BAB 3
Metodologi Penelitian

3.1 Studi Pustaka

Studi pustaka dilakukan untuk mendapatkan informasi mengenai topik penelitian yang dapat bersumber dari dokumen, buku, artikel, atau bahan tertulis lainya yang berupa teori, laporan penelitian, atau penemuan sebelumnya, baik bersifat *online* maupun *offline source* yang bertujuan memberikan informasi.

Topik yang digali dalam langkah studi pustaka dalam penelitian ini adalah topik yang berkaitan dengan penyimpanan bukti digital, *storage cluster*, *software defined storage*, instalasi dan konfigurasi Ceph, peningkatan keamanan informasi sistem *storage* serta informasi lain yang relevan dengan topik penelitian.

3.2 Perancangan

Merupakan tahap perancangan *digital evidence storage* yang akan dibangun menggunakan *software defined storage*: Ceph *storage cluster*. Tahap perancangan ini meliputi proses analisa kebutuhan *hardware* dan *software* serta pembuatan topologi rancangan sistem *storage* yang akan dibangun sehingga dapat memenuhi kriteria yang telah ditentukan sebelumnya yaitu, *scalability*, *confidentiality*, *integrity*, dan *availability*.

Dalam implementasi penelitian ini *digital evidence storage* dibangun dalam lingkungan *virtual machine* (VM). Meskipun dibangun dalam lingkungan virtual namun sudah dapat memberikan gambaran sistem *storage* sebenarnya. *Virtual machine* yang digunakan adalah Proxmox VE 5.3-8. Proxmox VE merupakan salah satu *virtual machine* (VM) yang populer dengan fitur yang cukup lengkap dibanding *software* virtual komputer yang lain, tipe virtualisasi server yang *free* dan mudah dalam instalasi. Selain itu, beberapa keuntungan menggunakan Proxmox VE yaitu, kinerja terbaik, instalasi yang telah dioptimalkan, sehingga lebih cepat, mudah dalam manajemen dan cocok untuk kelas enterprise (Onno W. Purbo, 2012:37).

Software defined storage (SDS) yang digunakan dalam penelitian ini adalah Ceph storage cluster, dalam hal ini digunakan Ceph Mimic Stable Version 13.2.5. Ceph merupakan SDS *open source*, berjalan pada perangkat keras komoditas apa pun, sehingga tidak ada vendor yang terkunci, dan menyediakan biaya per GB yang rendah. Ceph dapat

mengurus semua kebutuhan user dengan kriteria seperti *low cost*, reliabilitas, dan skalabilitas yang menjadi ciri utama Ceph (Singh, 2015).

Informasi yang dihimpun dari website Ceph ("Ceph Ceph storage - Ceph," n.d.) menyebutkan bahwa Ceph memiliki kemampuan melakukan transformasi infrastruktur IT dan memampukan kita melakukan manajemen data yang besar. Landasan dari Ceph adalah RADOS yang merupakan kependekan dari *Reliable Autonomic Distributed Object Storage* yang menyediakan *storage* untuk objek, block dan file system dalam sebuah *cluster* membuat Ceph fleksibel, dapat diandalkan dan mudah untuk dikelola (S. A. Weil et al., 2006).

RADOS memungkinkan Ceph untuk dapat *self-healed* dan *self-manage*. Jika terjadi bencana, Ceph dapat memberikan reliabilitas terhadap *multiple failure*. Ceph mendeteksi dan memperbaiki kegagalan di setiap zona kegagalan sebagai disk, node, jaringan, rak, baris pusat data, pusat data, dan bahkan berbagai lokasi geografi. Ceph mencoba untuk mengelola situasi secara otomatis dan mengatasinya sebisa mungkin tanpa gangguan data (D'Atri, Bhembre, & Singh, 2017).

Untuk dapat memahami Ceph, kita perlu mempelajari CRUSH sebagai landasan algoritma Ceph. CRUSH (*Controlled Replication Under Scalable Hashing*) sendiri adalah algoritma distribusi data *pseudo-random* yang secara efisien dan kuat mendistribusikan replika objek di seluruh cluster penyimpanan yang heterogen dan terstruktur. CRUSH hanya membutuhkan deskripsi hierarkis yang ringkas tentang perangkat yang terdiri dari kluster penyimpanan dan pengetahuan tentang kebijakan penempatan replika. Pendekatan ini memiliki dua keunggulan utama: pertama, sepenuhnya didistribusikan sehingga setiap pihak dalam sistem yang besar dapat secara bebas menghitung lokasi objek apa pun; dan kedua, apa yang dibutuhkan oleh metadata kecil kebanyakan statis, hanya berubah ketika perangkat ditambahkan atau dihapus (S. Weil, Brandt, Miller, & Maltzahn, 2007).

Ceph dikembangkan dengan tujuan utama adalah skalabilitas, kinerja dan reliabilitas yang tinggi. Ceph secara langsung menangani masalah skalabilitas dan sekaligus mencapai performa yang tinggi, reliabilitas, dan availabilitas melalui tiga fitur desain mendasar: data dan metadata terpisah (*decoupled data and metadata*), manajemen metadata terdistribusi dinamis (*Dinamic distributed metadata management*), dan penyimpanan objek terdistribusi otonom yang andal (*Reliable Autonomic Distributed Object Storage*/ RADOS) (S. A. Weil et al., 2006).

