

# MAHIR VIRTUAL PRIVATE SERVER

Edisi PHP Developer



Musa Amin



# **MAHIR VIRTUAL PRIVATE SERVER**

Edisi PHP Developer

Musa Amin

**BITNESIA**

2 Januari 2024

# DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR GAMBAR.....	v
DAFTAR TABEL.....	viii
PRAKATA.....	ix
BAB 1 SISTEM OPERASI LINUX.....	1
Distribusi Linux.....	2
Ubuntu.....	3
Filesystem Hierarchy Standard (FHS).....	5
Shell.....	7
Shell Prompt.....	8
Perintah Dasar Linux.....	9
Manajemen User.....	22
Membuat User.....	22
Memberi Akses sudo.....	22
Melepas Akses sudo.....	23
Mengubah Password.....	23
Menghapus User.....	23
Kepemilikan dan Izin Akses File.....	24
Ownership.....	24
Permissions.....	24
Menampilkan Kepemilikan dan Izin Akses.....	24
Mengubah Kepemilikan.....	26
Mengubah Izin Akses.....	27
Manajemen Paket apt.....	28
Perintah apt.....	29
Text Editor.....	29
vim - Vi IMproved, a programmer's text editor.....	29
nano - Nano's ANOther editor.....	31
Shell Script.....	32
Membuat Shell Script.....	32
BAB 2 VIRTUAL PRIVATE SERVER (VPS).....	34
Kapan Memakai VPS?.....	34
Tanggung Jawab Pengguna.....	35
Keterampilan yang Dibutuhkan.....	36
Membeli VPS.....	36
Menguji Kinerja (Benchmark) VPS.....	37
BAB 3 DOMAIN NAME SYSTEM (DNS).....	40
Authoritative DNS server.....	40
Cloudflare DNS.....	41
Cara Kerja Cloudflare.....	41
Cara Menggunakan Cloudflare DNS.....	43
DNS Records.....	46

Tips DNS.....	47
BAB 4 REMOTE SERVER.....	49
Remote VPS di Windows.....	50
Remote VPS di Linux/macOS.....	53
BAB 5 KONFIGURASI AWAL.....	55
Mengatur Time Zone.....	55
Update dan Upgrade.....	56
Menambah User.....	57
Mengonfigurasi SSH Server.....	60
Mengubah Nomor Port.....	60
Memblokir root dari Login SSH.....	63
Mengubah Metode Otentikasi.....	63
Mengaktifkan Ubuntu Pro.....	69
BAB 6 TRANSFER FILE.....	72
SFTP/SCP di Windows.....	72
SCP di Linux/macOS.....	75
rsync di Linux/macOS.....	75
BAB 7 GIT VERSION CONTROL SYSTEM.....	77
Install Git.....	77
Git dengan GitHub.....	77
BAB 8 APACHE WEB SERVER.....	80
Install Apache.....	80
Konfigurasi Virtual Host.....	81
BAB 9 NGINX WEB SERVER.....	84
Install Nginx.....	84
Konfigurasi Server Block.....	85
BAB 10 MARIADB DATABASE.....	88
Install MariaDB.....	88
Mengamankan Instalasi MariaDB.....	89
Mengelola Database.....	90
Mengelola User.....	91
Memindahkan Direktori Data.....	92
Remote Access MariaDB.....	94
Remote Access via TCP/IP.....	95
Remote Access via SSH Tunnel.....	96
Backup Database MariaDB.....	100
Restore Database MariaDB.....	101
BAB 11 POSTGRESQL DATABASE.....	102
Install PostgreSQL.....	102
Mengelola Database.....	103
Mengelola User.....	104
Mengonfigurasi Otentikasi.....	104
Memindahkan Direktori Data.....	105
Remote Access PostgreSQL.....	106
Remote Access via TCP/IP.....	107
Remote Access via SSH Tunnel.....	108

Backup Database PostgreSQL.....	111
Restore Database PostgreSQL.....	111
BAB 12 PHP HYPERTEXT PREPROCESSOR.....	112
Install PHP.....	112
PHP dengan Apache.....	113
PHP dengan Nginx.....	113
Pengujian Instalasi PHP.....	115
Konfigurasi PHP.....	115
Multiple PHP Version.....	117
ionCube Loader.....	121
Composer.....	123
Install Composer.....	123
PHP dengan MariaDB.....	124
PHP dengan PostgreSQL.....	125
PHP dengan SQL Server.....	126
Install Driver SQL Server.....	126
Pengujian Driver SQL Server.....	127
BAB 13 PHPMYADMIN.....	128
Install phpMyadmin.....	128
Konfigurasi phpMyAdmin.....	128
phpMyAdmin dengan Apache.....	129
phpMyAdmin dengan Nginx.....	130
BAB 14 PGADMIN.....	131
Install pgAdmin.....	131
BAB 15 SECURE SOCKET LAYER (SSL).....	137
SSL Let's Encrypt.....	137
Install certbot.....	138
Memperbarui SSL Let's Encrypt.....	139
SSL Cloudflare.....	140
Konfigurasi SSL di Apache.....	142
Konfigurasi SSL di Nginx.....	144
BAB 16 TINY FILE MANAGER.....	147
Install Tiny File Manager.....	147
Konfigurasi Tiny File Manager.....	148
BAB 17 WORDPRESS.....	152
System Requirements.....	153
Install WordPress.....	153
BAB 18 CODEIGNITER.....	155
Install CodeIgniter.....	156
BAB 19 LARAVEL.....	158
Install Laravel.....	159
BAB 20 NODE.JS.....	162
Install Node.js.....	163
Install Node.js via Default Repository.....	163
Install Node.js via NodeSource Repository.....	164
Install Node.js via NVM.....	164

BAB 21 MEMCACHED & REDIS.....	167
Memcached.....	167
Install Memcached.....	168
Redis.....	169
Install Redis.....	169
BAB 22 LOG FILES.....	170
Menampilkan Log.....	170
auth.log - Authorization Log.....	171
lastlog - Login Terakhir.....	171
Apache Log File.....	171
Nginx Log File.....	171
BAB 23 SYSTEM MONITORING.....	172
htop - Interactive Process Viewer.....	173
free - Penggunaan Memory.....	174
df - Penggunaan Disk Space.....	175
HetrixTools - Uptime & System Monitoring.....	175
Uptime Monitoring.....	175
Domain dan SSL Monitoring.....	176
Server Monitoring.....	176
Notifikasi ke Telegram.....	177
BAB 24 FIREWALL.....	179
Cloud Firewall.....	179
Konfigurasi Cloud Firewall.....	180
iptables.....	182
Konfigurasi iptables.....	183
Shell Script iptables.....	188
Uncomplicated Firewall (ufw).....	189
Konfigurasi ufw.....	189
Bash script ufw.....	192
Cloudflare Web Application Firewall (WAF).....	193
Fitur Cloudflare WAF.....	193
Konfigurasi Cloudflare WAF.....	194
BAB 25 BACKUP & RESTORE.....	197
Backup Data.....	197
Sebelum Backup Data.....	197
Melakukan Backup Data.....	198
Backup Data ke Object Storage.....	207
Restore Data.....	208
Sebelum Restore Data.....	208
Melakukan Restore Data.....	209
BAB 26 CONTROL PANEL.....	211
CloudPanel.....	211
Install CloudPanel.....	212
Menambahkan PHP Site.....	214

# DAFTAR GAMBAR

Gambar 1: Time line masa dukungan Ubuntu.....	5
Gambar 2: Linux File Systems.....	7
Gambar 3: Contoh hasil perintah 'ls -l'.....	25
Gambar 4: Vim berada dalam Command mode.....	30
Gambar 5: Vim berada dalam Insert mode.....	31
Gambar 6: Nano text editor.....	31
Gambar 7: Contoh informasi akses VPS.....	37
Gambar 8: DNS Management di Penyedia Domain.....	41
Gambar 9: Website dengan Cloudflare vs tanpa Cloudflare.....	42
Gambar 10: Form Add site.....	43
Gambar 11: Free Plan.....	43
Gambar 12: Scan DNS records.....	43
Gambar 13: DNS records hasil scan.....	44
Gambar 14: Informasi perubahan nameserver.....	44
Gambar 15: Mengubah nameserver.....	45
Gambar 16: Tombol konfirmasi pemeriksaan nameserver.....	45
Gambar 17: Menambahkan domain di Cloudflare selesai.....	45
Gambar 18: Contoh DNS records.....	47
Gambar 19: PuTTY Session.....	51
Gambar 20: PuTTY Security Alert.....	52
Gambar 21: PuTTY login ke server.....	52
Gambar 22: PuTTY berhasil login ke server.....	53
Gambar 23: Login ke server dengan OpenSSH client.....	54
Gambar 24: OpenSSH client telah login ke server.....	54
Gambar 25: Pesan permintaan restart.....	57
Gambar 26: Hasil perintah netstat.....	61
Gambar 27: Konfigurasi port SSH server.....	62
Gambar 28: Status SSH service.....	62
Gambar 29: Kolom Port di PuTTY.....	63
Gambar 30: PuTTY Key Generator.....	64
Gambar 31: Public key hasil generate PuTTYGen.....	65
Gambar 32: Generate SSH key.....	65
Gambar 33: SSH Public key.....	66
Gambar 34: PuTTY Credentials.....	67
Gambar 35: Otentikasi dengan key di PuTTY.....	68
Gambar 36: Login SSH dengan nama host.....	69
Gambar 37: Token Ubuntu Pro.....	70
Gambar 38: Ubuntu pro attach.....	70
Gambar 39: Ubuntu pro status.....	71
Gambar 40: Membuat Session di WinSCP.....	73
Gambar 41: Pengaturan SSH Authentication di WinSCP.....	74
Gambar 42: WinSCP telah login ke server.....	75

