Implementasi FTP Server Dengan Performa Secure File Transfer Protocol Berbasis Virtualiasasi Dan Container

Mahandika Dias Sahputra[1], I Made Suartana[2]
[1,2] Jurusan Teknik Informatika Fakultas Teknik Universitas Negeri Surabaya
[1]mahandika.17051204055@mhs.unesa.ac.id
[2]imadesuartana@unesa.ac.id

Abstrak — Dalam upaya memanfaatkan jaringan komputer, File Transfer Protocol (FTP) merupakan metode pilihan yang tepat dalam penyimpanan data dengan kecepatan transfer yang lebih baik, terutama file berukuran besar, namun seiring dengan semakin berkembangnya teknologi, semakin banyak alat-alat yang digunakan untuk melakukan kejahatan termasuk penyerangan, seperti Sniffing contohnya. Sniffing dilakukan dengan memantau atau menganalisa paket data yang dikirim kan dari komputer client ke komputer server melalui jaringan komputer dan merupakan tindakan hacking yang paling mudah dan sulit diantisipasi dengan berbagai tools yang digunakan dan gratis misalnya Wireshark, tentu saja bisa berbahaya untuk user yang tak berhati-hati terhadap keamanan jaringan komputer.

Pengiriman file yang aman diperlukan penerapan transfer da ta & autentikasi pada FTP server dengan memakai *secure shell*, sehingga proses transfer data harus melalui mekanisme kredensial dan pengiriman file terlebih dahulu di enkripsi menjadi chipertext. Pembatasan upload file yang berada dalam jaringan FTP server dilakukan agar tidak banyak menyimpan data & mengakibatkan penuhnya kapasitas disk yang ada. Untuk mengatasi setiap user dalam jaringan FTP server agar tidak menyimpan data melebihi kapasitas yang ada maka pergunakanlah pembatasan disk quota pada setiap user, se hingga user pada jaringan FTP server tidak sembarangan menyimpan data/file pada komputer FTP server.

Hasil penelitian ini untuk pengiriman file via SFTP menggunakan jaringan container mendapatkan hasil rata-rata 101,044KBps, sedangkan jika menggunakan jaringan virtual machine mendapatkan hasil 116,709KBps. Penggunaan topologi peer to peer pada environtment container dan virtual machine serta operating system yang sama, pengiriman file menggunakan SFTP pada virtual machine dan container memiliki perbedaan kecepatan rata-rata sebesar 15,665KBps.

Kata Kunci - *File Transfer Protocol (FTP), Secure Shell (SSH),* Virtualisasi, *Docker Container*.

I. PENDAHULUAN

Perkembangan Teknologi pada saat ini berkembang sangat pesat pada tiap-tiap aspek, baik dalam bidang teknologi maupun ilmu pengetahuan. Dalam perkembangan teknologi yang pesat ini juga memiliki dampak pada penyebaran informasi secara cepat serta mudahnya untuk mendapatkan informasi tersebut. Perkembangan teknologi ini memiliki pengaruh yang kuat dengan majunya suatu instansi, hal ini dikarenakan perkembangan teknologi membuat semua orang lebih mudah dalam menyelesaikan tugas-tugasnya. Seperti dalam melakukan pengiriman serta menjaga keamanan data berbentuk digital misalnya. Seseorang dapat dengan mudah melakukan transfer data digital serta tidak perlu khawatir tentang keamanan data kantor dengan menggunakan *File Transfer Protocol* (FTP).

File merupakan suatu data digital yang bentuknya dapat dilihat dan ditransfer. Dalam upaya memanfaatkan jaringan komputer, *File Transfer Protocol* (FTP) merupakan metode pilihan yang tepat dalam penyimpanan data dengan kecepatan transfer yang lebih baik, terutama file berukuran besar, namun seiring dengan semakin berkembangnya teknologi, makin ba nyak juga alat-alat yang dipakai untuk melakukan kejahatan termasuk penyerangan, seperti *Sniffing* contohnya. *Sniffing* dilakukan dengan memantau atau menganalisa paket data yang dikirimkan dari komputer *client* ke komputer *server* melalui jaringan komputer dan merupakan tindakan *hacking* yang paling mudah dan sulit diantisipasi dengan berbagai *tools* yang digunakan dan gratis misalnya *Wireshark*, tentu saja bisa berbahaya. pengguna yang tidak berhati-hati terhadap keamanan jaringan komputer. Wireshark ialah alat analisis protokol jaringan yang paling terkemuka dan banyak digunakan di Indonesia. Ini memungkinkan kita melihat apa yang terjadi di jaringan kita pada tingkat mikroskopis. Wireshark memiliki antarmuka pengguna grafis (GUI). Wireshark menyajikan beberapa keunggulan yang membuat nya menarik untuk penggunaan sehari-hari. Ini dikhususkan untuk pekerja harian & analis paket ahli saat menawarkan ber bagai fitur untuk memikat masing-masing [1]. Wireshark me miliki beberapa manfaat yakni: *supported protocols, cost, user-friendliness, program support&operating system support.*

Keamanan jaringan merujuk pada metode dan usaha yang digunakan untuk melindungi sistem agar terhindar dari serangan pihak yang tidak bertanggung jawab yang berupaya mencuri data. Ancaman keamanan yang ada adalah kemampuan penyadap untuk menangkap dan mengambil semua lalu lintas masuk dan keluar, termasuk nama pengguna, kata sandi, dan informasi data olahan di jaringan. FTP hingga saat ini masih menjadi pilihan sebagai media yang dipergunakan untuk mentransfer data dengan proses *upload* dan *download* yang cepat dari *server* ke *client* [2].

Maka, diperlukan penerapan transfer data dan autentikasi pada server FTP dengan menggunakan Secure Socket Layer (SSL) dan Secure Shell (SSH), sehingga proses autentikasi dan transfer data dienkripsi terlebih dahulu menjadi *chipertext*. Jika tidak ada batasan untuk mengunggah file, pengguna dalam jaringan server FTP dapat terus menyimpan data di komputer server FTP, yang dapat mengakibatkan kapasitas hard disk penuh. Untuk mengatasi masalah ini, disarankan untuk menerapkan pembatasan disk quota pada

setiap pengguna dalam jaringan FTP server, sehingga user pada jaringan FTP *server* tidak sembarangan menyimpan data atau file pada komputer FTP *server*. Selain itu Dalam penelitian ini, penulis akan merancang sebuah sistem kontrol server dengan memanfaatkan protokol SSH (Secure Shell). Protokol ini memungkinkan administrator untuk melakukan kontrol terhadap server tanpa harus berada di dekat komputer server itu sendiri. SSH dirancang khusus untuk melakukan kontrol jarak jauh (remote) terhadap server. Maka dengan protokol SSH (*Secure Shell*) ini administrator dapat melakukan *controlling* terhadap *server* dari jarak jauh. Tujuan dibangunnya jaringan FTP adalah untuk mempermudah dalam melakukan pertukaran data yang menjadi lebih efisien. Selain itu dengan dibangunnya sistem FTP server dapat dijadikan sebagai media penyimpanan alternatif dari pengguna. SSH adalah sebuah protokol jaringan yang memfasilitasi komunikasi dan pertukaran data melalui saluran aman antara dua perangkat dalam jaringan. SSH mengadopsi model komunikasi client-server. Port standar yang digunakan oleh SSH adalah port 22, yang telah ditetapkan sebagai jalur untuk server SSH. Sebuah klien SSH umumnya digunakan untuk menginisiasi koneksi antara client dan server SSH, memungkinkan pengendalian jarak jauh terhadap server tersebut [3]. *Secure Shell* (SSH) menggunakan enkripsi simetris untuk memastikan privasi dan integritas data yang dipertukarkan melalui jaringan. Aspek keamanan dan keandalan merupakan hal penting terhadap data yang dikirimkan melalui jaringan. Perangkat lunak yang telah dirancang dan dibangun dapat digunakan untuk mengirim data melalui protokol SSH dan memberikan kompresi menggunakan algoritma. Perhitungan waktu transmisi rata-rata juga disediakan olehnya [4].

Selain itu, SSH juga mensupport fitur untuk pengiriman berkas / file melalui SFTP (*Secure File Transfer Protocol*) atau SCP (*Secure Copy*). SFTP menyediakan metode untuk melindungi proses transfer data antara client dan server. SFTP merupakan bagian dari SSH dan dikenal dengan koneksi shell yang aman. SSH adalah protokol jaringan yang memungkinkan komputer terhubung secara aman menggunakan enkripsi. Proses pengiriman data melalui SFTP mengubah data menjadi kode-kode tertentu yang hanya dapat dimengerti oleh client, sehingga membantu mencegah kebocoran data & mengurangi risiko peretasan data. Beberapa manfaat menggunakan SFTP yaitu data selalu dienkripsi untuk mencegah akses oleh pihak yang tidak berwenang, memverifikasi keaslian sumber data, menggunakan autentikasi kunci publik untuk mengesahkan pengguna yang mengakses server, mematuhi persyaratan keamanan internasional, dan menyediakan autentikasi host agar klien dapat memverifikasi server yang ingin dihubungkan [5].