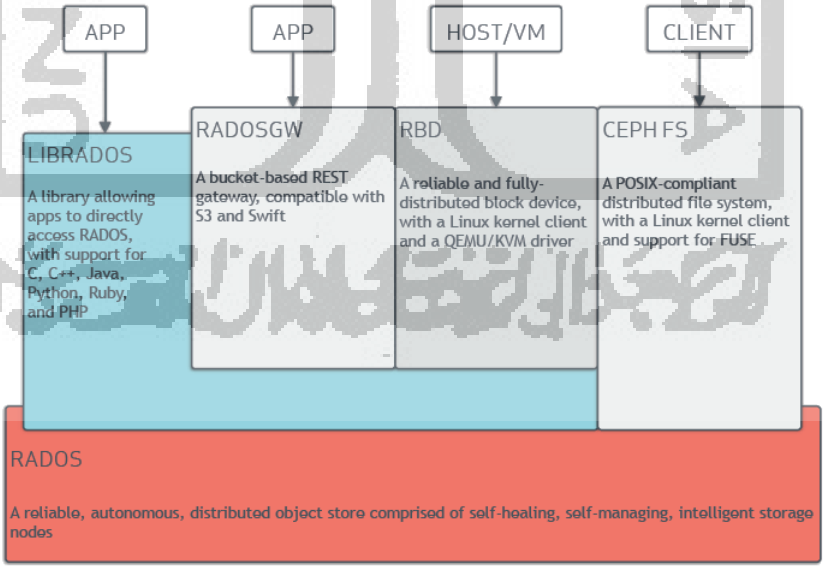
- *decoupled data and metadata* -- Ceph memaksimalkan pemisahan manajemen sistem file metadata dari penyimpanan data file. Ceph menghilangkan daftar alokasi sepenuhnya. Sebagai gantinya, fungsi sederhana digunakan untuk memberi nama objek yang berisi data file berdasarkan nomor node, rentang byte, dan strategi *striping*, sementara fungsi distribusi data tujuan khusus menetapkan objek ke perangkat penyimpanan tertentu. Hal ini memungkinkan pihak mana pun untuk menghitung (alih-alih mencari) nama dan lokasi objek yang terdiri dari konten file, menghilangkan kebutuhan untuk memelihara dan mendistribusikan daftar objek, menyederhanakan desain sistem, dan mengurangi beban kerja *cluster* metadata.
- Manajemen Metadata Terdistribusi Dinamis -- Karena operasi metadata file system membentuk sebanyak setengah dari beban kerja sistem file tipikal, manajemen metadata yang efektif sangat penting untuk kinerja sistem secara keseluruhan. Ceph menggunakan arsitektur *cluster* metadata baru berdasarkan *Dynamic Subtree Partition* yang secara adaptif dan cerdas mendistribusikan tanggung jawab untuk mengelola hierarki direktori sistem file di antara puluhan atau bahkan ratusan Metadata Server (MDS). Partisi hierarkis (dinamis) menjaga lokalitas di setiap beban kerja MDS, memfasilitasi pembaruan yang efisien dan pengambilan awal yang agresif untuk meningkatkan kinerja untuk beban kerja umum. Secara signifikan, distribusi beban kerja di antara server metadata didasarkan sepenuhnya pada pola akses saat ini, memungkinkan Ceph untuk secara efektif memanfaatkan sumber daya MDS yang tersedia di bawah beban kerja apa pun dan mencapai penskalaan hampir *linear* dalam jumlah MSDs.
- *Reliable Autonomic Distributed Object Storage* (RADOS) -- Sistem besar yang terdiri dari ribuan perangkat secara inheren dinamis: mereka dibangun secara bertahap, mereka tumbuh dan berkontraksi ketika penyimpanan baru dikerahkan dan perangkat lama dinonaktifkan, kegagalan perangkat sering dan diharapkan, dan volume besar data dibuat, dipindahkan, dan dihapus. Semua faktor ini mensyaratkan bahwa distribusi data berevolusi untuk secara efektif memanfaatkan sumber daya yang tersedia dan mempertahankan tingkat replikasi data yang diinginkan. Ceph mendelegasikan tanggung jawab untuk migrasi data, replikasi, deteksi kegagalan, dan pemulihan kegagalan ke gugus OSD yang menyimpan data, sementara pada tingkat tinggi, OSD secara kolektif menyediakan penyimpanan objek logis tunggal untuk klien dan server metadata. Pendekatan ini memungkinkan Ceph untuk lebih efektif memanfaatkan

kecerdasan (CPU dan memori) yang ada pada setiap OSD untuk mencapai penyimpanan objek yang andal dan tersedia dengan skala linear.

Ceph *storage cluster* adalah dasar untuk semua penyebaran Ceph. Ceph *storage cluster* terdiri dari dua jenis daemon: Ceph OSD Daemon (OSD) menyimpan data sebagai objek pada node penyimpanan; dan Ceph Monitor (MON) menyimpan salinan master peta *cluster*. Ceph *storage cluster* dapat berisi ribuan node penyimpanan. Sistem setidaknya memiliki satu Ceph Monitor dan dua Ceph OSD Daemon untuk replikasi data.

Ceph *storage system* terpadu mendukung penyimpanan objek, block dan file melalui Ceph Object Storage (RADOSGW), Ceph block devices (RBD) dan Ceph Filesystem (CephFS). Diagram arsitektur Ceph *storage system* dapat dilihat dalam gambar 2.2.

- **Ceph object storage** dapat diakses melalui Amazon S3 (*Simple Storage Service*) dan OpenStack Swift REST (*Representational State Transfer*).
- **Ceph block devices** dapat dijadikan virtual disk pada Linux maupun virtual machine. Teknologi RADOS yang dimiliki Ceph memampukan Ceph melakukan snapshot dan replikasi.
- **Ceph Filesystem** (CephFS) menyediakan sistem file yang sesuai dengan POSIX (*Portable Operating System Interface for UNIX*) yang dapat digunakan dengan *mount* atau sebagai FUSE (*filesystem in user space*).



Gambar 3.1 Arsitektur Ceph

Implementasi CephFS memerlukan satu *metadata server* (MDS) yang akan dikonfigurasi pada salah satu dari node *cluster*. *Cluster* MDS dapat memperluas dan dapat menyeimbangkan *filesystem* secara dinamis untuk mendistribusikan data secara merata di antara host *cluster*. Ini memastikan kinerja tinggi dan mencegah beban berat pada host tertentu di dalam *cluster*.

CephFS dirancang untuk digunakan pada perangkat *server*, dan tidak dimaksudkan untuk dipasang pada desktop *user*. *User* dapat menggunakan driver kernel sistem operasi untuk memasang sistem *file* CephFS seperti yang dilakukan perangkat lokal atau sistem *file* jaringan Gluster. Ada juga opsi driver FUSE *user space*. Setiap instalasi harus mempertimbangkan dua metode pemasangan: FUSE lebih mudah diperbarui, tetapi *driver* kernel asli dapat memberikan kinerja yang lebih baik secara terukur (D'Atri et al., 2017).