Gambar 43: Menambahkan SSH key di GitHub.....	78
Gambar 44: Apache default page.....	81
Gambar 45: Nginx default page.....	85
Gambar 46: Pemisahan App server dengan DB server.....	94
Gambar 47: SSH tunnel untuk database.....	95
Gambar 48: PuTTYgen - Load private key.....	98
Gambar 49: Membuat MySQL Connection.....	99
Gambar 50: MySQL Workbench - Menampilkan database dan tabel.....	100
Gambar 51: Menampilkan database PostgreSQL.....	103
Gambar 52: pgAdmin - Pengaturan Connection.....	109
Gambar 53: pgAdmin - Pengaturan SSH Tunnel.....	110
Gambar 54: pgAdmin - Menampilkan database dan isi tabel.....	110
Gambar 55: Pesan sukses pengujian sqlsrv dan pdo_sqlsrv.....	127
Gambar 56: Form login pgAdmin 4.....	133
Gambar 57: pgAdmin - Register Server tab General.....	134
Gambar 58: pgAdmin - Register Server tab Connection.....	135
Gambar 59: pgAdmin dashboard.....	135
Gambar 60. Coudflare SSL/TLS encryption mode.....	140
Gambar 61. Origin certificates.....	141
Gambar 62. Generate SSL di Cloudflare.....	141
Gambar 63. Origin Certificate.....	141
Gambar 64. Private Key.....	142
Gambar 65. Password Hash Generator.....	148
Gambar 66. Halaman login Tiny File Manager.....	151
Gambar 67. Direktori /var/www di Tiny File Manager.....	151
Gambar 68: Halaman default CodeIgniter.....	156
Gambar 69: Alur data web app dengan cache.....	167
Gambar 70. Pemakaian CPU dan RAM di htop.....	174
Gambar 71. Process dan pemakaian CPU/RAM di htop.....	174
Gambar 72. HetrixTools Server Monitoring.....	177
Gambar 73. Notifikasi HetrixTools di Telegram.....	178
Gambar 74: Server dengan Network-Based Firewall.....	179
Gambar 75: Cloud Firewall Inbound Rules.....	181
Gambar 76: Cloud Firewall Outbound Rules.....	181
Gambar 77: Server dengan Cloudflare dan Network-Based Firewall.....	182
Gambar 78: Cloud Firewall Inbound Rules dengan Cloudflare.....	182
Gambar 79: Server dengan Host-Based Firewall.....	183
Gambar 80: Server dengan WAF.....	193
Gambar 81: Menu WAF di Cloudflare.....	194
Gambar 82: Custom rule Dasbor Aplikasi.....	195
Gambar 83: Activity log - Matched service.....	196
Gambar 84: Activity log - Request details.....	196
Gambar 85: rclone authorize ke akun Google Drive.....	203
Gambar 86: Isi folder DATA_BACKUP_APP_SERVER di Google Drive.....	207
Gambar 87: Membuat akun admin CloudPanel.....	213
Gambar 88: CloudPanel Dashboard.....	214



Gambar 89: Menambahkan PHP Site.....215

# DAFTAR TABEL

Tabel 1: Direktori di Linux.....	5
Tabel 2. Izin akses Numeric Mode.....	27
Tabel 3. Operator izin akses dengan Symbolic Mode.....	28
Tabel 4. Karakter untuk izin akses Symbolic Mode.....	28
Tabel 5. Tombol perintah yang sering digunakan di nano.....	32
Tabel 6: Daftar Aplikasi dan Versi PHP.....	117
Tabel 7: Cloud Firewall Input Rules tanpa Cloudflare.....	180
Tabel 8: Penjelasan Konfigurasi iptables.....	184
Tabel 9: Daftar File Shell Script iptables.....	189
Tabel 10: Daftar File Bash Script ufw.....	192

# PRAKATA

Alhamdulillah, puji syukur kepada Allah Swt atas selesainya penulisan ebook *Mahir Virtual Private Server Edisi PHP Developer*. Virtual Private Server (VPS) kini semakin sering digunakan sebagai server untuk menjalankan aplikasi PHP yang sebelumnya masih menggunakan shared hosting. Namun, keterampilan dalam mengelola VPS masih sering menjadi kendala dalam proses deployment aplikasi ke server.

Berdasarkan adanya permasalahan tersebut, ebook ini lahir sebagai panduan bagi PHP developer yang ingin mempelajari keterampilan dalam mengelola server dalam bentuk VPS. Meskipun ebook ini ditujukan khusus bagi PHP developer, tetapi ebook ini juga dapat dipelajari oleh orang-orang yang ingin memulai mempelajari keterampilan Linux system administration.

Topik pembahasan di dalam ebook ini mencakup berbagai aspek yang relevan dengan pengelolaan dan konfigurasi VPS untuk aplikasi PHP. Mulai dari pengenalan sistem operasi Linux dan perintah dasarnya, bagaimana cara mengakses server dan konfigurasi SSH server, penggunaan layanan DNS Cloudflare, instalasi dan konfigurasi Apache dan Nginx web server, konfigurasi SSL/TLS untuk HTTPS, instalasi dan manajemen MariaDB database, instalasi dan konfigurasi PHP, instalasi aplikasi dan PHP framework, hingga system monitoring, mengamankan server dengan firewall dan backup data secara otomatis.

Semoga ebook *Mahir Virtual Private Server Edisi PHP Developer* ini dapat menjadi acuan yang bermanfaat bagi para PHP developer dan orang-orang yang ingin mempelajari pengelolaan VPS.

Pontianak, Juli 2023

Penulis

Musa Amin

# BAB 1

## SISTEM OPERASI LINUX

Sistem operasi Linux adalah sistem operasi open source yang didesain berdasarkan kernel Linux. Linux diciptakan oleh seorang mahasiswa Finlandia, Linus Torvalds, pada tahun 1991. Sistem operasi ini telah menjadi salah satu sistem operasi yang paling populer dan digunakan di seluruh dunia, baik di lingkungan server maupun desktop.

Berikut adalah beberapa kelebihan sistem operasi Linux:

1. Open source: Linux adalah sistem operasi open source yang memungkinkan pengguna untuk mengakses kode sumbernya dan mengembangkannya. Hal ini membuat Linux lebih fleksibel dan dapat disesuaikan dengan kebutuhan pengguna.
2. Gratis: Linux tidak memerlukan biaya lisensi seperti sistem operasi komersial lainnya. Hal ini menjadikannya pilihan yang ekonomis untuk organisasi dan individu.
3. Stabil dan andal: Linux dikenal sebagai sistem operasi yang stabil dan andal. Linux dapat berjalan tanpa terganggu selama berbulan-bulan bahkan tahunan. Sistem operasi ini juga dapat digunakan pada mesin yang memiliki spesifikasi hardware yang rendah.
4. Aman: Linux memiliki fitur keamanan yang cukup kuat, termasuk hak akses yang berbeda-beda, firewall, dan enkripsi file. Selain itu, karena kode sumber Linux dapat diakses oleh siapa saja, bug dan kerentanan keamanan dapat ditemukan dan diperbaiki dengan cepat.
5. Banyak distro tersedia: Terdapat banyak varian Linux yang dikenal sebagai distro. Setiap distro memiliki kelebihan dan kekurangan sendiri. Pengguna dapat memilih distro yang paling sesuai dengan kebutuhan mereka.
6. Kompatibel dengan banyak hardware: Linux dapat berjalan pada banyak jenis hardware, termasuk komputer desktop, server, router, dan perangkat embedded. Hal ini menjadikan Linux pilihan yang populer untuk penggunaan di perangkat internet of things (IoT).