Docker merupakan aplikasi berbasis teknologi *opensource* yang memungkinkan pengembang/siapa pun untuk membuat, menjalankan, menguji & meluncurkan aplikasi dalam sebuah kontainer. Docker mempermudah proses pembuatan paket aplikasi beserta komponennya dalam kontainer yang terisolasi. Kontainer ini dapat dijalankan di infrastruktur lokal tanpa memerlukan konfigurasi ulang. Docker dikenal sangat ringan dan cepat dibandingkan dengan mesin virtual berbasis *hyper visor*. Docker menggunakan metode file system sebagai lapi san untuk menjalankan kontainer, sehingga hanya menyalin lapisan perubahan dari *image* dasar untuk membuat kontainer baru dengan image dasar yang sama [6]. Hal ini membuat penggunaan sumber daya mesin *host* lebih efisien dibanding kan dengan perangkat lunak virtualisasi berbasis *hypervisor.*

Virtualisasi adalah teknologi yang memungkinkan pengguna untuk mengisolasi dan menjalankan beberapa sistem operasi serta layanan lainnya secara bersamaan pada satu perangkat keras, tanpa memerlukan pengetahuan spesifik tentang spesifikasi fisik seperti sistem operasi, penyimpanan data, memori, dan bandwidth secara langsung. Tujuannya adalah untuk mengoptimalkan penggunaan prosesor, menghindari pemborosan daya listrik, dan mengurangi biaya yang dikeluarkan untuk server. Ada berbagai jenis virtualisasi server, baik yang bersifat open source maupun berbayar [7].

Pada penelitian terdahulu yang dilakukan oleh Ahmad Musa (2020) yang berjudul "*Forensic Analysis of Peer-to-Peer Network Traffic with Wireshark*". Penelitian ini menganalisis bagaimana Wireshark dapat digunakan untuk memantau dan melacak pengguna P2P secara detail. Studi kasus menunjukkan bahwa packet sniffing dapat digunakan untuk mengalahkan anonimitas pengguna online peer-to-peer. Oleh karena itu, melalui penegakan hukum, pengguna P2P dapat dilacak dengan bantuan bukti kredibel yang diperoleh menggunakan Wireshark dan analisis kami menggunakan berbagai alat lainnya. Penelitian kami ditantang dengan kemungkinan rekan-rekan bersembunyi di bawah server proxy atau jaringan pribadi virtual (VPN) untuk menutupi identitas mereka. Tantangan ini dapat diatasi dengan akses yang dimiliki oleh penyelidik digital yang didukung pemerintah. Perusahaan yang terlibat dalam penyamaran dapat dimintai keterangan untuk mengungkap identitas para tersangka [8].

Berdasarkan analisa dari penelitian tersebut, penulis ingin melakukan penelitian dengan judul "Implementasi FTP Server Denagn Performa Secure File Transfer Protocol Berbasis Virtualiasasi Dan Container". Penelitian ini akan menerapkan proses transfer file dalam jaringan menggunakan protocol keamanan SSH pada virtualisasi jaringan sehingga pertukaran file dengan enkripsi menjadi lebih aman.

II. METODOLOGI PENELITIAN

A. Analisa Kebutuhan

Dalam implementasi performa secure file FTP server pada virtualisasi & container, harus menganalisa beberapa kebutuhan terlebih dahulu. Proses ini sangat penting, karena akan dipakai sebagai acuan dalam mendukung proses penelitian ini. Analisa kebutuhan terbagi menjadi beberapa bagian, yaitu:

1. Kebutuhan Data

   Penelitian ini mengambil sumber data yang akan digunakan dari beberapa referensi. Terdapat dua jenis metode pengumpulan data pada penelitian ini, yaitu studi literatur dan juga observasi. Peneliti mengutip beberapa referensi yang relevan dari berbagai macam literatur tentang pembahasan FTP *server*, virtualisasi, container, SSH dan Virtual Box.

   a. Studi Literatur

Literatur yang dipakai meliputi jurnal-jurnal nasional dan internasional, makalah, serta beberapa sumber dari internet.

b. Observasi

Penelitian ini juga melakukan observasi dengan mengunjungi beberapa situs referensi tentang FTP *server*, virtualisasi, container, SSH dan Virtual Box.

2. Analisis Kebutuhan Sistem

Analisis kebutuhan system memiliki tujuan untuk me mudahkan proses perancangan & implementasi dalam pengembangan sistem. Spesifikasi perangkat termasuk salah satu faktor penting yang diperlukan untuk mendu kung proses implementasi Performa Secure File Transfer Protocol Berbasis Virtualiasasi Dan Container adalah :

a. *Processor intel core* i3-6006U
b. RAM 4 GB
c. *Harddisk* 1 TB
d. *Windows* 10 64-bit
e. Linux Ubuntu 64-bit

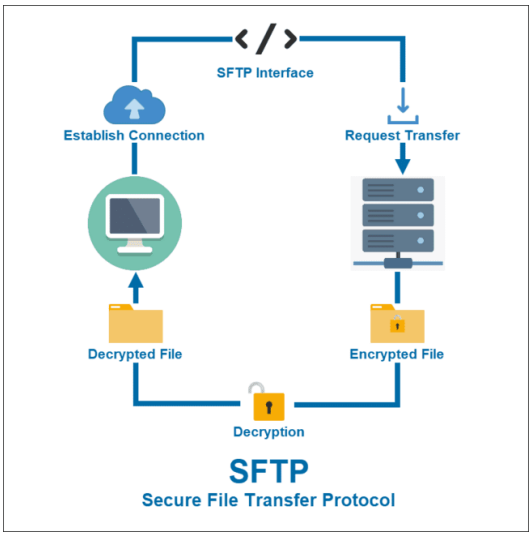
Sedangkan perangkat lunak yang diperlukan dalam penelitian ini adalah sebagai berikut :

a. GNS3 sebagai media untuk mengaplikasikan topologi dan simulasi jaringan dalam *virtual*
b. *Virtual box* digunakan untuk menjalankan *operating system virtual* sebagai komputer klien dan server
c. *Operating system* linux berfungsi sebagai OS untuk menjalankan perintah *command/transfer file*
d. Wireshark sebagai aplikasi untuk menganalisa lalu lintas jaringan
e. Filezilla sebagai aplikasi pengujian pengirim data [9].

B. Desain Sistem

Pada penelitian ini penulis akan mengimplementasi SFTP server untuk keamanan dan kemudahan transfer data. di Dalam desain sistem terdapat tahapan yang akan dilakukan sebagai berikut:

1. Arsitektur SFTP



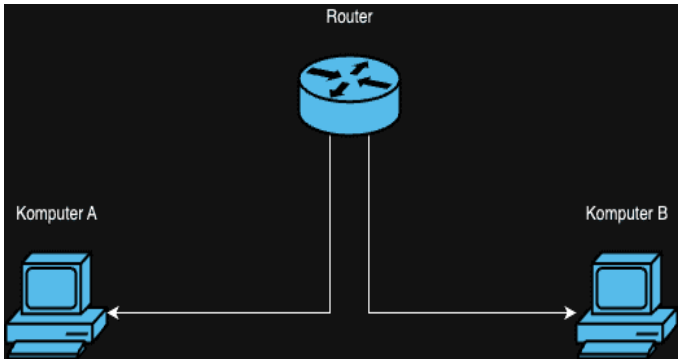
Gbr 1. Arsitektur SFTP (sumber:https://phoenixnap.com/kb/how-does-sftp-work)

Ketika sistem klien meminta *transfer file*, SFTP membuat koneksi aman antara klien dan *server* SFTP. Koneksi ini biasanya melalui port 22. SFTP kemudian menggunakan protokol SSH untuk mengenkripsi *file* yang diminta dan mentransfernya ke klien. Klien dapat mendeskripsi dan mengakses *file* menggunakan salah satu dari beberapa metode otentikasi yang ditawarkan SSH. Metode ini mencakup kombinasi ID pengguna dan kata sandi atau sepasang kunci SSH. Tidak seperti FTP, ID pengguna dan kata sandi dienkripsi dalam SFTP. Pengguna juga dapat mengatur *server* SFTP dengan cara yang tidak memerlukan otentikasi, meskipun opsi ini kurang aman [10].