Berikut adalah keuntungan penggunaan CephFS sebagaimana yang tertulis dalam website Ceph ("Ceph File system - Ceph," n.d.):

- Memberikan keamanan data yang lebih kuat.
- Menyediakan penyimpanan *filesystem* yang hampir tidak terbatas.
- Aplikasi yang menggunakan *filesystem* dapat menggunakan CephFS. Tidak diperlukan integrasi atau penyesuaian.
- Ceph secara otomatis menyeimbangkan *filesystem* untuk memberikan kinerja maksimum.
- Ceph secara unik mengirimkan objek, blok, dan penyimpanan file dalam satu sistem terpadu

Untuk memudahkan manajemen layanan dan sinkronisasi file barang bukti digital antara file hasil akuisisi dan Ceph *client* digunakan software ownCloud. OwnCloud dapat mengatur layanan *cloud storage* mulai dari *user*, *group user*, *limited* akses, *size quota user*, fitur sinkronisasi otomatis, melakukan *transfer* data menggunakan enkripsi SSL dan melakukan verifikasi *checksum* terhadap file yang ter-*upload* atau *download*. Owncloud bekerja dengan melakukan sinkronisasi file secara *bidirectional* antara komputer PC dan server. Sinkronisasi dilakukan menggunakan csync, alat sinkronisasi file dua arah yang menyediakan klien baris perintah dan juga pustaka. Modul khusus untuk csync ditulis untuk menyinkronkan dengan server WebDAV bawaan Cloud. OwnCloud juga memungkinkan untuk melakukan kontrol penuh atas lokasi data dan transfer, sambil menyembunyikan infrastruktur penyimpanan yang mendasarinya, yang dapat terdiri dari beberapa sumber daya penyimpanan. Berikut adalah fitur yang dimiliki oleh ownCloud:

1. Dapat menyimpan file, folder, kontak, galeri foto dan lain-lain pada server dan dapat diakses melalui ponsel, desktop maupun web browser.
2. Dapat melakukan sinkronisasi data dari PC, laptop, dan smartphone ke komputer server.
3. Berbagi data dengan orang lain atau umum sesuai dengan kebutuhan termasuk kemampuan dalam pengunaan URL publik.
4. Memiliki user interface yang memungkinkan untuk memanjemen, mengupload, download, sharing file dan folder dengan cara yang sangat mudah.
5. Memiliki fitur khusus bagi pengguna untuk membatalkan penghapusan data yang secara tidak sengaja dihapus.
6. Memiliki fitur pencarian yang responsif yang memungkinkan pencarian data dilakukan berdasarkan nama serta jenis file.
7. Kontak dapat diatur dalam kelompok sehingga dapat mengakses kontak berdasarkan rekan kerja dan lain-lain.
8. Dapat dikembalikan pada file sebelumnya setelah file dimodifikasi.
9. Support autentikasi menggunakan LDAP.
10. Pengeditan teks online.
11. *Viewer* untuk berbagai format file serta
12. dukungan untuk layanan penyimpanan Cloud eksternal (mis. Dropbox atau Google Drive).

Fitur utama yang membuat penulis memilih OwnCloud adalah kemampuan ownCloud untuk melakukan sinkronisasi file antara berbagai sistem operasi dan penerapan fitur checksum yang memeriksa integritas file saat mengunggah dan mengunduh dengan mengkomputasi checksum setelah transfer file selesai.

Selama pengunggahan file, server mengkomputasi checksum SHA1, MD5, dan Adler-32 dan membandingkan salah satunya dengan checksum yang disediakan oleh komputer klien. Jika terjadi ketidakcocokan, server akan mengembalikan kode Status HTTP 400 (*Bad Request*) sehingga memberi sinyal kepada komputer klien bahwa unggahan gagal. Server kemudian membuang unggahan, dan komputer klien mem-*blacklist* file tersebut. Selanjutnya, komputer klien akan mencoba kembali mengunggah file menggunakan *back-off* eksponensial. Jika berhasil (mencocokkan checksum) komputasi checksum akan disimpan oleh server di `oc_filecache` di samping file.

Jika ada unggahan terpotong, Checksum dari file lengkap dikirim dengan setiap potongan file. Tetapi server hanya membandingkan checksum setelah menerima checksum yang dikirim dengan chunk terakhir. Saat mengunduh, server mengirimkan checksum dalam header HTTP dengan file. (format yang sama seperti di atas). Jika tidak ada checksum yang ditemukan di `oc_filecache` (penyimpanan eksternal yang baru dipasang) itu dikomputasi dan disimpan dalam `oc_filecache` pada unduhan pertama. Checksum kemudian diberikan pada semua unduhan berikutnya tetapi tidak pada unduhan pertama. ("Appendix B: History and Architecture — ownCloud Client Manual documentation," n.d.)

OwnCloud dan Ceph merupakan alat dan sistem untuk manajemen data terdistribusi. OwnCloud dalam hal ini berfungsi sebagai layanan sinkronisasi file, sedangkan Ceph adalah sebagai sistem penyimpanan datanya.

Selanjutnya, untuk mengakomodasi agar sistem *storage* dapat diakses dari lokasi yang jauh menggunakan jaringan internet, sistem *storage* dihubungkan menggunakan jaringan *Virtual Private Network* (VPN). VPN merupakan suatu jaringan *private* yang mempergunakan sarana jaringan komunikasi publik yaitu internet dengan memakai *tunneling protocol* sebagai prosedur pengamanannya. VPN merupakan suatu jaringan komunikasi lokal yang terhubung melalui media jaringan publik, infrastruktur publik yang paling banyak digunakan adalah jaringan internet. Didalam VPN terdapat perpaduan teknologi tunneling dan enkripsi yang membuat VPN menjadi teknologi yang handal untuk mengatasi permasalahan keamanan didalam jaringan.

Afrianto & Setiawan (2014) menyebutkan teknologi VPN menyediakan beberapa fungsi utama untuk penggunaannya. Fungsi-fungsi utama tersebut antara lain sebagai berikut.