7. Banyak aplikasi gratis dan open source tersedia: Linux memiliki banyak aplikasi open source yang tersedia untuk digunakan, seperti Apache untuk web server, MySQL untuk database server, dan LibreOffice untuk suite office. Selain itu, Linux juga mendukung banyak bahasa pemrograman, sehingga cocok digunakan untuk pengembangan perangkat lunak.
8. Fleksibilitas: Linux dapat disesuaikan dengan kebutuhan pengguna. Pengguna dapat menyesuaikan antarmuka, instalasi, konfigurasi, dan pengaturan sistem operasi dengan cara yang berbeda-beda.

Dalam keseluruhan, kelebihan sistem operasi Linux meliputi kemampuan untuk disesuaikan, stabilitas, keamanan, fleksibilitas, dan banyak aplikasi gratis dan open source yang tersedia. Linux juga cocok digunakan pada banyak jenis hardware dan mendukung banyak bahasa pemrograman. Hal ini menjadikannya pilihan yang populer untuk penggunaan di lingkungan server, desktop, dan embedded systems.

## **Distribusi Linux**

Distribusi Linux atau yang juga dikenal sebagai distro adalah varian atau bentuk dari sistem operasi Linux yang disesuaikan dan dikembangkan oleh komunitas atau organisasi tertentu. Distribusi Linux ini biasanya terdiri dari kernel Linux, sistem manajemen paket, antarmuka pengguna, dan aplikasi-aplikasi pendukung.

Setiap distribusi Linux memiliki ciri khas, fitur, dan kelebihan yang berbeda-beda, sehingga pengguna dapat memilih distribusi yang paling cocok dengan kebutuhan dan preferensi mereka. Beberapa distribusi Linux yang populer dan sering digunakan di antaranya:

1. Ubuntu: Ubuntu adalah distribusi Linux yang sangat populer dan mudah digunakan. Ubuntu dikembangkan oleh perusahaan Canonical Ltd. dan didasarkan pada Debian. Ubuntu menyediakan berbagai aplikasi populer, termasuk LibreOffice, Firefox, dan Thunderbird.
2. Debian: Debian adalah distribusi Linux yang stabil dan handal. Debian didasarkan pada kernel Linux dan dikembangkan oleh komunitas sukarelawan. Debian menyediakan lebih dari 50.000 paket perangkat lunak gratis dan open source.
3. Fedora: Fedora adalah distribusi Linux yang dikembangkan oleh komunitas dan disponsori oleh Red Hat. Fedora sering digunakan

sebagai platform pengembangan perangkat lunak dan menyediakan berbagai aplikasi dan alat-alat untuk pengembangan perangkat lunak.

4. CentOS: CentOS adalah distribusi Linux yang didasarkan pada kode sumber Red Hat Enterprise Linux (RHEL). CentOS dikembangkan oleh komunitas dan sering digunakan sebagai sistem operasi server.
5. Arch Linux: Arch Linux adalah distribusi Linux yang dirancang untuk pengguna yang lebih berpengalaman. Arch Linux menyediakan kontrol penuh atas instalasi dan konfigurasi sistem operasi, sehingga cocok digunakan oleh pengguna yang ingin mengoptimalkan kinerja sistem operasi mereka.
6. OpenSUSE: OpenSUSE adalah distribusi Linux yang didasarkan pada SUSE Linux Enterprise. OpenSUSE menyediakan berbagai aplikasi dan alat-alat untuk pengembangan perangkat lunak dan sering digunakan sebagai sistem operasi server dan desktop.

Setiap distribusi Linux memiliki karakteristik, fitur, dan dukungan yang berbeda-beda, sehingga penting untuk memilih distribusi yang paling sesuai dengan kebutuhan pengguna. Biasanya, distribusi Linux dapat diunduh dan diinstal dengan bebas, dan pengguna juga dapat memilih untuk berkontribusi pada pengembangan distribusi Linux melalui komunitas pengembang.

## Ubuntu

Ubuntu adalah salah satu distribusi Linux paling populer dan paling banyak digunakan di dunia. Ubuntu dikembangkan dan didistribusikan oleh Canonical Ltd., sebuah perusahaan swasta yang didirikan oleh pengembang Linux asal Afrika Selatan, Mark Shuttleworth.

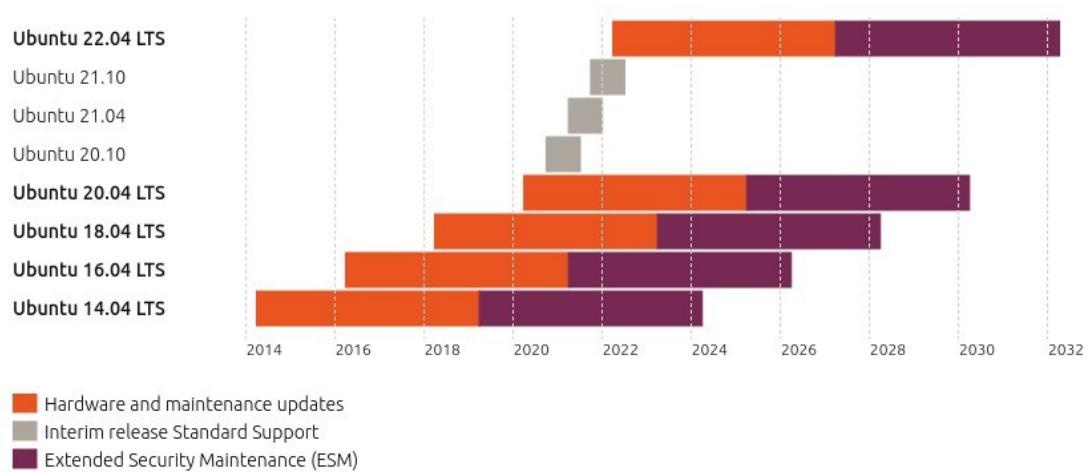
Ubuntu dirancang untuk mudah digunakan oleh pengguna dengan tingkat keahlian yang berbeda-beda, baik untuk pengguna rumahan maupun untuk pengguna bisnis. Salah satu kelebihan Ubuntu adalah kemudahan instalasi dan konfigurasi, serta adanya dukungan komunitas yang besar dan aktif.

Ubuntu memiliki beberapa varian dengan perbedaan di sisi desktop environment dan aplikasi khas desktop-nya yaitu Kubuntu, Lubuntu, Xubuntu, Ubuntu Budgie, Ubuntu Kylin, Ubuntu Mate, dan Ubuntu Studio. Ubuntu juga dijadikan basis distribusi induk oleh beberapa distribusi Linux lain untuk desktop seperti Linux Mint, elementary OS, Zorin OS, KDE Neon, Linux Lite, dan Pop! OS.

Untuk kebutuhan sistem operasi server, Ubuntu merupakan pilihan yang tepat untuk server karena memiliki beberapa kelebihan dan fitur yang dapat mendukung kebutuhan bisnis dan teknis pada penggunaan server. Berikut adalah beberapa alasan mengapa Ubuntu cocok untuk digunakan sebagai sistem operasi server:

1. Keamanan yang baik: Ubuntu dilengkapi dengan fitur keamanan yang baik, seperti firewall, pembaruan keamanan otomatis, dan manajemen pengguna yang ketat. Selain itu, Ubuntu juga terus menerus melakukan pembaruan keamanan dan memberikan dukungan jangka panjang (long-term support) untuk setiap rilisnya.
2. Stabilitas dan reliabilitas: Ubuntu didesain dengan fokus pada stabilitas dan kehandalan, sehingga cocok untuk digunakan sebagai sistem operasi server yang harus berjalan selama 24 jam sehari dan 7 hari seminggu.
3. Kemudahan konfigurasi dan manajemen: Ubuntu memiliki antarmuka pengguna yang mudah digunakan dan dapat dikonfigurasi dengan mudah, sehingga pengguna dapat dengan mudah mengelola server mereka tanpa memerlukan pengetahuan teknis yang mendalam.
4. Komunitas pengguna dan pengembang yang besar dan aktif: Ubuntu memiliki komunitas pengguna dan pengembang yang besar dan aktif, sehingga pengguna dapat dengan mudah mencari bantuan atau solusi atas masalah yang dihadapi.
5. Dukungan perangkat keras yang luas: Ubuntu mendukung berbagai jenis perangkat keras dan memiliki driver yang baik, sehingga mudah untuk memasang Ubuntu pada server dan memastikan kompatibilitas perangkat keras.
6. Repositori paket yang luas: Ubuntu memiliki repositori paket yang sangat luas dan terus diperbarui, yang menyediakan berbagai aplikasi dan perangkat lunak open source, sehingga memudahkan pengguna untuk menginstal dan memperbarui perangkat lunak pada server mereka.
7. Tersedia dalam versi LTS (Long-Term Support): Ubuntu menyediakan versi LTS (Long-Term Support) yang memberikan dukungan jangka panjang (hingga 10 tahun) dan pembaruan keamanan reguler, sehingga cocok untuk digunakan pada server yang memerlukan stabilitas dan keamanan jangka panjang.

