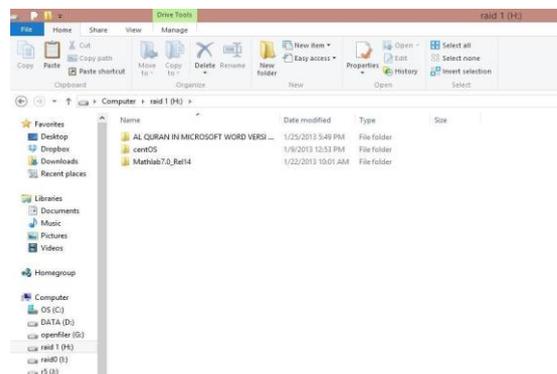
RAID 1 adalah karena duplikasinya sehingga tidak perlu takut data akan hilang apabila hanya 1 *harddisk* yang rusak atau dihapus.



Sumber: Nawal & Anton (2017)

**Gambar 12. Harddisk SCSI 5 telah dihapus**

Dapat terlihat pada gambar 11, bahwa penulis telah menghilangkan 1 *harddisk* (*harddisk* SCSI 5) sebagai uji coba untuk RAID 1 Penulis lalu melakukan pengecekan pada *drive* RAID 1 untuk melihat apakah setelah melakukan penghapusan 1 *harddisk* yang digunakan pada RAID 1, data yang tersimpan di dalam *drive* RAID 1 akan tetap dapat diakses. Hasil yang didapat adalah data yang tersimpan pada *drive* RAID 1 tetap dapat diakses dan tidak ada yang berubah sama seperti pada gambar 12

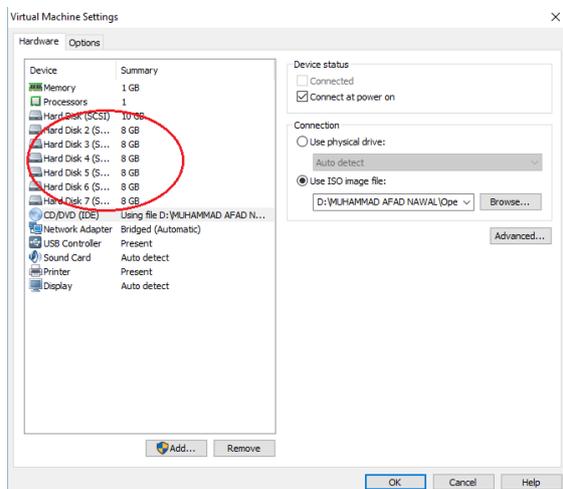


Sumber: Nawal & Anton (2017)

**Gambar 13. Isi *driver* RAID 1**

Untuk pengujian terakhir penulis menguji RAID 5 dimana RAID 5 itu adalah RAID yang dapat mentoleransi kegagalan satu atau lebih *disk* tanpa kehilangan data. Mengingat tiga *disk* A, B, C, dan tiga blok data yang akan digunakan sebagai penyimpanan. Ada dua kelemahan dari menggunakan tingkat RAID 5 yaitu selama sinkronisasi (membangun kembali *array* RAID), *read-write* kinerja *array* sangat berkurang karena *overhead* perhitungan. Jika ada *disk* lain gagal selama sinkronisasi, seluruh *array*

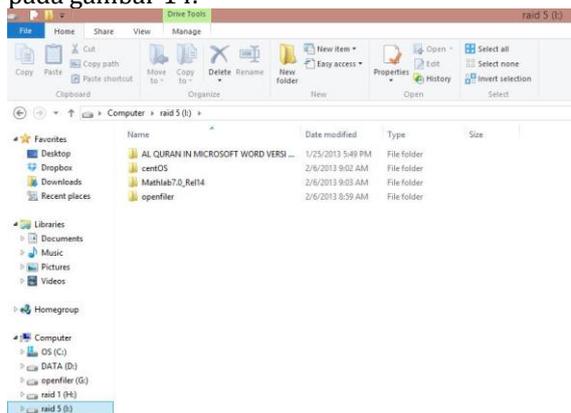
rusak. Mengingat bahwa sifat RAID 5 adalah tidak menggunakan *harddisk* khusus untuk menyimpan paritinya, namun paritinya tersebut disebar ke seluruh *harddisk* maka untuk total *capacity* penyimpanan adalah *size* 2 *harddisk* yang digunakan pada RAID 5.