C. Topologi Jaringan

Topologi ialah suatu komponen penting dalam berjalannya suatu jaringan komputer. Karena topologi merupakan cara untuk menghubungkan komputer satu dengan komputer lainnya untuk membentuk pola hubungan antarterminal dalam suatu sistem jaringan dapat mempengaruhi tingkat efektivitas kinerja jaringan. Berikut beberapa contoh dari jenis topologi yang dapat diimplementasikan dalam jaringan komputer ya itu topologi *Bus*, topologi *Star*, topologi *Peer to Peer* [11]. Metodologi penelitian dapat dijelaskan melalui serangkaian langkah-langkah sebagai berikut:

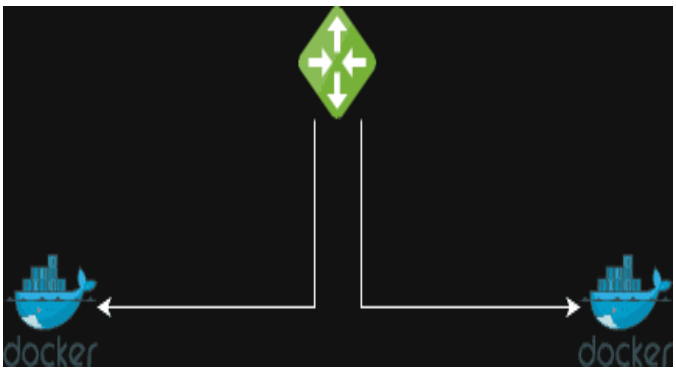
1. Topologi Jaringan *Virtual Machine*



Gbr 2. Topologi jaringan *virtual machine*

Gambar diatas merupakan desain topologi jaringan dari penelitian ini. Jaringan yang digunakan merupakan Peer to peer pada komputer A dan komputer B untuk pengiriman file menggunakan SFTP.

2. Topologi Jaringan *Docker Container*



Gbr 3. Topologi jaringan *Docker Container*

Gambar diatas merupakan desain topologi jaringan container dari penelitian ini. Jaringan yang digunakan merupakan Peer to peer pada container A dan container B untuk pengiriman file menggunakan SFTP.

D. VSFTPD

Very Secure FTP *Daemon* (VSFTPD) dimana *Daemon* dapat diartikan sebagai *service*/sebuah program yang berjalan terus menerus untuk tujuan mengendalikan layanan berkala yang harus didapatkan oleh sebuah *system* komputer. VSFTPD dikembangkan dibawah lisensi GNU (*General Public License*) sebagai sarana untuk men*develope* FTP server pada sistem UNIX, termasuk sistem operasi LINUX, seperti VSFTPD menawarkan keamanan yang dijanjikan namanya. Selain itu, ukurannya yang kecil meningkatkan kecepatan pemrosesan data. VSFTPD mendukung konfigurasi virtual IP, integrasi SSL untuk enkripsi, IPv6, serta pengguna virtual. Kemudahan dalam konfigurasi membuat VSFTPD populer di kalangan pengguna FTP server, dengan banyak keunggulan lainnya.

E. Docker

Docker terdiri dari banyak komponen yang saling terkait dalam menjalankan proses dan tugasnya. Ketergantungan antar komponen tersebut berarti bahwa ketika salah satu komponen tidak tersedia, kinerja Docker dapat terpengaruh. Perbedaan utama antara Docker dan mesin virtual (virtual machine) adalah bahwa kontainer dalam Docker memanfaatkan sumber daya yang terisolasi dan alokasi seperti virtual machine, tetapi dengan arsitektur yang berbeda yang membuat kontainer lebih portabel dan efisien. Mesin virtual memerlukan ruang yang signifikan untuk sistem operasi tamu (guest OS), sedangkan Docker tidak memerlukan ruang besar untuk sistem operasinya. Ini karena Docker memungkinkan penggunaan satu instansi sistem operasi untuk menjalankan beberapa kontainer secara bersamaan [12].

F. Virtualisasi

Virtualisasi adalah teknologi yang berbeda dari mesin fisik atau physical machine. Mesin fisik adalah server yang memiliki berbagai komponen keras seperti power supply, motherboard, memori, disk, dan sebagainya. Di sisi lain, virtualisasi memungkinkan aplikasi untuk berjalan seolah-olah mereka berada dalam satu mesin yang berdiri sendiri, padahal sebenarnya aplikasi tersebut berjalan di atas mesin lain bersama dengan aplikasi lainnya. Virtualisasi adalah teknik atau cara untuk menciptakan suatu entitas dalam bentuk virtual, yang berbeda dengan eksistensi sesungguhnya. Dalam konteks komputasi, virtualisasi digunakan untuk mengemulasikan perangkat keras fisik komputer atau menciptakan perangkat yang tidak ada dalam bentuk virtual, yang memberikan fleksibilitas dan efisiensi dalam pengelolaan sumber daya komputasi [13].

G. Analisa Hasil Penelitian

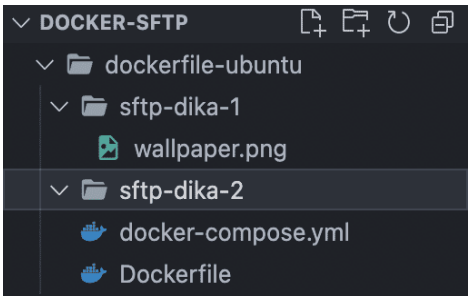
Penarikan kesimpulan dari analisa hasil penelitian me ngacu pada data yang diteliti, yaitu menghasilkan sistem yang saling mengirim file antara ruang percobaannya secara terstruktur dan keamanannya pun terjamin.

III. HASIL DAN PEMBAHASAN

Implementasi dalam perbandingan performa pengiriman file melalui ssh dilakukan dengan dua environtment yaitu virtualisasi os/container (docker container) dan juga virtualisasi mesin (virtual box).

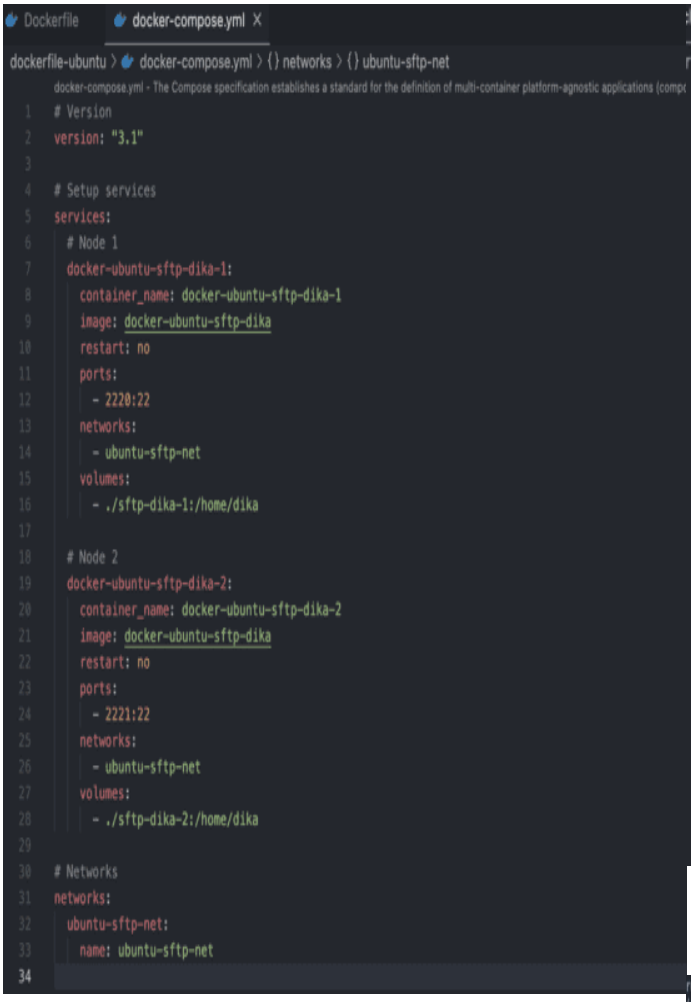
A. Virtualisasi OS (Docker Container)

Pengiriman file melalui virtualisasi container dilakukan dengan menggunakan docker container. Base operating system yang digunakan adalah Ubuntu 20.4. Dalam ubuntu tersebut telah diinstall openssh server dan library untuk melakukan test koneksi, konfigurasi ssh server juga diterapkan dalam dockerfile tersebut.



Gbr 4. Struktur direktori

Gambar diatas merupakan sktruktur direktori dari docker-sftp. Terdapat dua file dari folder sftp-dika-2 yaitu file docker docker compose.yml dan file Dockerfile.



Gbr 5. Dockerfile

Gbr. 5 merupakan source code dari file Dockerfile. File ini akan merancang build image.

Gbr 6. Docker-compose

Gambar diatas merupakan Docker-compose. Ini dilakukan untuk menjalankan image tersebut layaknya container.

Gbr 7. Docker build image

Gbr 7 merupakan Docker build image. Ini dilakukan untuk membuild image.

Gbr 8. Docker compose up

Gambar diatas merupakan Docker compose up. Perintah docker-compose up ini dipergunakan untuk mengaktifkan se mua *image* yang sudah di bangun.