Dengan kelebihan dan fitur-fitur tersebut, Ubuntu merupakan pilihan yang tepat untuk digunakan sebagai sistem operasi server.



Gambar 1: Time line masa dukungan Ubuntu

## Filesystem Hierarchy Standard (FHS)

Struktur direktori pada sistem operasi Linux, termasuk Ubuntu, didasarkan pada Filesystem Hierarchy Standard (FHS) yang merupakan standar untuk organisasi struktur direktori pada sistem operasi Unix-like. Struktur direktori pada sistem operasi Linux terdiri dari beberapa direktori utama dan sub-direktori, di antaranya adalah:

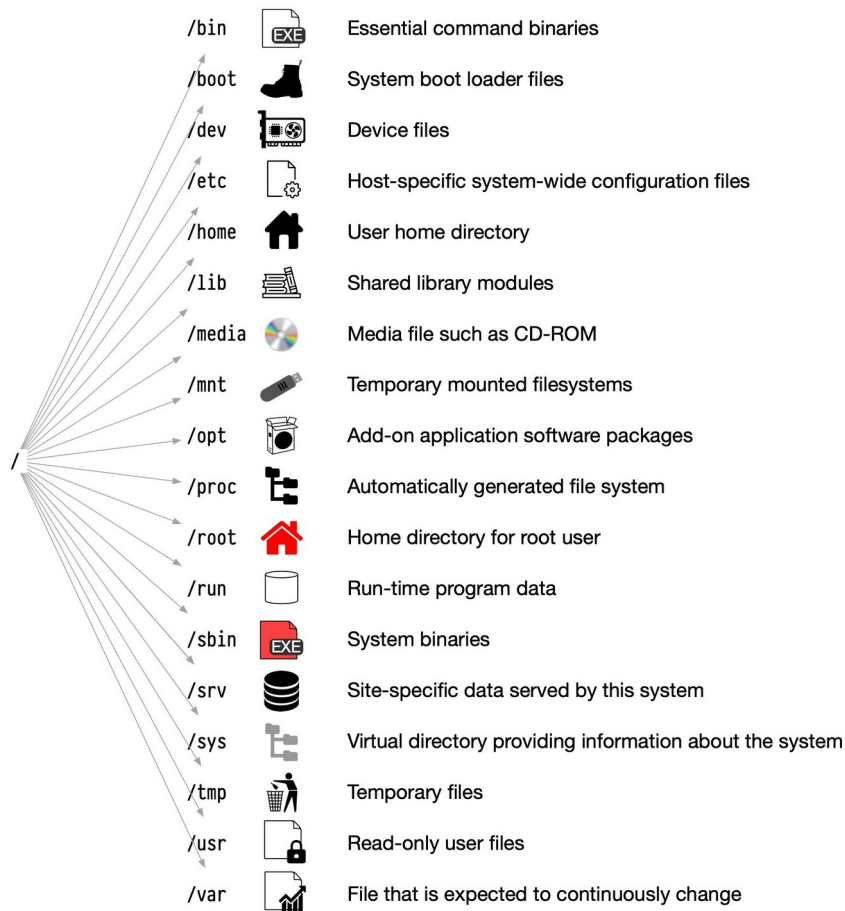
Tabel 1: Direktori di Linux

Direktori	Deskripsi
/ (dibaca root)	Direktori root adalah direktori utama pada sistem operasi Linux. Semua direktori lainnya berada di bawah direktori root.
/bin	Direktori bin berisi file biner (executable) yang diperlukan oleh sistem dan pengguna.
/boot	Direktori boot berisi file yang diperlukan saat booting sistem operasi, seperti kernel, bootloader, dan konfigurasi boot.
/dev	Direktori dev berisi file perangkat (device files) yang merepresentasikan perangkat hardware dan virtual, seperti hard drive, keyboard, mouse, dan printer.
/etc	Direktori etc berisi file konfigurasi sistem dan aplikasi.
/home	Direktori home berisi direktori pengguna (user directories) yang berisi data setiap



Direktori	Deskripsi
	pengguna.
/lib	Direktori lib berisi library (shared object files) yang diperlukan oleh sistem dan aplikasi.
/media	Direktori media berisi direktori mount (mount points) yang digunakan untuk mengakses file yang disimpan pada perangkat penyimpanan eksternal, seperti USB drive dan CD-ROM.
/mnt	Direktori mnt berisi direktori mount (mount points) yang digunakan untuk mengakses file yang disimpan pada perangkat penyimpanan internal, seperti hard drive dan SSD.
/opt	Direktori opt berisi aplikasi tambahan yang tidak terkait dengan sistem operasi.
/proc	Direktori proc berisi file virtual yang merepresentasikan proses dan pengaturan sistem operasi.
/root	Direktori root adalah direktori home untuk pengguna root.
/run	Direktori run berisi file temporary (temporary files) yang dibuat saat booting sistem dan dihapus saat shutdown.
/sbin	Direktori sbin berisi file biner (executable) yang hanya dapat diakses oleh pengguna root dan digunakan untuk mengelola sistem.
/srv	Direktori srv berisi data yang disediakan oleh server.
/sys	Direktori sys berisi file virtual yang merepresentasikan perangkat dan konfigurasi kernel.
/tmp	Direktori tmp berisi file temporary (temporary files) yang dapat dihapus secara otomatis oleh sistem.
/usr	Direktori usr berisi file dan direktori yang terkait dengan aplikasi dan pengguna.
/var	Direktori var berisi file log, file temporary (temporary files), dan data variabel (variable data) yang dihasilkan oleh sistem dan aplikasi.

Informasi filesystem hierarchy dapat dibaca secara lengkap melalui perintah `man hier`. Perintah `man hier` akan menampilkan manual untuk Filesystem Hierarchy Standard (FHS). Pengguna dapat mempelajari lebih lanjut tentang struktur direktori pada sistem operasi Linux dan tujuan dari setiap direktori dan sub-direktori yang ada.



Gambar 2: Linux File Systems

## Shell

Shell pada sistem operasi adalah sebuah program yang berfungsi sebagai antarmuka antara pengguna dengan sistem operasi. Shell memungkinkan pengguna untuk memasukkan perintah-perintah pada command-line interface (CLI) atau terminal dan menjalankan aplikasi pada sistem operasi.

Shell pada sistem operasi Linux atau Unix terdiri dari beberapa jenis, namun yang paling umum digunakan adalah Bash (Bourne-again shell) yang disediakan secara default pada hampir semua distribusi Linux termasuk Ubuntu. Bash adalah sebuah shell open-source yang dikembangkan oleh Brian Fox untuk sistem operasi GNU (GNU's Not Unix) yang kompatibel dengan shell sh (Bourne shell) pada Unix.

Selain Bash, terdapat juga ksh (Korn shell), csh (C shell), dan tcsh (Tenex C shell) yang tersedia pada beberapa distribusi Linux. Setiap jenis shell memiliki karakteristik dan fitur yang berbeda, namun pada dasarnya memiliki fungsi yang sama yaitu sebagai antarmuka untuk menjalankan perintah-perintah pada sistem operasi.

Shell pada Linux memiliki fitur yang cukup lengkap dan fleksibel, seperti kemampuan untuk menjalankan perintah secara bersamaan (concurrent), melakukan pengolahan teks, manajemen file dan direktori, mengatur lingkungan kerja, dan masih banyak lagi. Selain itu, shell pada Linux juga mendukung penggunaan variabel, fungsi, dan pemrosesan script atau skrip yang berisi serangkaian perintah yang dapat dijalankan secara otomatis pada sistem operasi Linux.

Dalam penggunaannya, shell pada Linux umumnya digunakan oleh administrator sistem atau pengguna tingkat lanjut yang membutuhkan akses ke sistem operasi secara langsung dan lebih efektif. Namun, shell pada Linux juga dapat digunakan oleh pengguna biasa untuk menjalankan perintah-perintah sederhana pada sistem operasi seperti memindahkan atau menghapus file, mengubah hak akses file, atau menjalankan aplikasi pada terminal.