Sumber: Nawal & Anton (2017)

**Gambar 13. Harddisk SCSI 8 telah dihapus**

Dapat terlihat pada gambar 13, bahwa penulis telah menghilangkan 1 *harddisk* (*harddisk* SCSI 8) sebagai uji coba untuk RAID 5. Penulis lalu melakukan pengecekan pada *drive* RAID 5 untuk melihat apakah setelah melakukan penghapusan 1 *harddisk* yang digunakan pada RAID 5, data yang tersimpan di dalam *drive* RAID 5 akan tetap dapat diakses. Hasil yang didapat adalah data yang tersimpan pada *drive* RAID 5 tetap dapat diakses dan tidak ada yang berubah sama seperti pada gambar 14.



Sumber: Nawal & Anton (2017)

**Gambar 14. Isi drive RAID 5**

Tabel 1. Tabel perbandingan antara tingkatan RAID

Perbandingan Antara Tingkatan RAID			
	RAID 0	RAID 1	RAID 5
Kelebihan	Kapasitas <i>harddisk</i> yang dimiliki untuk penyimpanan data adalah keseluruhan <i>harddisk</i> yang dimiliki, tanpa ada pengurangan	<i>Performance</i> lebih baik dari RAID 0, memiliki duplikasi sehingga jika <i>harddisk</i> mengalami kerusakan data akan tetap dapat diakses	Apabila terjadi <i>harddisk fails</i> maka data akan tetap dapat diakses, efisien dalam penggunaan ruang dan menyediakan redundansi data
Kekurangan	Jika salah satu <i>harddisk fails</i> dalam RAID 0, maka data akan hilang tanpa ada penggantinya	Membutuhkan <i>size harddisk</i> yang besar sehingga jika <i>harddisk</i> mengalami kerusakan data akan tetap dapat diakses	Jika <i>harddisk fails</i> , <i>performance</i> menurun

Sumber: Nawal & Anton (2017)

**KESIMPULAN**

SAN dapat menjadi salah satu alternatif dalam menghadapi permasalahan *storage*. SAN dapat menjamin duplikasi data, sehingga bila terjadi kerusakan pada salah satu *harddisk* data tetap dapat diakses. Dari hasil pengujian beberapa tingkatan RAID menunjukkan bahwa setiap tingkatan RAID memiliki kelebihan dan kekurangan masing-masing. iSCSI dapat dijadikan alternatif untuk SAN karena lebih murah biaya.

**REFERENSI**

Aameek Singh, Madhukar Korupolu, & Dushmanta Mohapatra. (2013). Server Storage Virtualization: Integration and Load Balancing in Data Centers. IBM Research Center

- Cisco Systems. 2017. iSCSI Protocol Concepts and Implementation. Diambil dari: [http://www.diskdrive.com/iSCSI/reading-room/white-papers/Cisco\\_iSCSI\\_Protocol\\_Concepts.pdf](http://www.diskdrive.com/iSCSI/reading-room/white-papers/Cisco_iSCSI_Protocol_Concepts.pdf)
- Jon Tate, Pall Beck, Hector Hugo Ibarra, Shanmuganathan Kumaravel dan Libor Miklas. 2016. Introduction to Storage Area Networks. ISBN 0738441430. United States: Summary of Changes for SG24-5470-06 for Introduction to Storage Area Networks as created or updated on January 14, 2016
- Mitchell, Bradley. 2016. What is Computer Network Storage?. Diambil dari: <https://www.lifewire.com/what-is-computer-network-storage-818004>. (13 Desember 2016)
- Nawal, M., A., & Anton. Laporan Akhir Penelitian Mandiri. Tangerang: AMIK BSI Tangerang
- Openfiler, 2015. Openfiler. Diambil dari <https://www.openfiler.com/products>. (14 Desember 2016)
- Reza Yusuf Merdekantara &, Elkaf Rahmawan Pramudya. 2015. Implementasi dan Analisis Clustering Pada NFS Menggunakan Storage Area Network (San) dengan Perbandingan Average I/O Performance.
- Sabiq, Ahmad & Setiadi, Yazid. 2012. Analisa Kinerja Protokol iSCSI Melalui Jaringan Virtual. Jurnal TET, 1(3), 166-173
- SearchStorage, 2014. Definition iSCSI (Internet Small Computer System Interface). Diambil dari <http://searchstorage.techtarget.com/definition/iSCSI>. (14 Desember 2016)
- Wahyono, Teguh. 2007. Building dan Maintenance PC Server. Jakarta: Elex Media Komputindo.