Gbr 9. Docker image ls

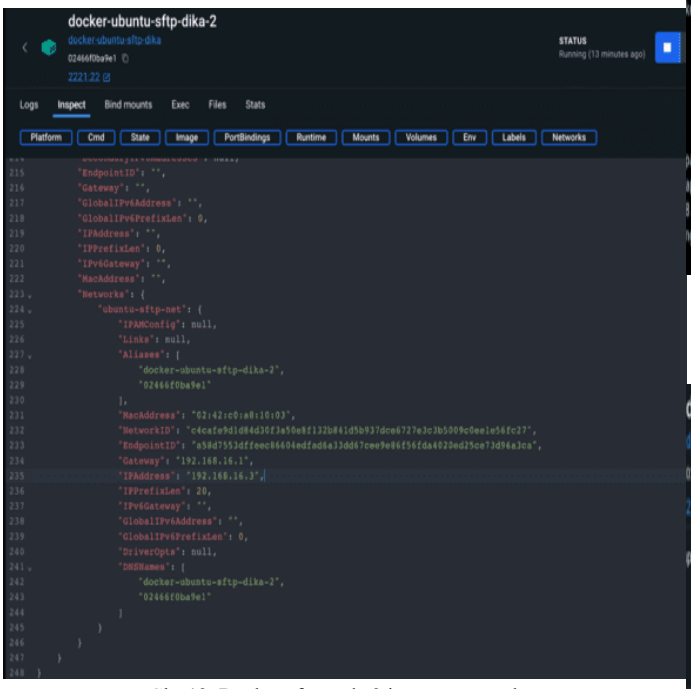
Gambar diatas merupakan Docker image ls. Perintah ini dilakukan untuk melihat image yang telah berjalan.

Gbr 10. Docker container ls

Gambar diatas merupakan Docker container ls. Perintah ini untuk melihat container yang telah berjalan.

Gbr 11. Docker sftp node 2 sebelum menerima file

Gbr 11 ialah Docker sftp node 2 sebelum menerima file.



Gbr 12. Docker sftp node 2 inspect network

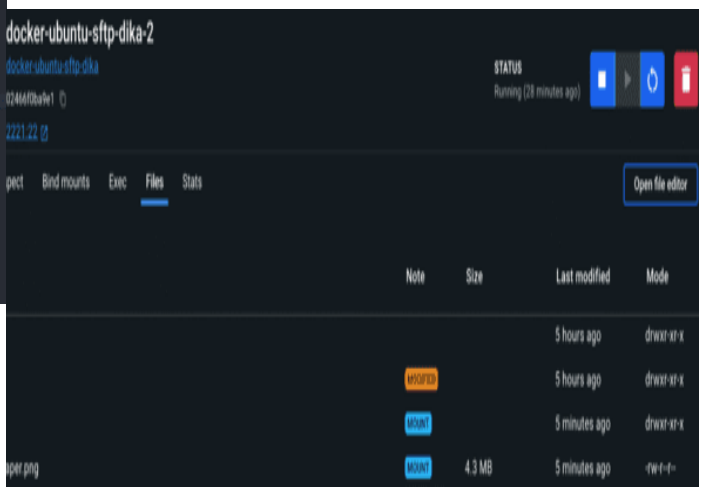
Gambar diatas merupakan gambar Docker sftp node 2 inspect network. Perintah docker inspect untuk melihat secara detail properti entitas docker pada container docker-ubuntu-sftp-dika-2.

Gbr 13. Cek koneksi antar container

Gbr 13 berfungsi untuk melakukan pengecekan koneksi antar container.

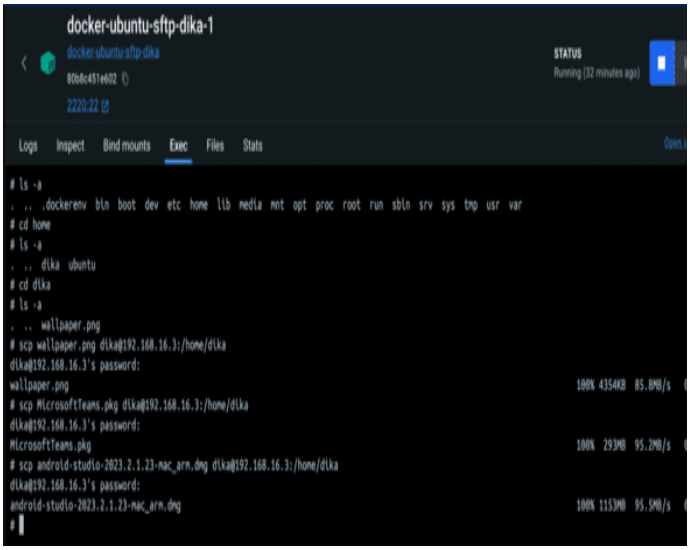
Gbr 14. Mengirim file SFTP docker container

Gambar diatas merupakan source code untuk perintah mengirim file sftp docker container.



Gbr 15. File SFTP pada node 2

Gbr 15 diatas merupakan file dari sftp pada node 2.



Gbr 16. Pengujian pengiriman dengan docker container

Gambar diatas merupakan source code pengujian pengiriman menggunakan docker container.

TABEL I
HASIL PENGUJIAN PERFORMA SFTP DENGAN DOCKER CONTAINER

No.	Ukuran file (byte)	Waktu Tempuh (hh:ss)	Rata-rata per second
1	4.354	00:00	4 KBps
2	307.068.737	00:03	102.356 KBps
3	1.208.590.551	00:12	100.715 KBps

Tabel I merupakan pengujian performa sftp dengan docker container. Untuk ukuran file 4354kb dengan kecepatan 00:00 dengan rata-rata 4 KBps. Untuk ukuran file 307.068.737 b dengan kecepatan 00:03 dengan rata-rata 102.356 KBps Lalu untuk ukuran file 1.208.590.551 dengan kecepatan 00:12. dengan rata-rata 100.715 KBps.

B. Virtualisasi Mesin (Virtual Box)

Pengiriman file melalui virtualisasi container dilakukan dengan menggunakan docker container. Base operating system yang digunakan adalah Ubuntu 20.4.

```
root@ubuntu22:/home/vboxuser# apt update
Hit:1 http://id.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://id.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://id.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Fetched 229 kB in 2s (112 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
49 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ubuntu22:/home/vboxuser# apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  linux-headers-6.5.0-26-generic linux-hwe-6.5-headers-6.5.0-26
  linux-image-6.5.0-26-generic linux-modules-6.5.0-26-generic
  linux-modules-extra-6.5.0-26-generic ubuntu-pro-client
The following packages will be upgraded:
  accountsservice alsa-ucm-conf bash cups cups-bsd cups-client cups-common
  cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc cups-server-common
  dnsmasq-base dpkg firmware-sof-signed gir1.2-accountsservice-1.0 less
  libaccountsservice0 libcups2 libcupsimage2 libexpat1 libgpgme11 libgpgmepp6
```

Gbr 17. Update dan upgrade

Gbr 17 merupakan gambar untuk perintah update dan upgrade ubuntu, digunakan untuk memperbarui repositori paket sistem agar memastikan ubuntu mendapatkan versi program terbaru.

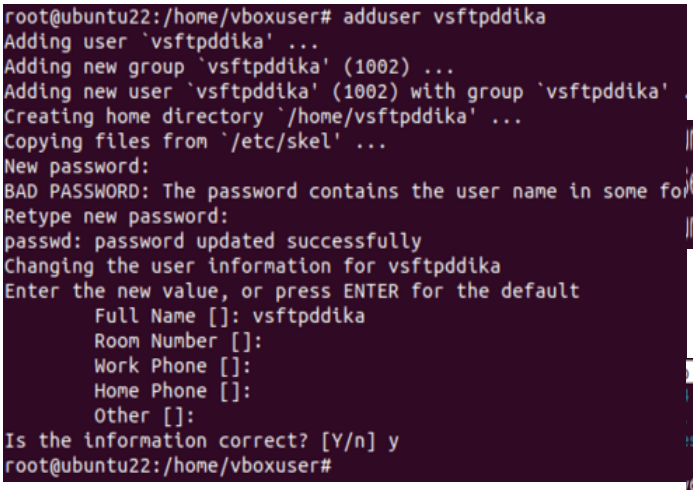
```
ubuntu22:/home/vboxuser# apt install vsftpd
ing package lists... Done
ding dependency tree... Done
ing state information... Done
following packages were automatically installed and are no longer required:
nux-image-6.5.0-18-generic linux-modules-6.5.0-18-generic
ux-modules-extra-6.5.0-18-generic
 apt autoremove' to remove them.
following NEW packages will be installed:
ftpd
graded, 1 newly installed, 0 to remove and 0 not upgraded.
to get 123 kB of archives.
 this operation, 326 kB of additional disk space will be used.
 http://id.archive.ubuntu.com/ubuntu jammy/main amd64 vsftpd amd64 3.0.5-0u
1 [123 kB]
ed 123 kB in 1s (186 kB/s)
nfiguring packages ...
cting previously unselected package vsftpd.
ding database ... 241794 files and directories currently installed.)
aring to unpack .../vsftpd_3.0.5-0ubuntu1_amd64.deb ...
cking vsftpd (3.0.5-0ubuntu1) ...
ing up vsftpd (3.0.5-0ubuntu1) ...
ted symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service -> /l
ystemd/system/vsftpd.service.
```

Gbr 18. *Install* VSFTPD

Gbr 18 merupakan gambar untuk menginstall ubuntu. Disini penulis menggunakan FTP *Server* yaitu VSFTPD karena *server* FTP berlisensi GPL yang dikembangkan untuk menyediakan cara yang aman namun cepat untuk mentransfer file.

Gbr 19. Status VSFTPD

Gbr 19 merupakan gambar untuk melihat status VSTPD Setelah instalasi, sangat penting untuk memeriksa apakah layanan VSFTPD aktif dan berjalan tanpa kesalahan menggunakan utilitas "status" dengan perintah "systemctl".

