BORN2BEROOT

Che cos'è una macchina virtuale?

Una macchina virtuale è un computer simulato all'interno di un computer fisico. E' un ambiente isolato che emula un sistema informatico completo, con un proprio sistema operativo, RAM, CPU etc..

Come funziona una macchina virtuale?

Una macchina virtuale funziona attraverso un software chiamato **HYPERVISOR o VIRTUAL MACHINE MONITOR**.
Questo software gestisce i diversi comportamenti chiave:
- la CPU(il cervello del computer – Central Prossessing Unit – Esegue i calcoli e le istruzioni del programma e coordina il funzionamento di tutte le latre parti del computer).
- La RAM(La memoria temporanea del computer -Random Access Memory – veloce da accedere tiene conto dei lavori svolti in un determinato momento, quando spegni il pc si cancella.
-Hard Disk(il magazzino permanente del computer, qui vengono conservati tutti i file, i programmi installati, il sistema operativo. Può essere SSH(solid state drive) o HDD (hard disk drive).
-Scheda di Rete(Componente che consente al pc di connettersi a internet).

Lo scopo delle macchine virtuali

Una macchina virtuale permette di eseguire piu sistemi operativi contemporaneamente sulla stessa macchina, mantenendo ogni ambiente completamente separato e sicuro. Utile per ottimizzare l'uso del hardware, testare software pericolosi.
Inoltre c'è la possibilità di creare SNAPSHOT dell'intero sistema.

SCELTA DEL SISTEMA OPERATIVO – DEBIAN

Debian → più semplice per i principianti, è stabile, ben documentato. E' supportato da una grande comunità.

DIFFERENZE TRA DEBIAN E ROCKY

Filosofia → **Debian** ha una distribuzione indipendente e comunitaria, focalizzata sulla stabilità e la libertà di software mentre **Rocky** deriva da Red Hat Entreprise Linux (RHEL) nata per sostituire CentOS, è pensata per gli ambienti entreprise.

Gestore di Pacchetti → **Debian** usa il gestore di pacchetti gpkg per il basso livello, si occupa dell'installazione dei file **.deb**, cioè un archivio che contiene il software compilato e pronto all'installazione. **Gpkg** viene udato da **APT** come backend.
APT è il **advanced packet tool** di Debian, esso gestisce le dipendenze automaticamente e si occupa del download e dell'installazione dei pacchetti (apt-get install; apt-get update; apt-get upgrade etc..).
Rocky Linux usa il gestore di pacchetti RPM con **dnf** (successore di **yum**). Offre migliori prestazini e gestione delle dipendenze. RMP gestisce file con estensione **.rpm,** che sono file binari eseguibili(cioè file pronti all'installazione).

Sistema di Rilascio → **Debian** si basa su tre rami di sviluppo principali che garantiscono diversi livelli di stabilità e aggiornamento del software. Il ramo **stable** che rappresenta la versione ufficiale e stabile di DEBIAN, rilasciata ogni 2-3 anni (la + affidabile e sicura) **[ultima versione del 2023 Debian 12 "Bookworm"]**.
Rocky → segue il ciclo di rilascio basato su RHEL. Ogni versione viene rilasciata poco dopo una major release di RHEL. Ogni versione è supportata per 10 anni.

Differenza tra APT, Aptitude e AppArmor

APT → gestore di pacchetti base

Rappresenta la soluzione più accessibile per la gestione di pacchetti, pensata per l'utente quotidiano che necessità di operazioni base come installare, aggiornare o rimuovere software. Ha un interfaccia *Command Line Interface* moderna, cioè l'interfaccia a riga di comando che permette all'utente di interagire con il sistema operativo.

Aptitude → gestore di pacchetti avanzato

Si posiziona come strumento più spfisticato, ideale per amministratori di sistema e utenti avanzati. Ha una CLI e una *text user interface* che crea una sorta di pseudo grafica utilizzando i caratteri ASCII. Utilizzo principale: risoluzione intelligente delle dipendenze[1].

AppArmor → sistema di sicurezza MAC(Mandatory Access Control)

Opera su un piano completamente diverso. E' un sistema di sicurezza integrato nel kernel che fornisce un livello aggiuntivo di protezione attraverso il controllo degli accessi delle applicazioni. {Per Rocky il sistema di controllo è *SELinux*, attivo di default, fornisce sicurezza in più}.