1. *Confidentially* (Kerahasiaan)

   Dengan digunakannnya jaringan publik yang rawan pencurian data, maka teknologi VPN menggunakan sistem kerja dengan cara mengenkripsi semua data yang lewat melauinya. Dengan adanya teknologi enkripsi tersebut, maka kerahasiaan data dapat lebih terjaga. Walaupun ada pihak yang dapat menyadap data yang melewati internet bahkan jalur VPN itu sendiri, namun belum tentu dapat membaca data tersebut, karena data tersebut telah teracak. Dengan menerapkan sistem enkripsi ini, tidak ada satupun orang yang dapat mengakses dan membaca isi jaringan data dengan mudah.

2. *Data Integrity* (Keutuhan data)

   Ketika melewati jaringan internet, sebenarnya data telah berjalan sangat jauh melintasi berbagai negara. Pada saat perjalanan tersebut, berbagai gangguan dapat terjadi terhadap isinya, baik hilang, rusak, ataupun dimanipulasi oleh orang yang tidak seharusnya. Pada VPN terdapat teknologi yang dapat menjaga keutuhan data mulai dari data dikirim hingga data sampai di tempat tujuan.

3. *Origin Authentication* (Autentikasi sumber)

   Teknologi VPN memiliki kemampuan untuk melakukan autentikasi terhadap sumber-sumber pengirim data yang akan diterimanya. VPN akan melakukan pemeriksaan terhadap semua data yang masuk dan mengambil informasi dari sumber datanya. Kemudian, alamat sumber data tersebut akan disetujui apabila proses autentikasinya berhasil. Dengan demikian, VPN menjamin semua data yang dikirim dan diterima berasal dari sumber yang seharusnya. Tidak ada data yang dipalsukan atau dikirim oleh pihakpihak lain.

4. *Non-repudiation*

   Yaitu mencegah dua pihak dari menyangkal bahwa mereka telah mengirim atau menerima sebuah file mengakomodasi perubahan.

5. Kendali akses

   Menentukan siapa yang diberikan akses ke sebuah sistem atau jaringan, sebagaimana informasi apa dan seberapa banyak seseorang dapat menerima.

Dalam penelitian ini dibangun 2 jalur koneksi untuk melakukan akses ke dalam sistem yaitu melalui jaringan kabel dan jaringan nirkabel (wireless maupun GSM). VPN yang digunakan dalam penelitian ini adalah openVPN. Selain agar dapat diakses menggunakan jaringan internet, konfigurasi OpenVPN juga digunakan sebagai upaya pengamanan transmisi data. Pemilihan OpenVPN sebagai infrastruktur jalur koneksi dikarenakan OpenVPN berbasis teknologi *open source* seperti OpenSSL encryption library dan protokol SSL V3/TLS V1 serta memiliki tingkat keamanan yang tinggi. OpenVPN memiliki keunggulan yaitu cepat, fleksibel, dan aman. Tidak peduli sistem atau platform operasi yang digunakan, sistem akan terlindungi dengan baik. Perlindungan keamanan tersebut dapat terjamin karena OpenVPN menggunakan *library* OpenSSL untuk melakukan enkripsi. OpenSSL mendukung beberapa algoritma kriptografi yang berbeda seperti 3DES, AES, RC5, Blowfish. Seperti IPSec, VPN menerapkan algoritma AES yang sangat aman dengan kunci 256 bit. Dalam dunia enkripsi, AES merupakan teknologi terbaru dan

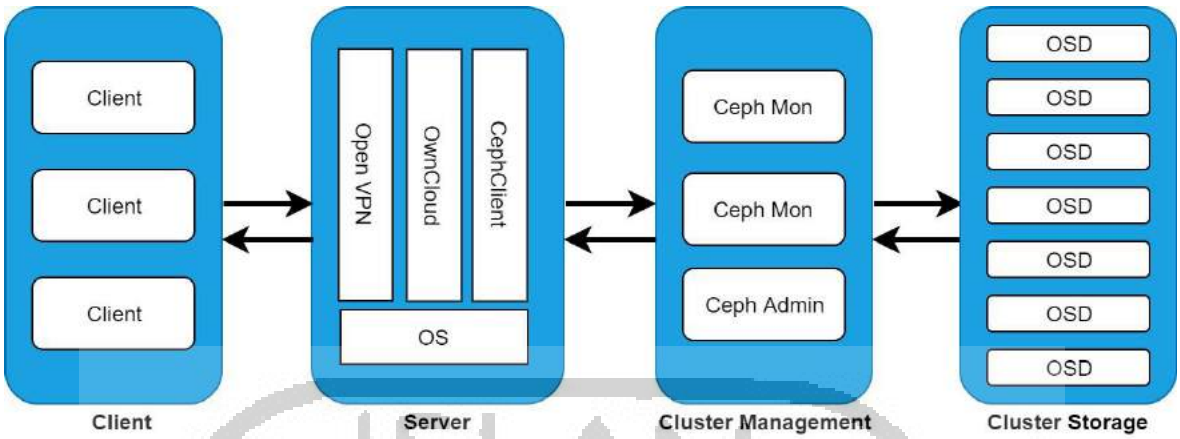
dipertimbangkan sebagai ‘standar emas’ karena teknologi ini belum diketahui kelemahannya.

Tabel 3.1 Perbandingan openVPN, PPTP, L2TP dan SSTP (“Best VPN Protocols: OpenVPN vs PPTP vs L2TP vs Others (Comparison),” n.d.)