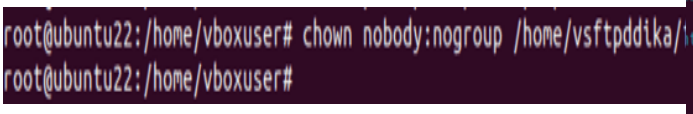


Gbr 20. *Add user*

Gbr 20 merupakan gambar untuk melakukan *add user* pada ubuntu. *User* yang dibuat dengan nama pengguna "vstpddika". Kemudian, tetapkan kata sandi yang kuat.

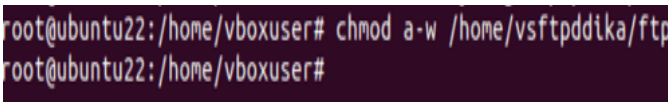
Gbr 21. Membuat *folder* FTP

Gbr 21 merupakan gambar untuk membuat *folder* FTP. Folder yang dibuat terletak pada "/home/vsftpddika/ftp".



Gbr 22 Konfigurasi *ownership*

Gbr 22 merupakan gambar untuk konfigurasi *ownership* direktori.

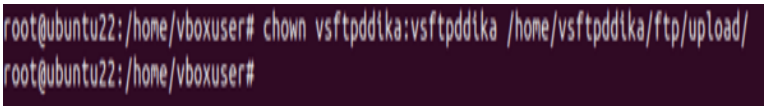


Gbr 23. *Remove write permission* FTP *folder*

Gbr 23 merupakan gambar untuk melakukan *remove write permission* pada FTP *folder.*

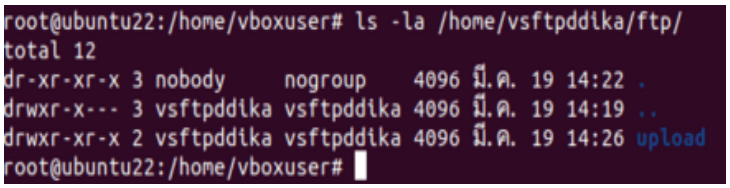
Gbr 24. Membuat direktori untuk upload *file*

Gbr 24 merupakan gambar untuk membuat direktori untuk upload *file.*



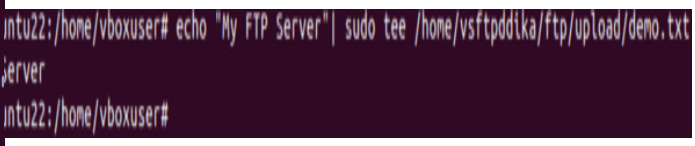
Gbr 25. Konfigrasi *folder upload ownership*

Gbr 25 merupakan gambar *konfigurasi folder upload ownership* untuk menetapkan folder tersebut ke user vstpddika.



Gbr 26. Cek *permission* direktori FTP

Gbr 26 merupakan gambar cek *permission* direktori FTP.



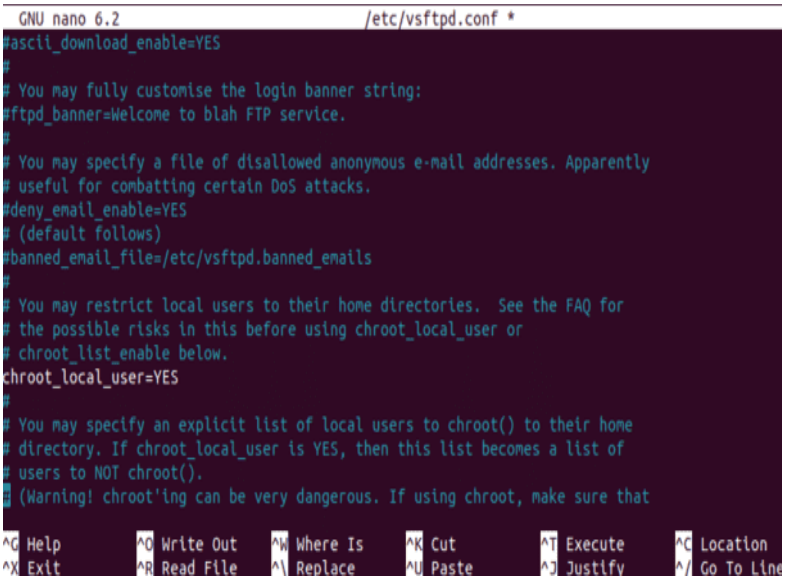
Gbr 27. Tes FTP

Gbr 27 merupakan gambar untuk perintah melakukan tes FTP.

Gbr 28. Konfigurasi VSFTPD 1

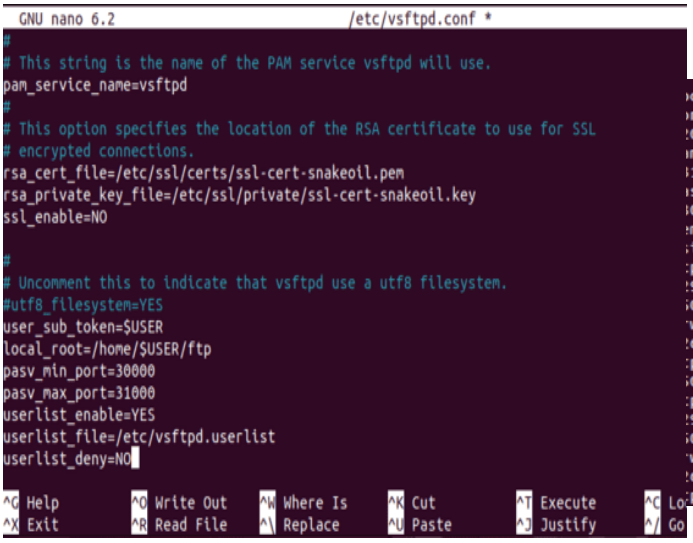
Gbr 28 merupakan gambar konfigurasi VSFTPD 1. Pastikan pada "anonymous_enable" diset menjadi "NO" dan pada "local_enable" diset menjadi "YES".Konfigurasi tersebut berguna untuk mencegah login anonim dan mengizinkan login lokal, karena setelah mengaktifkan login lokal berarti setiap pengguna normal yang terdaftar di file

/etc/passwd dapat digunakan untuk login. Dan hapus tanda "#" didepan "write enable=YES" karena beberapa perintah FTP memungkinkan pengguna untuk menambah, mengubah, atau menghapus file dan direktori pada sistem file.



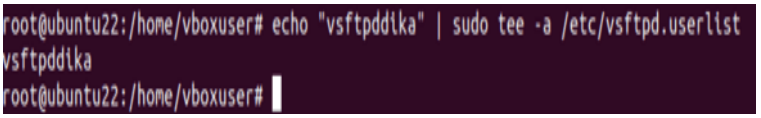
Gbr 29. Konfigurasi VSFTPD 2

Gbr 29 merupakan gambar konfigurasi VSFTPD 2. Batalkan komentar pada "chroot_local_user=YES" untuk men cegah *user* yang terhubung dengan FTP mengakses file / perintah apapun diluar direktori.



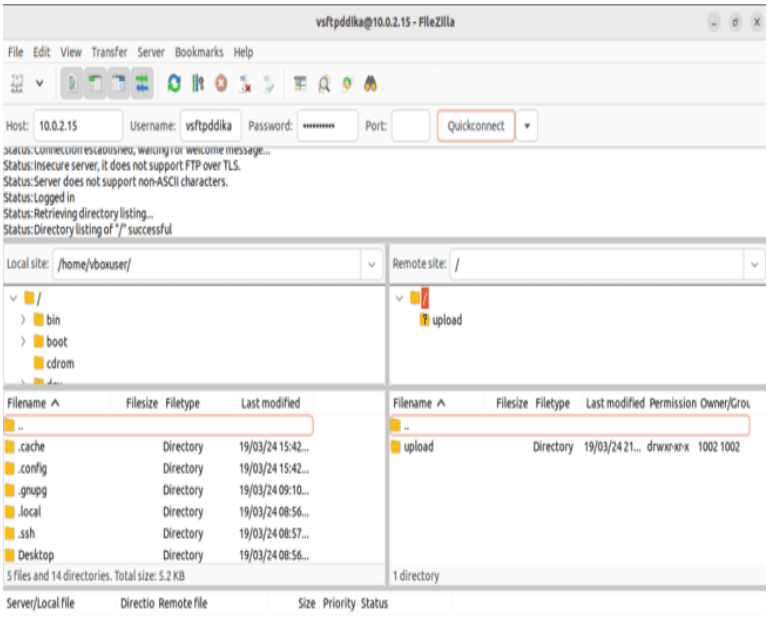
Gbr 30. Konfigurasi VSFTPD 3

Gbr 30 merupakan gambar konfigurasi VSFTPD 3. Menambahkan direktif "user_sub_token" yang nilainya ialah variabel $USER. Kemudian tambahkan direktiff "local_root", konfigurasi ini memastikan bahwa *user* ini dan *user* selanjut nya akan diarahkan ke direktori *home uset* yang sesuai saat masuk. Batasi jangkauan *port* yang dapat dipergunakan FTP pasif untuk memastikan ketersediaan koneksi yang cukup. Lalu untuk mengizinkan akses FTP berdasarkan kasus per kasus, atur konfigurasi sehingga user hanya memiliki akses ketika mereka secara eksplisit ditambahkan ke "userlist", bukan secara default.