## Shell Prompt

Shell prompt adalah tampilan teks yang muncul pada layar saat kita membuka terminal atau menjalankan shell pada sistem operasi. Shell prompt biasanya berisi informasi seperti nama pengguna, nama host, direktori kerja saat ini, dan karakter prompt yang menandakan bahwa shell siap menerima perintah. Shell prompt dapat diubah sesuai dengan kebutuhan pengguna, seperti mengubah warna atau menambahkan informasi tambahan pada prompt.

Default shell prompt pada bash shell, tertera username, hostname, dan path direktori kerja saat ini dengan karakter prompt "\$" atau "#", tergantung pada apakah pengguna login sebagai pengguna biasa atau sebagai superuser (root). Contoh default shell prompt sebagai pengguna biasa:

```
username@hostname:~$
```

Pada contoh di atas, "username" adalah nama pengguna, "hostname" adalah nama host, "~" adalah alias dari direktori /home/username, direktori kerja saat ini, dan "\$" adalah karakter prompt sebagai user biasa.

Shell prompt ketika login sebagai root:

```
root@hostname:~#
```

## Perintah Dasar Linux

Perintah dasar Linux adalah serangkaian perintah atau command-line interface (CLI) yang paling sering digunakan pada sistem operasi Linux. Perintah-perintah tersebut dapat digunakan untuk melakukan tugas-tugas dasar seperti manajemen file dan direktori, pengolahan teks, pengaturan lingkungan kerja, dan banyak lagi.

### Operasi File dan Direktori

Perintah dasar yang berhubungan dengan operasi file dan direktori:

#### **pwd - Print working directory**

Perintah ini digunakan untuk menampilkan direktori kerja saat ini (current working directory).

Contoh penggunaan perintah `pwd`.

```
user@host:~$ pwd
/home/user/Documents
```

Dengan mengetahui direktori kerja saat ini, pengguna dapat dengan mudah menavigasi ke direktori yang diinginkan.

#### **cd - Change working directory**

Perintah ini digunakan untuk berpindah direktori. Format perintahnya `cd /direktori/tujuan`. Jika langsung menjalankan perintah `cd` tanpa disertai dengan direktori tujuan, hasilnya akan berpindah ke direktori `/home/user`.

Pindah ke direktori `/var/www`.

```
$ cd /var/www
```

Linux mengenal case sensitive, yaitu membedakan huruf kecil dan huruf besar baik itu untuk perintah maupun nama file atau direktori.

Pindah ke direktori sebelumnya.

```
$ cd -
```

Pindah ke direktori induk satu tingkat di atasnya.

```
$ cd ..
```

Pindah ke direktori induk dua tingkat di atasnya.

```
$ cd ../../
```

Tabel 5. Tombol perintah yang sering digunakan di nano

Tombol	Deskripsi
CTRL+G	Membuka halaman petunjuk pemakaian
CTRL+O	Menyimpan file
CTRL+X	Keluar dari nano
CTRL+W	Pencarian teks
CTRL+\	Find dan replace teks
CTRL+C	Cancel
CTRL+Space	Maju satu kata
ALT+Space	Mundur satu kata
CTRL+P	Pindah ke baris sebelumnya
CTRL+N	Pindah ke baris berikutnya
CTRL+V	Pindah ke halaman berikutnya
CTRL+Y	Pindah ke halaman sebelumnya
CTRL+A	Pindah ke awal baris
CTRL+E	Pindah ke akhir baris

## Shell Script

Shell script adalah script atau program yang ditulis dalam bahasa scripting misalnya Bash (Bourne Again Shell) pada sistem operasi Unix-like. Shell script pada umumnya digunakan untuk mengotomatisasi tugas-tugas pada sistem operasi Unix-like.

Dengan menulis serangkaian perintah dan logika dalam sebuah script, pengguna dapat menjalankan sejumlah tugas atau perintah secara berurutan atau berdasarkan kondisi tertentu. Script tersebut dapat mencakup perintah-perintah shell, pemrosesan teks, pengontrolan aliran eksekusi, variabel, pengulangan, pengambilan input dari pengguna, dan masih banyak lagi.

## Membuat Shell Script

Langkah-langkah dasar dalam membuat shell script:

1. Menentukan tujuan script: tentukan apa yang ingin dicapai dengan bash script tersebut. Apakah itu melakukan pemrosesan file, menjalankan serangkaian perintah, atau tugas lainnya. Misalnya, kita ingin membuat script yang menjalankan perintah update, upgrade, dan instalasi Apache.
2. Membuat file script: membuat file baru dengan menggunakan text editor seperti nano atau vim. File script disarankan disimpan dengan ekstensi `.sh` agar memudahkan dikenali sebagai file shell script.

```
$ nano install.sh
```

3. Menentukan shell: pada baris pertama, tentukan shell yang akan digunakan, yaitu bash. Jadi, baris pertama script:

```
#!/bin/bash
```

4. Menulis kode script: menuliskan script dengan menggunakan sintaks bash. Di sini, kita menuliskan perintah yang menjadi tujuan script ini, yang kita telah ditentukan di tahap pertama.

```
sudo apt update  
sudo apt upgrade -y  
sudo apt install apache2 -y
```

5. Simpan dan keluar: simpan script yang telah ditulis (CTRL+O) dan keluar dari text editor (CTRL+X).
6. Memberikan izin eksekusi: memberikan izin eksekusi ke file script agar dapat dijalankan.

```
$ chmod +x install.sh
```

7. Menguji script: menjalankan script dan memeriksa apakah script berjalan sesuai yang diharapkan dan tujuan tercapai. Jika ada kesalahan, perbaiki dan uji kembali.

```
$ ./install.sh
```

8. Cara lain menjalankan script: bisa juga menjalankan script tanpa memberikan izin eksekusi. Langsung menjalankannya dengan perintah bash.

```
$ bash install.sh
```

9. Modifikasi dan perbaiki: sesuaikan dan perbaiki script sesuai dengan kebutuhan. Kita dapat menambahkan logika tambahan, menambahkan kondisi, atau mengubah perilaku script sesuai dengan kebutuhan.

Tutorial dasar Bash Scripting dari freecodecamp.org → [Bash Scripting Tutorial - Linux Shell Script and Command Line for Beginners](#).

```
john@Laptop: ~$ ssh server
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-72-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue May 30 09:27:31 AM UTC 2023
```

Gambar 36. Login SSH dengan nama host

## Mengaktifkan Ubuntu Pro

Ubuntu Pro adalah layanan langganan keamanan dan pemeliharaan yang ditawarkan oleh Canonical, perusahaan di balik Ubuntu. Layanan ini memberikan dukungan keamanan selama 10 tahun untuk sistem operasi dan lebih dari 23.000 aplikasi, termasuk aplikasi populer seperti Docker, WordPress, dan Node.js.

Ubuntu Pro dirancang untuk organisasi yang memerlukan keamanan dan kepatuhan jangka panjang untuk perangkat lunak mereka. Dengan Ubuntu Pro, perusahaan atau pengguna dapat memastikan bahwa perangkat lunak selalu terbaru dan aman.

Ubuntu Pro tersedia secara gratis untuk penggunaan pribadi hingga 5 PC. Member Ubuntu dapat menggunakannya secara gratis hingga 50 PC. Untuk penggunaan bisnis, langganan standar Ubuntu Pro memiliki biaya \$25/tahun untuk workstation atau \$500/tahun untuk server. Tersedia percobaan gratis selama 30 hari.

### Langkah-Langkah Mengaktifkan Ubuntu Pro

1. Membuat akun terlebih dulu di [login.ubuntu.com](https://login.ubuntu.com).
2. Login ke [dasbor Ubuntu Pro](#).
3. Copy Token yang ditampilkan.