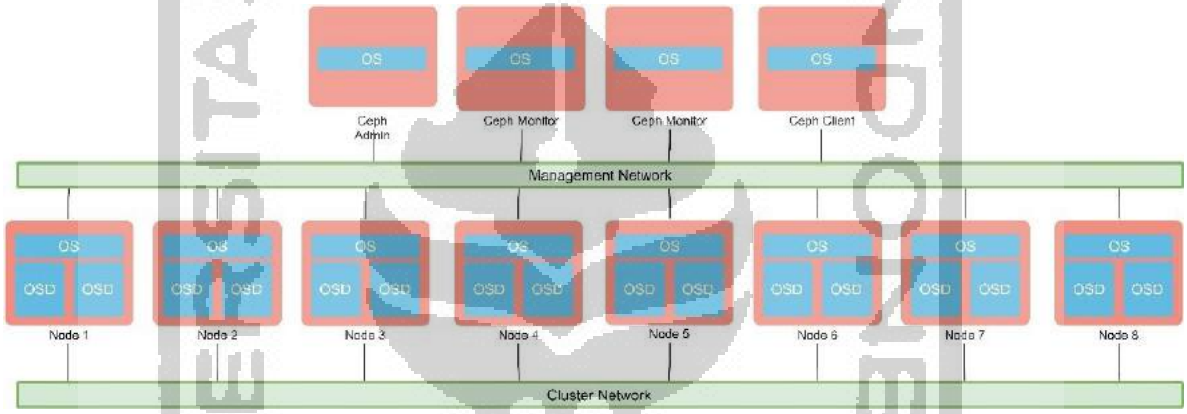
	OpenVPN	**PPTP**	**L2TP/IPsec**	**SSTP**
Encryption	160-bit, 256-bit	128-bit	256-bit	256-bit
Security	Very high	Weak	High	High
Speed	Fast	Speedy, due to low encryption	Medium, due to double encapsulation	Fast
Stability	Very stable	Very stable	Stable	Very stable
Compatibility	Strong desktop support, but mobile could be improved. Requires third-party software.	Strong Windows desktop support.	Multiple device and platform support.	Windows-platform, but works on other Linux distributions.

Selain penggunaan teknologi OpenVPN, untuk meningkatkan proteksi keamanan dalam hal *confidentiality* juga digunakan protocol HTTPS. HTTPS dilengkapi dengan sistem keamanan (*security*) berupa SSL (*Secure Socket Layer*) yang berperan sebagai lapisan pelindung dari protokol jaringan suatu website dengan melakukan enkripsi data. Secara digital, cara kerja SSL adalah dengan mengunci *cryptographic key* ke informasi yang hendak diidentifikasi. Data pun akan terenkripsi dengan baik selama proses *transfer* sehingga pihak ketiga tidak bisa masuk dan mencuri informasi yang sensitif. Tak hanya *private key* dan *public key*, SSL juga memiliki *session key* untuk setiap *secure session* yang unik. Selama koneksi awal, *public key* dan *private key* akan digunakan untuk membuat *session key*, yang kemudian mengenkripsi dan mendekripsi data yang sedang ditransfer. *Session key* ini akan tetap valid untuk waktu yang terbatas dan hanya digunakan di session tertentu.

Berikut kami gambarkan arsitektur penyimpanan bukti digital yang akan kami bangun dengan menggunakan Ceph sebagai Software Defined Storage, ownCloud sebagai layanan antarmuka dan sinkronisasi file, serta OpenVPN sebagai perangkat pengamanan jalur koneksi.



Gambar 3.2 Arsitektur penyimpanan bukti digital dengan Ceph storage cluster



Gambar 3.3 Arsitektur Ceph cluster storage

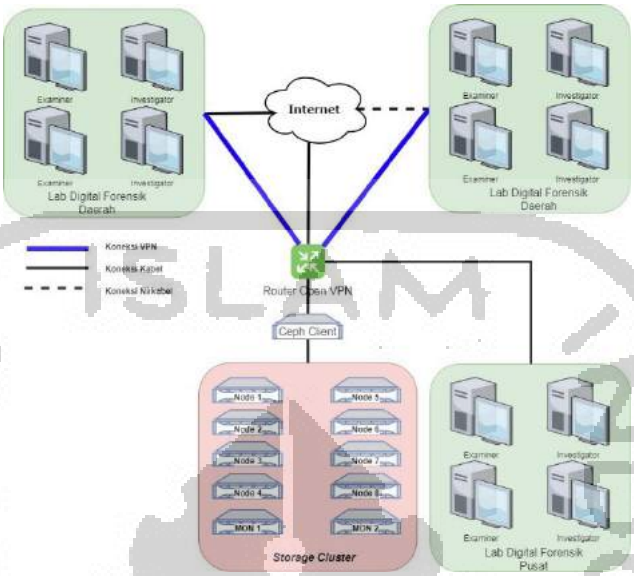
Dalam gambaran arsitektur penyimpanan barang bukti di atas, Ceph merupakan system storage yang diakses menggunakan antarmuka ownCloud dan jalur yang diamankan oleh openVPN. OpenVPN dan ownCloud juga digunakan sebagai komponen pendukung untuk keamanan jalur dan untuk menjaga integrity barang bukti digital yang dikirimkan ke server. OpenVPN digunakan untuk menjaga keamanan transmisi, karena transfer dilakukan menggunakan jaringan Public atau Internet. Sedangkan ownCloud selain berfungsi sebagai layanan antarmuka dan sinkronisasi file, juga untuk menjaga *integrity* file dan memastikan file antara sumber dan yang sudah terupload di server tidak mengalama *corrupt.*

3.2.1 Hardware

Perangkat server dengan spesifikasi CPU(s) 32 x Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz (2 Socket) Memory 368Gb dan Hardisk 1,6 Tb.

3.2.2 Topologi Jaringan

Rancangan sistem *digital evidence storage* yang akan dibangun dalam penelitian ini digambarkan dalam Gambar 3.1.