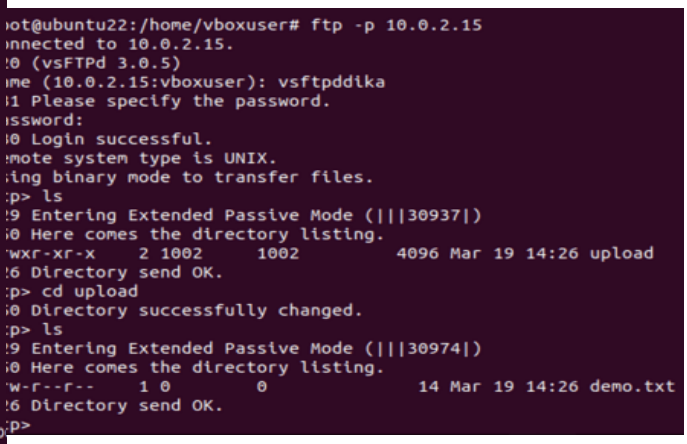
Gbr 31. *Add created user to the VSFTPD user list*

Gbr 31 merupakan gambar untuk *add created user to VSFTPD user list* letaknya di "/etc/vsftpd.userlist".



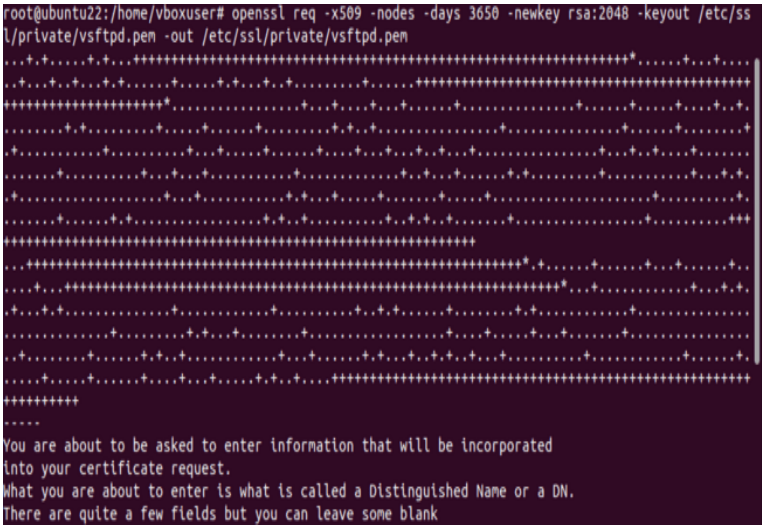
Gbr 32. *Tes VSFTPD filezilla*

Gbr 32 merupakan gambar untuk melakukan *tes VSFTPD filezilla.*



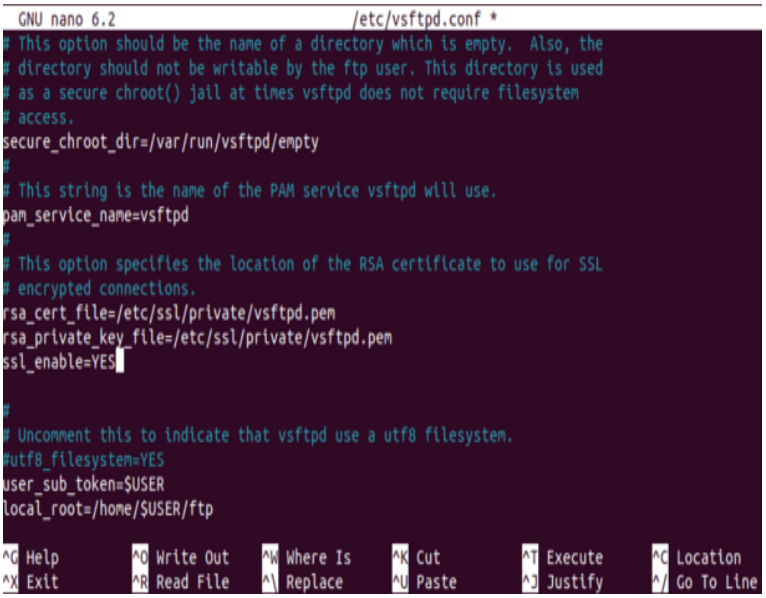
Gbr 33. Tes VSFTPD

Gbr 33 merupakan gambar untuk melakukan tes VSFTPD.



Gbr 34. *Enable TLS/SSL*

Gbr 34 merupakan gambar untuk mengaktifkan TLS/SSL untuk menyediakan enkripsi. Dan membuat sertifikat SSL untuk digunakan dengan VSFTPD.

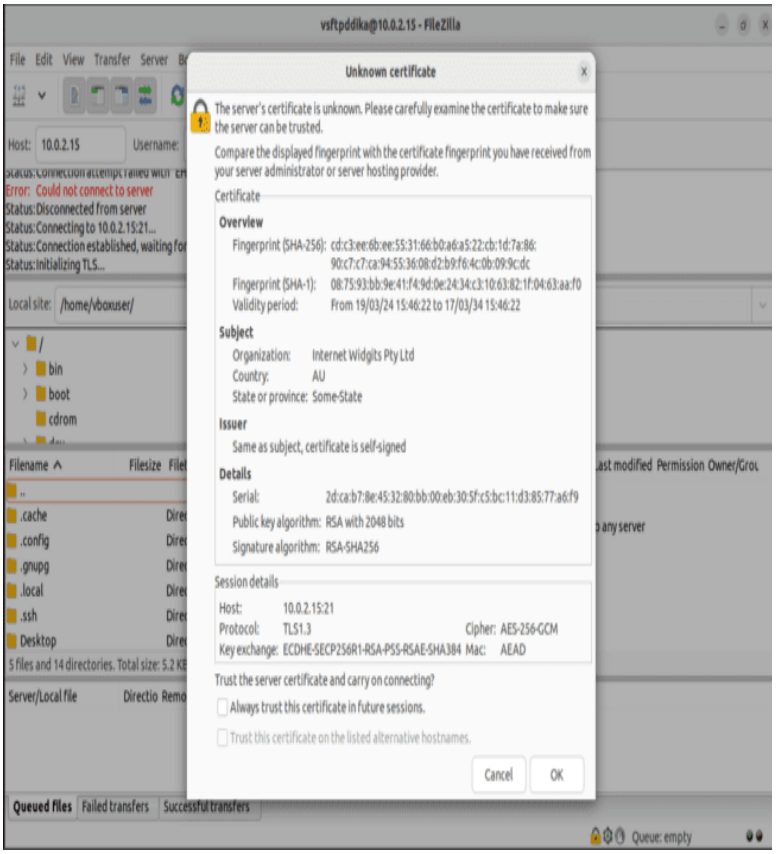


Gbr 35. konfigurasi *certificates*

Gbr 35 merupakan gambar untuk konfigurasi *certificates,* hapus tanda "#" didepan rsa_. Dan ubah ssl_enable menjadi YES untuk penggunaan SSL, yang akan mencegah klien yang tidak dapat menangani TLS untuk terhubung. Hal ini diperlukan untuk memastikan bahwa semua lalu lintas dienkripsi.

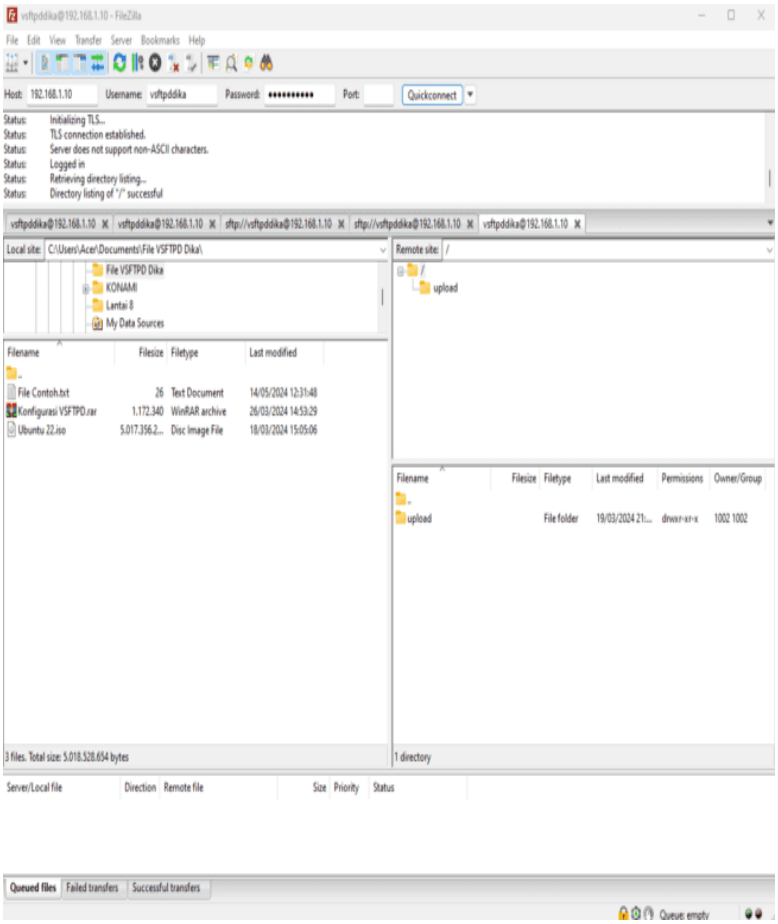
Gbr 36. *Setup* SSL atau TSL VSFTPD

Gbr 36 merupakan gambar untuk *setup* SSL / TSL VSFTPD berguna untuk menolak koneksi anonim melalui SSL dan mewajibkan SSL untuk transfer data dan *login*.