## Free Personal Token

Created	Expires	Billing	Cost
17 Mar 2023	Never	None	Free
Machine type	Machines	Active machines	
Physical	5	1 ⓘ	

## Token

```
🔗 kae4Tisah6iefoom7iowiesesaagh5
```

Gambar 37: Token Ubuntu Pro

4. Lalu attach Ubuntu server ke layanan Ubuntu Pro dengan menjalankan perintah seperti di bawah ini. Ganti TOKEN dengan Token yang telah di-copy.

```
$ sudo pro attach TOKEN
```

Contoh hasil perintahnya.

```
johndoe@server:~$ sudo pro attach C13rYMXD
Enabling default service esm-apps
Updating package lists
Ubuntu Pro: ESM Apps enabled
Enabling default service esm-infra
Updating package lists
Ubuntu Pro: ESM Infra enabled
Enabling default service livepatch
Installing canonical-livepatch snap
Canonical livepatch enabled.
This machine is now attached to 'Ubuntu Pro - free personal subscription'

SERVICE      ENTITLED  STATUS   DESCRIPTION
esm-apps      yes      enabled  Expanded Security Maintenance for Applications
esm-infra     yes      enabled  Expanded Security Maintenance for Infrastructure
livepatch     yes      enabled  Canonical Livepatch service
usg           yes      disabled Security compliance and audit tools

NOTICES
Operation in progress: pro attach

Enable services with: pro enable <service>

Account:
Subscription: Ubuntu Pro - free personal subscription
```

Gambar 38: Ubuntu pro attach

5. Menampilkan status layanan Ubuntu Pro.

```
$ pro security-status
```



Contoh hasil perintahnya.

```
john@server:~$ pro security-status
620 packages installed:
  614 packages from Ubuntu Main/Restricted repository
  1 package from Ubuntu Universe/Multiverse repository
  5 packages no longer available for download

To get more information about the packages, run
  pro security-status --help
for a list of available options.

Main/Restricted packages receive updates with LTS until 2027.

Universe/Multiverse packages are receiving security updates from
Ubuntu Pro with 'esm-apps' enabled until 2032. You have received no security
updates.
```

Gambar 39: Ubuntu pro status

6. Jika ingin menghentikan layanan Ubuntu Pro, jalankan perintah detach.

```
$ sudo pro detach
```

# BAB 7

## GIT VERSION CONTROL SYSTEM

Git adalah version control system yang terdistribusi, yang digunakan untuk melacak perubahan dalam source code perangkat lunak atau proyek pengembangan perangkat lunak. Diciptakan oleh Linus Torvalds pada tahun 2005, Git terkenal karena kecepatan, keandalan, dan dukungan terhadap pengembangan yang terdistribusi.

Git memungkinkan para developer bekerja secara bersama-sama pada proyek-proyek yang sama, baik secara individu maupun secara tim. Dengan Git, setiap perubahan pada source code disimpan dalam sebuah repository, yang merupakan log yang mencatat semua versi dan perubahan. Repository ini dapat disimpan di komputer lokal atau di remote server, yang memungkinkan kolaborasi yang efisien antar developer yang berbeda.

### Install Git

Menginstal Git di server.

```
$ sudo apt install git -y
```

### Git dengan GitHub

GitHub adalah platform Git repository yang menyimpan source code perangkat lunak serta catatan perubahannya. Dengan GitHub, memungkinkan developer dapat berkerja sama dalam tim dan bekerja secara remote.

Untuk dapat mengakses GitHub repository yang sifatnya private melalui git client (git), kita harus melakukan konfigurasi otentikasi terlebih dulu. Metode otentikasi yang digunakan yaitu menggunakan SSH key.

### Konfigurasi SSH Key untuk GitHub

1. Membuat SSH Key di server, misal memberi nama key-git

```
$ ssh-keygen
```

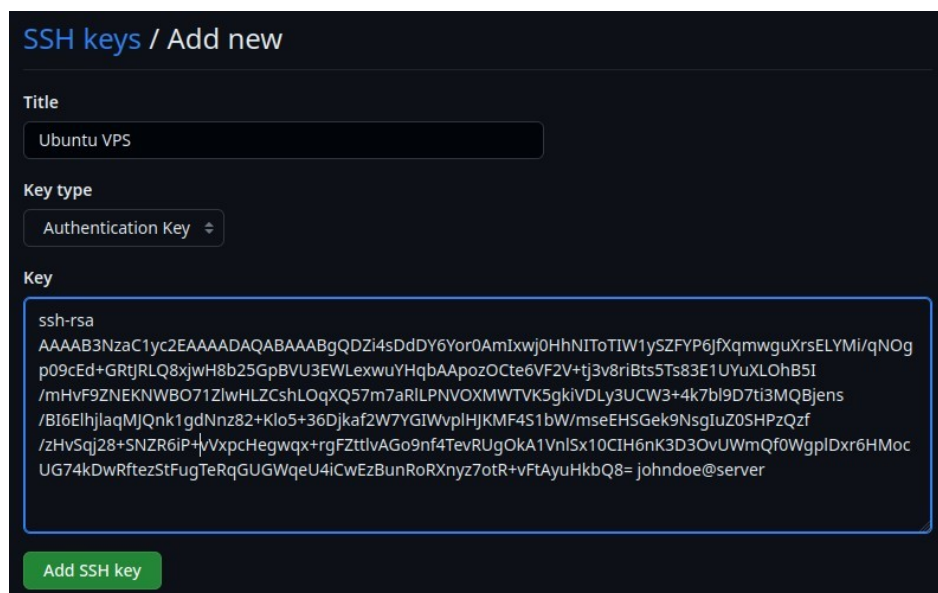
2. Masukkan key telah dibuat ke OpenSSH authentication agent

```
$ eval $(ssh-agent -s)
$ ssh-add ~/.ssh/key-git
```

### 3. Copy public key, key-git.pub

```
$ cat ~/.ssh/key-git.pub
```

4. Buka akun GitHub, klik menu Settings → SSH and GPG Keys
5. Klik New SSH key
6. Masukkan nama key pada kolom Title
7. Lalu paste public key pada kolom Key
8. Lalu klik tombol Add SSH key



Gambar 43: Menambahkan SSH key di GitHub

### 9. Membuat file config SSH client di server

```
$ nano ~/.ssh/config
```

### 10. Masukkan konfigurasi di bawah ini

```
Host github.com-username
Hostname github.com
IdentitiesOnly=yes
IdentityFile /home/johndoe/.ssh/key-git
User git
```

### 11. Menguji otentikasi ke GitHub via SSH

```
$ ssh github.com-username
```

12. Jika sukses, ditampilkan pesan seperti berikut

```
Hi username! You've successfully authenticated, but GitHub does not provide shell access.
```

13. Menguji clone private repository GitHub

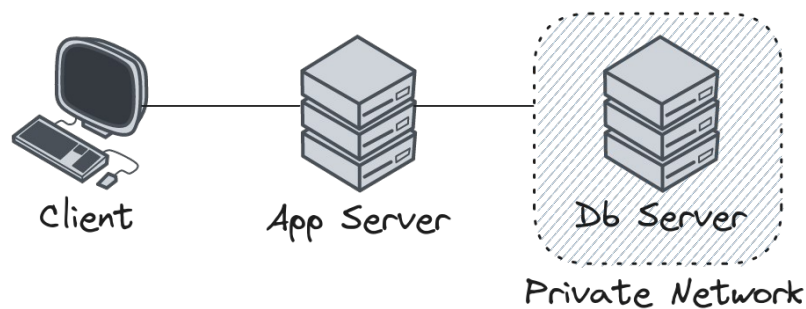
```
$ git clone github.com-username:username/private-repository.git
```

```
select @@datadir;
+-----+
| @@datadir |
+-----+
| /mariadb-data/ |
+-----+
```

## Remote Access MariaDB

Remote access pada MariaDB adalah kemampuan untuk mengakses MariaDB database dari jarak jauh (remote access) melalui jaringan, bukan hanya melalui server itu sendiri.

Contoh kasus penerapan remote access yang sering dilakukan yaitu ketika kita ingin memisahkan sumber daya komputasi antara application server dengan database server, keduanya di-deploy pada server yang terpisah. Database server harus dikonfigurasi agar application server dapat mengakses database secara remote. Pada kasus seperti ini, metode remote access yang diterapkan dengan membuka port MariaDB agar dapat diakses dari luar.



Gambar 46: Pemisahan App server dengan DB server

Contoh kasus lain yaitu developer ingin mengakses database yang berada di server dari komputer yang dipakainya secara aman. Biasanya untuk kebutuhan pengembangan aplikasi atau troubleshooting. Untuk kasus ini, metode remote access yang diterapkan yaitu melalui SSH tunnel.

```
$ sudo systemctl restart nginx
$ sudo systemctl restart php8.2-fpm
```

Memverifikasi apakah module sudah aktif.

```
$ php -m | grep -e sqlsrv -e pdo_sqlsrv
```

Hasilnya.

```
pdo_sqlsrv
sqlsrv
```

## Pengujian Driver SQL Server

Untuk melakukan pengujian ini, sebelumnya telah tersedia database Microsoft SQL Server yang siap untuk diakses.

Menguji modul sqlsrv dengan membuat file pengujian sqlsrv.php.

```
$ sudo nano /var/www/html/sqlsrv.php
```

Copy source code dari [s.id/1Th4R](https://s.id/1Th4R), lalu paste ke sqlsrv.php.