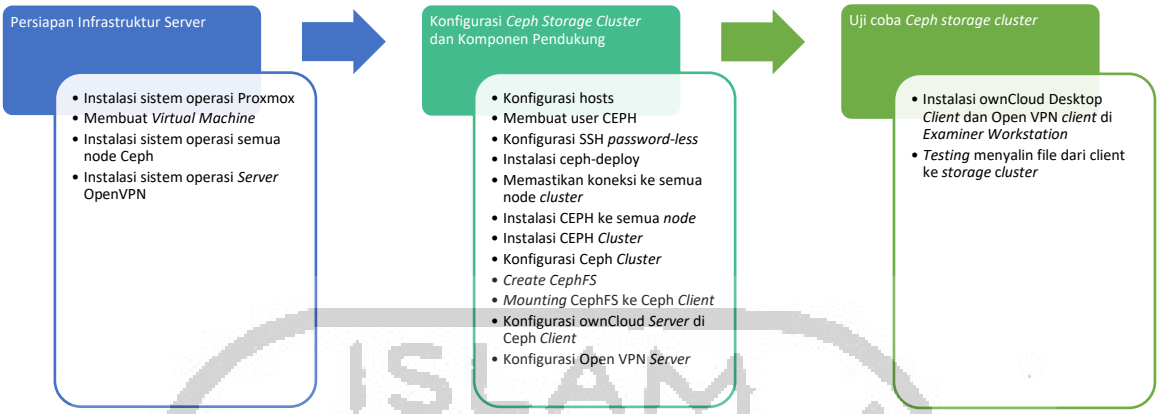


Gambar 3.4 Rancangan sistem *digital evidence storage* dalam penelitian

Rancangan sistem *digital evidence storage* yang dibangun terdiri dari 8 node *server storage*, 2 *server* Ceph monitor (MON) dan 1 Ceph *client*. Setiap node *server storage* memiliki 2 buah OSD yang akan digabungkan dalam sebuah *storage cluster*. Setiap OSD memiliki kapasitas 20Gb. Selanjutnya Ceph *storage cluster* akan ditautkan dari *server* Ceph monitor ke Ceph client. Ceph *client* bertindak sebagai *server* yang akan diakses oleh penyidik menggunakan jaringan terenkripsi. Dalam penelitian ini dibangun 2 jalur koneksi untuk melakukan akses ke dalam sistem yaitu melalui jaringan kabel dan jaringan nirkabel (wireless maupun GSM). Pengujian sistem akan dilakukan pada jaringan nirkabel menggunakan jaringan GSM untuk menggambarkan keadaan jika Laboratorium Digital Forensik di daerah belum terjangkau jaringan kabel (Fiber Optic).

3.3 Implementasi

Tahap implementasi sistem *digital evidence storage* dalam penelitian ini digambarkan pada Gambar 3.2. Terdiri dari beberapa tahapan berupa persiapan infrastruktur server, persiapan Ceph *storage cluster* dan komponen pendukung, serta uji coba Ceph *storage cluster*.



Gambar 3.5 Tahapan proses implementasi rancangan sistem *storage*

3.3.1 Persiapan Infrastruktur Server

Mempersiapkan infrastruktur server merupakan langkah awal dalam implementasi sistem *storage*. Dalam tahap persiapan infrastruktur server dilakukan dengan instalasi sistem operasi Proxmox *Virtual Environment* (Proxmox VE) ke dalam server fisik. Proxmox VE merupakan sistem operasi *Open Source* yang berfungsi untuk virtualisasi. Setelah sistem operasi Proxmox VE dikonfigurasi dengan baik, langkah selanjutnya adalah membuat *virtual machine* untuk semua node pada rancangan sistem *digital evidence storage*. Spesifikasi dari setiap node dalam sistem *storage* adalah sebagai berikut :

Tabel 3.2 Spesifikasi Node dalam Penelitian

No	Node	Spesifikasi
1	Ceph Admin	Processors : 4 Core Memory : 2 Gb Hardisk : 40Gb
2	Ceph Client	Processors : 4 Core Memory : 4 Gb Hardisk : 40Gb
3	Server Storage	Processors : 4 Core Memory : 2 Gb Hardisk 1 : 40Gb Hardisk 2 (OSD) : 20Gb Hardisk 3 (OSD) : 20Gb

No	Node	Spesifikasi
4	Ceph Monitor	Processors : 4 Core Memory : 2 Gb Hardisk : 40Gb
5	OpenVPN Server	Processors : 4 Core Memory : 2 Gb Hardisk : 20Gb

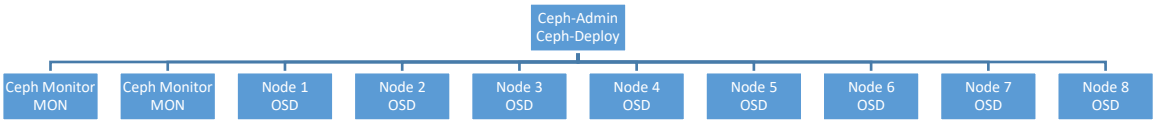
Setelah semua *virtual machine* berhasil dibuat, kemudian dilakukan instalasi sistem operasi pada setiap node menggunakan sistem operasi Ubuntu 18.04.

3.3.2 Konfigurasi Ceph Storage Cluster dan Komponen Pendukung

Dari proses sebelumnya, kita telah memiliki 8 node yang akan diinstall Ceph *storage cluster*. Instalasi dan konfigurasi Ceph *storage cluster* pada node dapat dilakukan sekaligus dalam satu proses alih-alih menginstall satu persatu node. Proses tersebut dapat dilakukan melalui ceph-admin yang berfungsi sebagai simpul yang akan melakukan konfigurasi pada semua node. Ceph-admin ini adalah node tambahan diluar 8 node yang telah disiapkan sebelumnya.

Sebelum melakukan konfigurasi pada semua node, diperlukan pemetaan *IP address* semua node ke dalam ceph-admin, yang dilakukan dengan mendaftarkan semua node ke dalam file `/etc/hosts`. Selain itu ceph-admin juga memerlukan akses SSH tanpa password ke semua node Ceph.

Proses instalasi dan konfigurasi pada semua node dapat dijalankan dari node ceph-admin menggunakan *tools* ceph-deploy. Ceph-deploy adalah alat yang memungkinkan penyebaran Ceph cluster dengan mudah dan cepat tanpa melibatkan konfigurasi manual yang rumit dan terperinci. Ceph-deploy menggunakan SSH untuk melakukan akses ke node Ceph lain dari node Ceph-admin.Ketika ceph-deploy log in ke node Ceph sebagai pengguna, pengguna tersebut harus memiliki hak akses sudo tanpa kata sandi. Dengan adanya ceph-deploy ini, sangat mudah untuk mengatur dan menghapus sebuah cluster. Hal lain yang dapat dilakukan dengan ceph-deploy adalah instalasi paket Ceph pada node jarak jauh, membuat *cluster*, menambahkan monitor, mengumpulkan / melupakan kunci, menambahkan OSD dan server metadata, mengkonfigurasi host admin atau menghapus *cluster*.