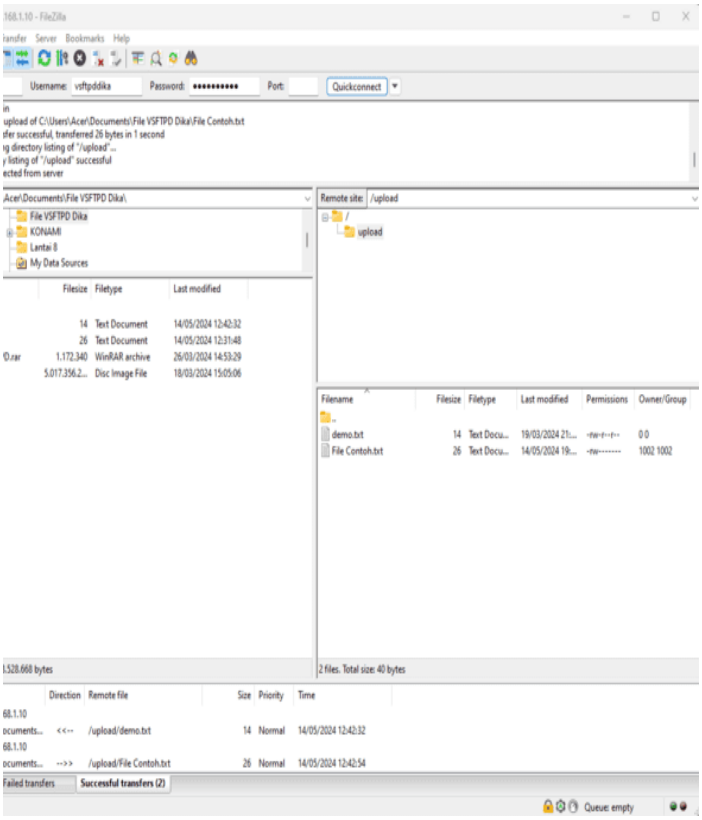
Gbr 37. Tes *login filezilla* dengan TLS/SSL *encryption*

Gbr 37 merupakan gambar untuk tes *login* di *filezilla* dengan TLS/SSL *encryption*.



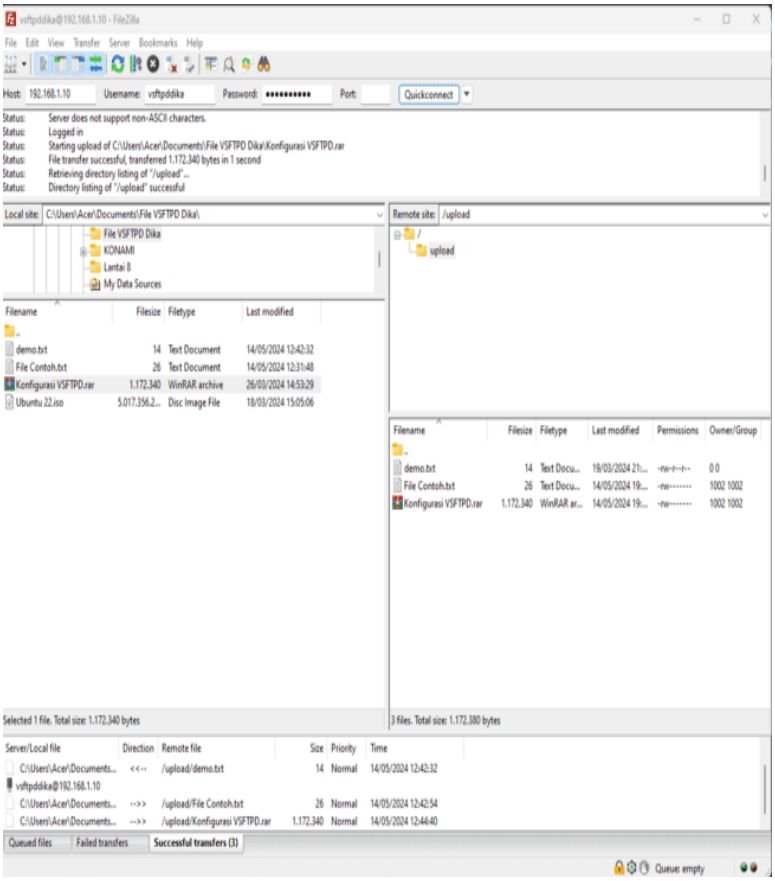
Gbr 38. Login *filezilla* di komputer *client*

Gbr 38. merupakan gambar untuk *login filezilla* di komputer *client* untuk tes transfer *file*.



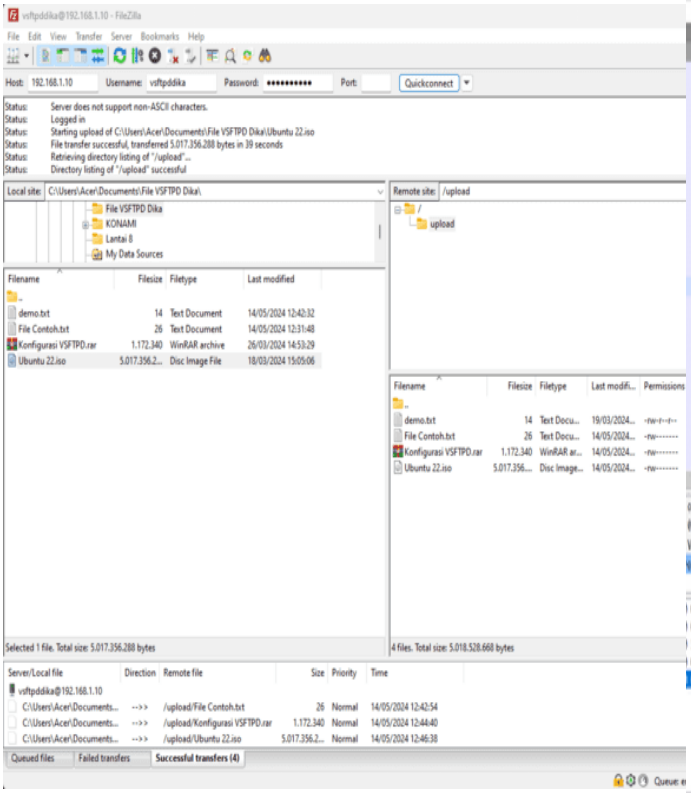
Gbr 39. Tes *transfer file* 1 KB

Gbr 39 merupakan gambar tes *transfer file* dengan ukuran 1 KB.



Gbr 40. Tes *transfer file* 1 MB

Gbr 40 merupakan gambar tes *transfer file* dengan ukuran 1 MB.



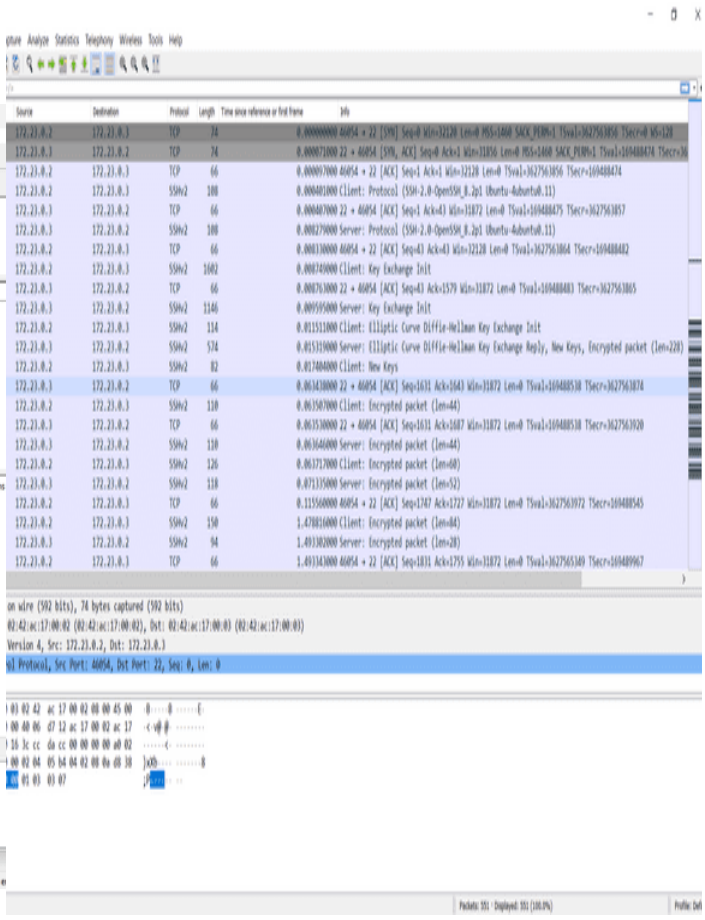
Gbr 41. Tes *transfer file* 5 GB

Gbr 41 merupakan gambar tes *transfer file* dengan ukuran 5 GB.