Sementara untuk menguji modul pdo\_sqlsrv, buat file pdo\_sqlsrv.php.

```
$ sudo nano /var/www/html/pdo_sqlsrv.php
```

Lalu copy source code dari [s.id/1Th5L](https://s.id/1Th5L), lalu paste ke pdo\_sqlsrv.php.

Sesuaikan serverName, databaseName, uid, dan pwd dari SQL Server. Setelah itu akses untuk melihat hasil kode programnya, apakah berhasil terhubung ke SQL Server atau tidak.

Jika berhasil, keduanya akan menampilkan pesan seperti di bawah ini.

## Success Results :

```
Microsoft SQL Server 2022 (RTM-CU6) (KB5027505) - 16.0.4055.4 (X64) Jun 29 2023 15:43:13
Copyright (C) 2022 Microsoft Corporation Developer Edition (64-bit) on Linux (Ubuntu 20.04.6 LTS)
```

Gambar 55: Pesan sukses pengujian sqlsrv dan pdo\_sqlsrv

# BAB 13

## PHPMYADMIN

phpMyAdmin adalah sebuah aplikasi web open source yang dirancang untuk mengelola MySQL database atau MariaDB database melalui antarmuka pengguna grafis. phpMyAdmin memungkinkan pengguna untuk melakukan berbagai tugas administrasi database melalui web browser tanpa perlu menggunakan perintah SQL secara langsung di server lewat command line. Dengan menggunakan phpMyAdmin, pengguna dapat membuat, mengedit, dan menghapus basis data, tabel, kolom, serta mengelola pengguna dan hak akses.

### Install phpMyadmin

Download phpMyAdmin versi 5.2.1. Untuk phpMyAdmin versi lain atau yang terbaru download di [phpmyadmin.net](http://phpmyadmin.net).

```
$ wget s.id/1PMQv -O phpmyadmin-v5.2.1.tar.gz
```

Mengekstrak file phpMyAdmin\*.tar.gz.

```
$ tar xzvf phpmyadmin*.tar.gz
```

Memindahkan direktori hasil ekstrak ke /var/www/phpmyadmin.

```
$ sudo mv phpMyAdmin*-english /var/www/phpmyadmin
```

### Konfigurasi phpMyAdmin

Berpindah ke direktori phpmyadmin dan menduplikasi file contoh konfigurasi phpMyAdmin.

```
$ cd /var/www/phpmyadmin
$ sudo cp config.sample.inc.php config.inc.php
```

Membuat direktori tmp untuk phpMyAdmin.

```
$ sudo mkdir tmp
```

Mengubah user dan group direktori phpmyadmin.

```
$ sudo chown -R www-data:www-data /var/www/phpmyadmin
```

Generate random string dengan menggunakan openssl. String yang dihasilkan akan dimasukkan sebagai nilai dari variabel opsi blowfish\_secret.

```
$ openssl rand -base64 24
```

Contoh hasil generate random string. Salin string yang dihasilkan.

```
qlfHc4gzKL8KRB2KBTapOP2Np6STNE5e
```

Membuka file konfigurasi `config.inc.php`.

```
$ sudo nano config.inc.php
```

Paste random string pada variabel `blowfish_secret`.

```
$cfg['blowfish_secret'] = 'qlfHc4gzKL8KRB2KBTapOP2Np6STNE5e';
```

Menonaktifkan akses user root database dengan menambahkan `AllowRoot false` di bawah opsi `AllowPassword`.

```
$cfg['Servers'][$i]['AllowNoPassword'] = false;  
$cfg['Servers'][$i]['AllowRoot'] = false;
```

Kemudian Save dan Exit dari nano.

## phpMyAdmin dengan Apache

URL untuk mengakses phpMyAdmin akan menggunakan IP address dan port misalnya 53006.

Membuat konfigurasi virtual host untuk phpMyAdmin.

```
$ sudo nano /etc/apache2/sites-available/phpmyadmin.conf
```

Masukkan konfigurasi virtual host untuk phpMyAdmin. Ganti `IP_SERVER` dengan Public IP VPS.

```
File: /etc/apache2/sites-available/phpmyadmin.conf
```

```
Listen 53006  
<VirtualHost *:53006>  
    ServerName IP_SERVER  
    DocumentRoot /var/www/phpmyadmin  
    ErrorLog /var/log/apache2/phpmyadmin_error.log  
    CustomLog /var/log/apache2/phpmyadmin_access.log combined  
</VirtualHost>
```

Mengaktifkan konfigurasi virtual host `phpmyadmin.conf`, restart Apache service, dan menampilkan status Apache service.

```
$ sudo a2ensite phpmyadmin.conf  
$ sudo systemctl restart apache2  
$ sudo systemctl status apache2
```

Akses `http://IP_SERVER:53006`, hasilnya halaman login phpMyAdmin.



dengan mudah diakses dan skala dengan baik, tidak peduli seberapa besar situs web kita berkembang.

## System Requirements

System requirements yang direkomendasikan agar dapat menjalankan WordPress dengan baik:

- PHP  $\geq$  7.4
- MySQL  $\geq$  5.7 atau MariaDB  $\geq$  10.3
- Nginx atau Apache dengan mod\_rewrite
- Dukungan HTTPS

## Install WordPress

Diasumsikan sebelum install WordPress, semua persyaratan dan konfigurasi awal telah siap:

- Domain dan telah diarahkan ke IP server, misal domain.com.
- Web server, Apache serta konfigurasi virtual host atau Nginx serta konfigurasi server block untuk domain.com.
- PHP versi minimal 7.4.
- Database MariaDB serta username dan permission.
- HTTPS telah aktif.

Metode instalasi WordPress yang akan diterapkan yaitu melalui command line dengan menggunakan tool WP-CLI. WP-CLI (WordPress Command Line Interface) adalah sebuah command line tool yang digunakan untuk mengelola WordPress. WP-CLI memungkinkan pengguna untuk melakukan tugas-tugas administratif pada WordPress melalui antarmuka command line, yang dapat diakses melalui terminal atau Command Prompt.

### Langkah Install WordPress via WP-CLI

Download WP-CLI. Jika link download melalui short link gagal, download langsung dari [GitHub](#).

```
$ sudo wget s.id/wpcli -O /usr/local/bin/wp
```

Memberikan permission executable.

```
$ sudo chmod +x /usr/local/bin/wp
```

Berpindah ke direktori document root.

```
$ cd /var/www/domain.com
```

Download WordPress core.

```
$ sudo wp core download --allow-root
```

Membuat konfigurasi koneksi ke database sekaligus membuat file konfigurasi WordPress `wp-config.php`.

```
$ sudo wp config create \  
--dbname="domaincom" \  
--dbuser="domaincom" \  
--dbpass="password" \  
--dbhost="localhost" \  
--allow-root
```

Install WordPress dan membuat akun admin.

```
$ sudo wp core install \  
--url="https://www.domain.com" \  
--title="PT.Domain.com" \  
--admin_user="admin" \  
--admin_password="password" \  
--admin_email="me@email.com" \  
--allow-root
```

Pesan yang ditampilkan jika install WordPress berhasil.

```
Success: WordPress installed successfully.
```

Mengubah user dan group file WordPress.

```
$ sudo chown -R www-data:www-data /var/www/domain.com
```

Menguji hasil instalasi dengan mengakses `https://www.domain.com`.