Gambar 3.6 Rancangan instalasi Ceph *storage cluster* dengan Ceph-deploy

Setelah ceph-deploy sudah terinstall pada node ceph-admin, selanjutnya melakukan konfigurasi Ceph *storage cluster* menggunakan ceph-deploy, adapun langkahnya adalah sebagai beikut :

1. Inisialisasi node Ceph Monitor
2. Instalasi paket Ceph ke semua node
3. Menyebarkan konfigurasi Ceph Monitor
4. Mendorong konfigurasi dan kunci *client.admin* ke semua node
5. Menambahkan OSD kedalam sistem
6. Melakukan pengecekan kesehatan *Cluster*

Setelah ceph cluster berjalan, dibutuhkan Ceph Filesystem (CephFS). CephFS merupakan filesystem yang mendukung POSIX yang menggunakan Ceph *storage cluster* untuk menyimpan data. Selanjutnya melakukan *mounting* CephFS ke node Ceph Client.

Langkah selanjutnya setelah melakukan konfigurasi CephFS adalah melakukan konfigurasi ownCloud pada *directory* hasil mounting CephFS tersebut. OwnCloud digunakan dalam penelitian ini untuk memudahkan manajemen layanan dan sinkronisasi file barang bukti digital antara file hasil akuisisi dan Ceph *client*. OwnCloud dapat mengatur layanan *cloud storage* mulai dari *user*, *group user*, *limited* akses, *size quota user*, fitur sinkronisasi otomatis, melakukan *transfer* data menggunakan enkripsi SSL dan melakukan verifikasi *checksum* terhadap file yang ter-*upload* atau *download*. Di dalam penelitian ini untuk melakukan sinkronisasi file barang bukti digital menggunakan OwnCloud Desktop Sync.

Konfigurasi OpenVPN server digunakan agar sistem *storage* dapat diakses menggunakan jaringan internet. Dalam penelitian ini dibangun 2 jalur koneksi untuk melakukan akses ke dalam sistem yaitu melalui jaringan kabel dan jaringan nirkabel (wireless maupun GSM). Selain agar dapat diakses menggunakan jaringan internet, konfigurasi OpenVPN juga digunakan sebagai upaya pengamanan transmisi data. Pemilihan OpenVPN sebagai infrastruktur jalur koneksi dikarenakan OpenVPN berbasis teknologi *open source* seperti OpenSSL encryption library dan protokol SSL V3/TLS V1 serta

memiliki tingkat keamanan yang tinggi. OpenVPN memiliki keunggulan yaitu cepat, fleksibel, dan aman. Tidak peduli sistem atau platform operasi yang digunakan, sistem akan terlindungi dengan baik. Open VPN server ini diinstall pada *virtual machine*.

3.3.3 Uji coba Ceph *storage cluster*

Uji coba Ceph *storage cluster* dengan melakukan instalasi ownCloud Desktop Client dan Open VPN *Client* di *Examiner Workstation*. Setelah itu di lakukan uji coba sinkronisasi file dari *client* ke *storage cluster*

3.4 Pengujian

Pada tahap ini dilakukan 4 skenario pengujian untuk mendapatkan data yang akan dianalisa. Hasil pengujian yang dijalankan selain diharapkan memberikan hasil yang baik dari kriteria skalabilitas, juga diharapkan dapat memenuhi kriteria *confidentiality*, *integrity*, dan *availability*. Hal ini berkaitan untuk menjaga agar bukti digital yang disimpan dalam *storage* dapat diterima dan digunakan sebagai barang bukti di persidangan.

Kriteria *confidentiality* yang berarti menjaga informasi dari orang yang tidak berhak mengakses, dalam penelitian ini dilakukan dengan :

- pemanfaatan teknologi OpenVPN
- menggunakan protokol HTTPS
- otorisasi user / password dengan token

Kriteria *integrity* yang berarti menjaga konsistensi, akurasi, dan kepercayaan terhadap data dari waktu ke waktu. Menjaga integrity juga berarti memastikan bahwa data tidak bisa diubah-ubah. Dalam penelitian ini dilakukan dengan :

- menerapkan *strong encryption* pada media penyimpanan dan transmisi data.
- menerapkan *strong authentication* dan *validation* pada setiap akses file/akun login/action yang diterapkan. Authentication dan validation dilakukan untuk menjamin legalitas dari akses yang dilakukan.
- menerapkan *access control* yang ketat ke sistem, yaitu setiap akun yang ada harus dibatasi hak aksesnya. Misal tidak semua penyidik memiliki hak akses ke barang bukti yang aktif diperiksa, hanya yang bertugas untuk menyelidikinya yang bisa mengakses, sementara penyidik lainnya hanya bisa melihat saja.

Maksud dari *availability* adalah memastikan sumber daya yang ada siap diakses kapanpun oleh *user / application/* sistem yang membutuhkannya. Dalam penelitian ini, aspek *availability* dicapai melalui penggunaan Ceph yang memiliki kemampuan *self-healed*, *self-managed*, dan *reliable* terhadap *multiple-failure*.

3.4.1 Pengujian Skalabiltas

Pengujian mengenai skalabilitas system *storage* dilakukan dengan menambahkan 2 node ke dalam sistem yang semula memiliki 8 node, sehingga total akan ada 10 node yang berjalan dalam sistem.