TABEL II
HASIL PENGUJIAN PERFORMA VSFTPD DENGAN VIRTUAL BOX

No.	Ukuran file (bytes)	Waktu Tempuh (hh:ss)	Rata-rata per second
1	26	00:01	0.026 KBps
2	1.172.340	00:03	390 KBps
3	5.017.356.288	**No.**	128.650 KBps

Tabel II merupakan gambar rincian pengujian performa VSFTPD dengan menggunakan virtual box. Untuk ukuran file 26 b dengan kecepatan 00:01 dengan rata-rata 0.026 KBps. Untuk ukuran file 1.172.340 b dengan kecepatan 00:03 dengan rata-rata 390 KBps. Lalu untuk ukuran file 5.017.356.288 b dengan kecepatan 00:39. dengan rata-rata 128.650 KBps.

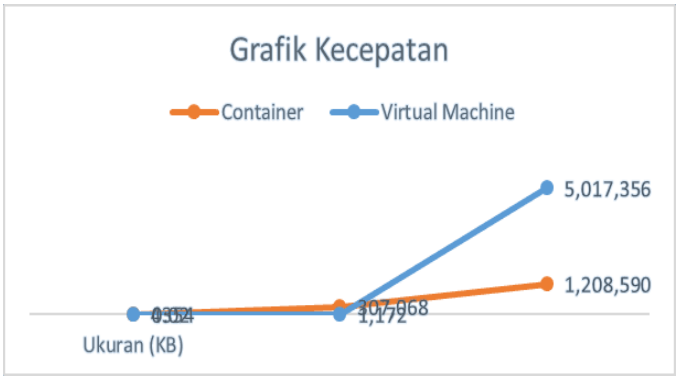


Gbr 42. Capture Wireshark pada SFTP

Gbr 42 merupakan gambar capture wireshark pada SFTP.

C. Perbandingan Grafik

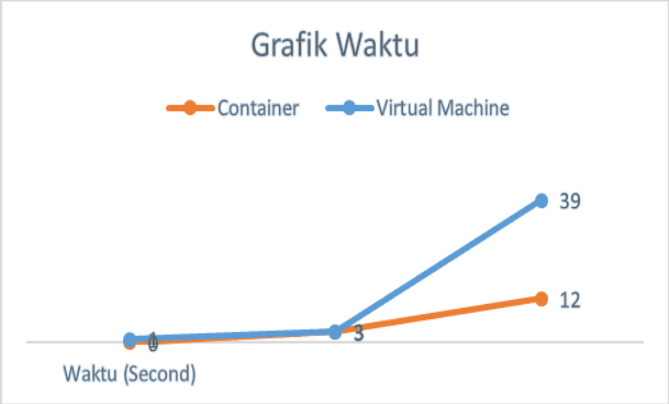
1. Grafik Kecepatan



Gbr 43. Grafik Kecepatan

Gbr 43 merupakan gambar dari grafik perbandingan kecepatan pada tes transfer file menggunakan Container dan juga Virtual Machine.

2. Grafik Waktu



Gbr 44. Grafik Waktu

Gbr 44 merupakan gambar dari grafik perbandingan waktu tempuh pada tes transfer file menggunakan Container dan juga Virtual Machine.

IV. KESIMPULAN

Dari hasil pengujian yang telah dilakukan, penulis me ngambil beberapa kesimpulan yaitu:

1. Hasil pengujian 8 kategori *fashion* yang terdiri dari *Training Data* sebanyak 6000 (750 data per kategori) dan *Test Data* sebanyak 1000 (125 data per kategori), menghasilkan akurasi sebesar 86,42.
2. Hasil penerapan algoritma CNN dapat diimplementasikan dalam sistem rekomendasi pada aplikasi *e-commerce* GOLS dengan menu fitur *search by image* menjadi *text*.

V. SARAN

Adapun saran untuk pengembangan pada penelitian ini supaya menjadi lebih baik lagi, terdapat beberapa saran yaitu:

1. Penambahan variasi *dataset* yang memiliki beberapa aspek. Aspek yang pertama dari penambahan kuantitas *dataset*, hal ini dapat dilakukan untuk menambah akurasi akhir pengujian. Aspek yang kedua yaitu berupa penambahan pada kategori yang bisa diimplementasikan lebih luas pada jenis *fashion.*
2. Memberian variasi tambahan pada aplikasi e-commerce GOLS dengan menu mengambil foto secara langsung (tidak import picture).

UCAPAN TERIMA KASIH

Peneliti senantiasa mengucap rasa syukur yang sangat besar kepada Tuhan YME atas segala berkah, rahmat & juga pertolonganNya, sehingga peneliti mampu menyelesaikan pro yek & artikel ilmiah ini dengan baik. Terimakasih penuliss ucapkan juga kepada Dosen Pembimbing yang telah memberi arahan dan masukan yang sangat bermanfaat buat peneliti. Kedua orang tua peneliti yang tak henti-hentinya mendoakan dan membantu dalam segala hal. Semua saudara-saudara Jurusan Teknik Informatika Angkatan 2017 terutama kelas TI 2017 B yang selalu memberikan dukungan & semangat se hingga peneliti dapat menyelesaikan skripsi ini dengan lancar. Inez Aurellia yang selalu menemani dan membantu lahir dan batin menguatkan dan menyemangati peneliti untuk menyelesaikan skripsi ini.

REFERENSI

[1] Bindu Dodiya, U. Kumar Singh, PhD. (2022). "Malicious Traffic analysis using Wireshark by collection of Indicators of Compromise". International Journal of Computer Applications Vol. 183, No. 53.

[2] Boy Rizky A., M. Iqbal, Arpan (2022). "*Design and Build Data Transfer Process Security on File Transfer Protocol (Ftp) Servers Using Transport Layer Security (Tls) Protocol*". Jurnal Mantik. Vol. 6 No. 2, 2664-2675.

[3] Sakti B., Aziz, A., & Doewes, A. (2013). "Uji Kelayakan Implementasi SSH sebagai Pengaman FTP Server dengan Penetration Testing". ITSMART: Jurnal Teknologi dan Informasi, 2(1), 44-51.

[4] Billy Susanto P., Chandra Wijaya (2019). "*Data Transmission Performance Analysis on the Usage of ZLib Compression over Secure Shell Protocol*". IJTB | *International Journal of Technology And Business*.

[5] Shinta, A. (2022). "Pengertian SFTP, Manfaat dan Cara Menggunakannya". [Online]. Tersedia: https://www.dewaweb.com/blog/pengertian-sftp/. [27 Desember 2023].

[6] Saleh Dwiyatno, Edy Rakhmat & Oki Gustiawan. (2020). "Implementasi Virtualisasi Server Berbasis Docker Container". Jurnal PROSISKO. Vol. 7, No.2.

[7] Deni Marta, M. Angga Eka Putra & Guntoro Barovih. (2019). "Analisis Perbandingan Performa Virtualisasi Server Sebagai Basis Layanan Infrastructure As A Service Pada Jaringan Cloud". Jurnal MATRIK. Vol. 19, No.1, Hal 1-8.

[8] Ahmad Musa. (2020). "Forensic Analysis of Peer-to-Peer Network Traffic with Wireshark". Sule Lamido University Journal of Science and Technology (SLUJST) Vol. 1, No. 2, pp. 92-99.

[9] Putra, C. A. (2012). Mengenal Ubuntu Server. [Online]. Tersedia :www.candra.web.id/mengenal-ubuntu-server/. [28 Desember 2023].

[10] Kovačević, A. (2021). How Does SFTP Work?. [Online]. Tersedia: https://phoenixnap.com/kb/how-f3s-sftp-work. [28 Desember 2023].

[11] Prama Wira G., Galih Putra K., Edi Kusuma N. (2013). "Implementasi Tools Network Mapper Pada Lokal Area Network (Lan)". Jurnal Media Infotama. *Vol.9, No.2.*

[12] Bik, R., & Fadlulloh, M. (2017). "Implementasi Docker Untuk Pengelolaan Banyak Aplikasi Web (Studi Kasus: Jurusan Teknik Informatika Unesa)". Jurnal Manajemen Informatika, Vol. 7 (No.2), 46-50.

[13] Rakhmi, Adi, dan Siti (2019). "Teknis Kerja Docker Container untuk Optimalisasi Penyebaran Aplikasi". Jurnal Penelitian Ilmu Komputer, *System Embedded & Logic*, Vol. 7 (No. 2), 167-176.