Menyaring log dengan kombinasi grep.

```
# more /var/log/file.log | grep "kata yang dicari"
```

## auth.log - Authorization Log

Log yang merekam aktivitas authorization seperti perintah sudo dan remote login SSH.

```
# tail /var/log/auth.log
```

## lastlog - Login Terakhir

Log yang merekam aktivitas login terakhir. Log file ini dapat ditampilkan dengan menggunakan perintah lastlog.

```
# lastlog
```

## Apache Log File

Apache memiliki dua log file, yang pertama access.log yang merekam setiap aktivitas request dari client. Yang kedua error.log yang merekam setiap error yang terjadi. Jika aplikasi web tidak berjalan sebagaimana mestinya periksa error.log. Setiap virtual host dapat memiliki log file tersendiri, terpisah dengan log file utama dan virtual host lainnya.

```
# tail -f /var/log/apache2/access.log
# tail -f /var/log/apache2/domain.com_access.log
# tail -f /var/log/apache2/error.log
# tail -f /var/log/apache2/domain.com_error.log
```

## Nginx Log File

Sama seperti Apache, Nginx juga memiliki log file untuk access dan error, serta log file masing-masing server block.

```
# tail -f /var/log/nginx/access.log
# tail -f /var/log/nginx/domain.com_access.log
# tail -f /var/log/nginx/error.log
# tail -f /var/log/nginx/domain.com_error.log
```

## BAB 23

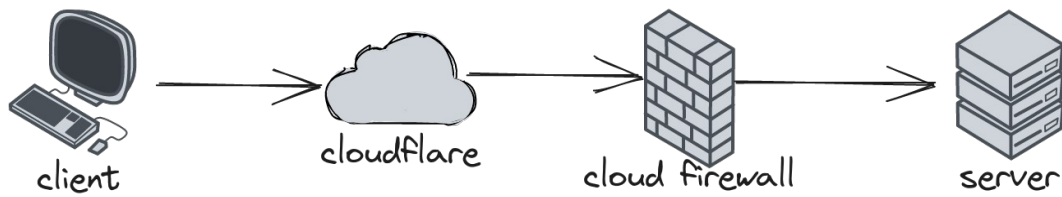
# SYSTEM MONITORING

System monitoring adalah proses pemantauan dan pengawasan terhadap komponen-komponen sistem komputer dalam hal ini adalah server untuk memastikan kinerjanya berjalan dengan baik. Tujuannya adalah untuk mendeteksi masalah atau anomali yang mungkin terjadi di dalam sistem, seperti gangguan kinerja, kegagalan perangkat keras, atau serangan keamanan.

Sistem monitoring dapat mencakup beberapa aspek, termasuk pemantauan ketersediaan sistem, penggunaan sumber daya, kinerja aplikasi, integritas data, dan keamanan.

Manfaat dari system monitoring meliputi:

- **Deteksi dini masalah:** Dengan memantau sistem secara terus-menerus, masalah atau kegagalan dapat dideteksi lebih cepat. Ini memungkinkan tindakan perbaikan diambil sebelum masalah menjadi lebih serius dan berdampak pada pengguna atau operasional bisnis.
- **Optimalisasi kinerja:** System monitoring membantu mengidentifikasi area sistem yang kinerjanya masih dapat ditingkatkan. Informasi yang diperoleh dari pemantauan dapat digunakan untuk mengoptimalkan penggunaan sumber daya, meningkatkan efisiensi, dan mengurangi down time yang tidak perlu.
- **Keamanan sistem:** System monitoring memainkan peran penting dalam deteksi dan pencegahan serangan keamanan. Dengan memantau aktivitas jaringan, log keamanan, dan parameter lainnya, sistem monitoring dapat membantu mengidentifikasi aktivitas yang mencurigakan atau ancaman keamanan yang potensial.
- **Perencanaan kapasitas:** Dengan pemantauan sistem yang teratur, data yang diperoleh dapat digunakan untuk menganalisis tren penggunaan sumber daya. Ini memungkinkan perencanaan kapasitas yang lebih baik, sehingga sumber daya dapat dialokasikan dengan efisien dan kebutuhan pertumbuhan dapat diprediksi.
- **Pelaporan dan analisis:** System monitoring menghasilkan laporan dan analisis berdasarkan data yang dikumpulkan. Ini membantu system administrator dan pemangku kepentingan lainnya dalam memahami



Gambar 77: Server dengan Cloudflare dan Network-Based Firewall

Kalau seperti itu, kita harus menentukan IP address sumbernya, dalam hal ini adalah IP address dari Cloudflare. Masukkan semua [IP address Cloudflare](#), baik itu IPv4 maupun IPv6. Dengan konfigurasi rules seperti itu, semua lalu lintas jaringan untuk HTTP dan HTTPS harus melewati Cloudflare. Sementara untuk lalu lintas protokol ICMP dan SSH (custom port TCP 51234) boleh diakses dari IP address manapun.

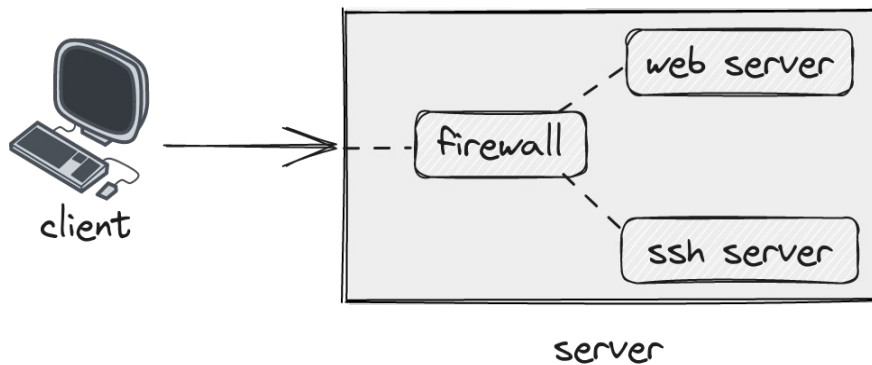
### Inbound Rules

Type	Protocol	Port Range	Sources
ICMP	ICMP		All IPv4 All IPv6
HTTP	TCP	80	103.21.244.0/22 173.245.48.0/20 2400:cb00::/32 2606:4700::/32
HTTPS	TCP	443	103.21.244.0/22 173.245.48.0/20 2400:cb00::/32 2606:4700::/32
Custom	TCP	51234	All IPv4 All IPv6

Gambar 78: Cloud Firewall Inbound Rules dengan Cloudflare

## iptables

iptables adalah perangkat lunak firewall pada sistem operasi Linux yang berfungsi untuk mengonfigurasi IP packet filter rules. Rules atau aturan-aturan tersebut dimasukkan ke dalam tabel yang berisi rantai-rantai aturan (chain of rules) yang menentukan bagaimana memperlakukan paket-paket lalu lintas jaringan.



Gambar 79: Server dengan Host-Based Firewall

iptables termasuk ke dalam kategori host-based firewall, dikonfigurasi langsung di host. Jika VPS yang digunakan tidak memiliki Cloud Firewall, kita menggunakan firewall host-based.

Secara default, iptables memiliki 4 table, yaitu Filter Table, NAT Table, Mangle Table, dan Raw Table. Kita menggunakan Filter Table untuk menyimpan rules yang menyaring paket-paket lalu lintas jaringan.

Filter Table memiliki 3 chain default, yaitu:

1. Chain INPUT: digunakan untuk menyaring paket-paket yang akan masuk ke server, apakah paket akan diterima (ACCEPT), ditolak (REJECT), atau ditolak tanpa pemberitahuan (DROP).
2. Chain OUTPUT: digunakan untuk menyaring paket-paket yang akan keluar dari server.
3. Chain FORWARD: digunakan untuk menyaring paket-paket yang akan diteruskan ke tujuan lain. Chain ini digunakan ketika server ingin difungsikan sebagai router atau gateway.

## Konfigurasi iptables

Pada sistem operasi Linux, secara default iptables telah tersedia. Tapi, kita perlu menginstalasi iptables-persistent agar rules yang sudah dimasukkan tetap tersimpan meskipun server sudah di-restart.

```
$ sudo apt install iptables-persistent -y
```

Menampilkan rules.

```
$ sudo iptables -L
```

Hasil perintah di atas jika rules belum ada.

```
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Pada hasil perintah di atas, policy untuk semua chain adalah ACCEPT. Artinya, secara default semua paket akan diterima.

### iptables Rules tanpa Cloudflare

Konfigurasi iptables yang pertama, jika kita tidak menggunakan atau tidak mengaktifkan Cloudflare proxy. Client akan mengakses server secara langsung ke server tanpa melewati proxy.

**PERINGATAN!** Kesalahan dalam konfigurasi firewall dapat mengakibatkan koneksi SSH ke server terputus. Jika terjadi, solusinya dengan mengakses server melalui web console yang tersedia di dasbor VPS. Kemudian, login dengan menggunakan username dan password (seperti login di server fisik) dan mengonfigurasi firewall agar server dapat diakses kembali melalui SSH.

```
$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
$ sudo iptables -A INPUT -s 127.0.0.0/8 -j ACCEPT
$ sudo iptables -A INPUT -p icmp -j ACCEPT
$ sudo iptables -A INPUT -p tcp --dport 51234 -j ACCEPT
$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
$ sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
$ sudo iptables -P INPUT DROP
$ sudo iptables -P FORWARD DROP
$ sudo netfilter-persistent save
$ sudo netfilter-persistent reload
$ sudo iptables -L
```

Penjelasan baris perintah konfigurasi iptables:

Tabel 8: Penjelasan Konfigurasi iptables

Rules	Deskripsi
-A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT	Menerima paket yang status koneksinya sudah terhubung dan yang terkait
-A INPUT -s 127.0.0.0/8 -j ACCEPT	Menerima paket yang berasal dari IP