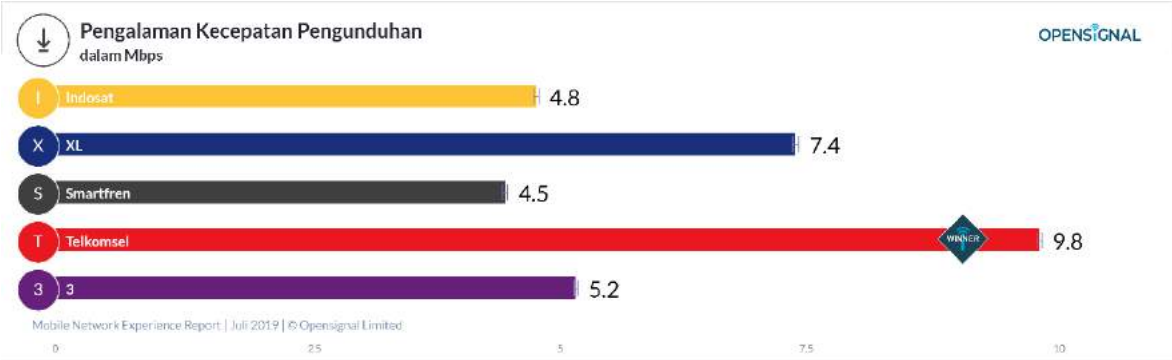
3.4.2 Pengujian Confidentiality

Skenario pengujian *confidentiality* dilakukan dengan menggunakan metode *sniffing* (penyadapan) pada setiap paket yang melewati jaringan, kemudian melihat dan menganalisa hasilnya. *Sniffing* adalah aktivitas menyadap paket data yang sedang berjalan pada *traffic* sebuah jaringan. Sedangkan alat yang digunakan untuk melakukan pengujian ini adalah Wireshark Network Protocol Analyzer.

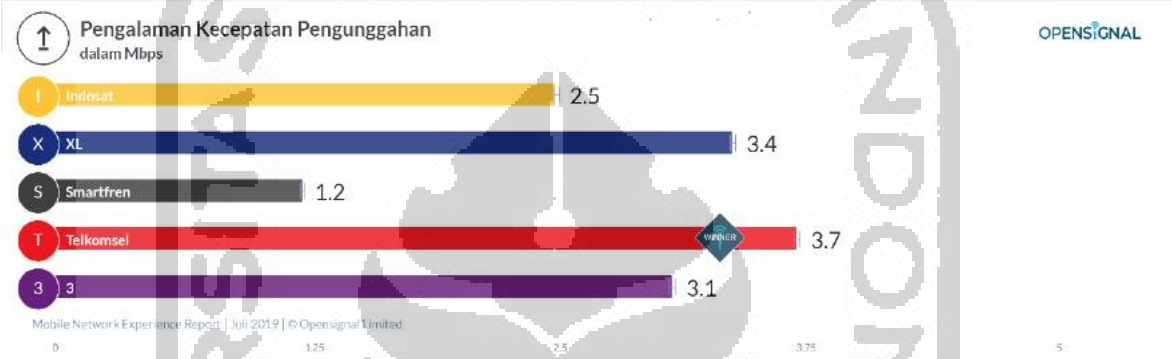
Wireshark merupakan *tool network analyzer* dan *capturing* packet secara visual dengan fasilitas yang cukup lengkap untuk melihat jalur data serta protokol yang digunakan.

3.4.3 Pengujian Integrity

Skenario pengujian *integrity* data dilakukan dengan pengujian sinkronisasi file dengan menggunakan jaringan 4G beberapa operator seluler dalam beberapa kualitas koneksi dengan parameter *ping time*, *jitter*, besaran *bandwidth*, serta *packet loss*. Pemilihan operator seluler dilakukan berdasarkan Laporan Pengalaman Jaringan Seluler di Indonesia Juli 2019 yang dirilis oleh opensignal.com. Operator seluler yang masuk dalam objek penelitian tersebut adalah Indosat, XL, Smartfren, Telkomsel dan 3. Laporan tersebut menyatakan bahwa Telkomsel menempati nilai tertinggi dengan rata-rata kecepatan pengunduhan 12.8 Mbps dan kecepatan pengunggahan 3.7 Mbps. Sementara nilai terendah dalam hal rata-rata kecepatan pengunduhan dan pengunggahan dimiliki oleh Smartfren, yaitu dengan nilai rata-rata kecepatan pengunduhan 4.5 Mbps dan kecepatan pengunggahan 1.2 Mbps.



Gambar 3.7 Grafik Pengalaman Kecepatan Pengunduhan Operator Seluler di Indonesia Periode Juli 2019 (Khatri, 2019)



Gambar 3.8 Grafik Pengalaman Kecepatan Pengunggahan Operator Seluler di Indonesia Periode Juli 2019 (Khatri, 2019)

Dari kedua grafik tersebut, dapat dilakukan pemeringkatan operator seluler berdasarkan pengalaman kecepatan pengunduhan dan pengunggahan dengan urutan sebagai berikut : Telkomsel, XL, 3, Indosat, dan Smartfren. Dalam penelitian ini dipilih operator Telkomsel untuk mewakili kondisi jaringan yang *reliable*, serta Indosat dan Smartfren untuk mewakili kondisi jaringan yang *unreliable*. Pengujian sinkronisasi file dilakukan dengan melakukan pembatasan besaran *bandwidth*.

3.4.4 Pengujian Availability

Skenario pengujian *availability* dilakukan dengan membuat kondisi beberapa node Ceph terputus ketika dilakukan transfer file. File yang ditransfer dalam hal ini berupa file rekaman CCTV dengan ekstensi .mp4 yang berukuran 91MB. Kami membuat 2 keadaan node terputus yaitu ketika 2 node terputus dan 4 node terputus lalu membandingkan hasilnya.

3.5 Penilaian Sistem oleh Ahli

Setelah melalui 4 skenario pengujian untuk menguji apakah sistem *storage* dapat berfungsi dengan baik dan mendapatkan hasil dari aspek skalabilitas, *confidentiality*, *integrity* dan

availability, dilakukan penilaian oleh ahli forensika digital. Penilaian dilakukan dengan kuisioner yang telah disiapkan sebelumnya. Jenis kuisioner yang diberikan adalah kombinasi antara kuisioner terbuka dan kuisioner tertutup. Hal ini peneliti lakukan agar dapat menggali informasi yang lebih dalam mengenai sistem *storage* penyimpanan bukti digital. Pertanyaan dalam kuisioner terkait dengan keberfungsian sistem *storage* dalam hal penyimpanan bukti digital. Pengisian kuisioner dilakukan setelah responden mencoba menggunakan sistem storage dengan cara mengunggah file, melakukan sinkronisasi, dan mengunduh file. Hasil kuesioner ini mutlak sesuai dengan ilmu pengetahuan responden dan tidak ada paksaan atau sejenisnya.