UFW- Uncomplicated Firewall

E' un'interfaccia per la configurazione di iptables(Un programma che serve a configurare il firewall di un sistema Linux, controllando il traffico di rete che entra ed esce dal computer. Iptables può creare un insieme di regole che definiscono esattamente come si deve comportare il firewall. Poiché iptables è molto potente, le configurazioni possono diventare complesse, ma consente un controllo molto fine sulla gestione del traffico di rete. Alcuni strumenti, come **UFW**, forniscono un'interfaccia più semplice per configurarlo.

COMANDO	COSA FA
sudo ufw status	Mostra se UFW è attivo e visualizza le regole di firewall attualmente applicate
sudo ufw status	Controlla lo status di ufw
sudo allow *numero_porta*	Permette il traffico di dati sulla porta inserita
sudo ufw enable	Attiva UFW sul sistema. Dopo averlo abilitato, il firewall inizierà a filtrare il traffico di rete in base alle regole configurate
sudo ufw disable	Disattiva UFW
Sufo ufw status numbered	Mostra le regole con un numero accanto a ciascuna
Sudo systemctl status ufw	Per verificare se il servizio UFW è attivo.

1 Le dipendenze sono requisiti software necessari affinché un programma possa funzionare correttamente. Sono librerie, pacchetti o altri programmi richiesti da un software per funzionare. Rappresentano infatti una relazione di dipendenza dove un programma necessita di altre componenti.

SECURE SHELL

E’ un protocollo di rete criptato che consente di accedere in modo sicuro a un sistema remoto. E’ ampiamente utilizzato per gestire server, dispositivi di rete e altre risorse in modo sicuro attraverso una connessione non protetta(come ad esempio internet). Garantisce confidenzialità, autenticazione e integrità. Come funziona? Quando un client si connette a un server via SSH, avviene un processo di autenticazione e cifratura che garantisce la comunicazione sicura.
Best practice: limitare l’accesso SSH a determinati utenti → modificare il file di configurazione **/etc/ssh/sshd_config** per consentire l’accesso solo a determinati utenti o gruppi; cambiare la porta predefinita (22) con una non standard (come la 4242 per Born2beroot).
Per usarlo bisogna installare OpenSSH
sudo apt install openssh-server

COMANDI SSH	
which ssh	Determina il percorso del comando ssh installato nel sistema output se comando installato: /usr/bin/ssh
ssh [*utente*]@[*indirizzo_IP_del_server]* **vhacman@localhost**	Connette a un server remoto usando ssh
sudo service ssh status	Esegue il comando con privilegi da amministratore, indica il servizio SSH e ne mostra lo stato attuale, cioè se è attivo, inattivo o se ci sono degli errori.

SUDO

SuperUser Do

E’ un programma di sistema che permette agli utenti di eseguire programmi con privilegi di sicurezza dell’utente root.
Ogni comando eseguito tramite sudo viene registrato nel sistema, creando una traccia di audit che può essere visualizzata per il monitoraggio della sicurezza. L'autenticazione in SUDO funziona richiedendo la password dell'utente per verificare la sua identità.

SUDO POLICIES

```
Defaults passwd_tries=3
Defaults badpass_message="Clave incorrecta"
Defaults logfile="/var/log/sudo_config"
Defaults log_input, log_output
Defaults iolog_dir="/var/log/sudo"
Defaults requiretty
Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
```

Il sistema di sicurezza limita a 3 i tentativi di inserimento della password prima di bloccare l'accesso, contrastando attacchi brute-force. In caso di errore, mostra il messaggio "passwd errata". I comandi sudo vengono registrati in **/var/log/sudo.config**, mentre i log dettagliati sono salvati in **/var/log/sudo**. La policy **requiretty**[2] impone l'uso di un terminale reale per sudo, prevenendo esecuzioni non autorizzate. Infine, un **secure path** definisce le directory consentite per i comandi sudo, evitando l'esecuzione di software malevolo.

2TTY (TeleTYpewriter) è un terminale che fornisce un'interfaccia testuale per interagire con il sistema operativo. È essenzialmente una finestra di comando che permette agli utenti di: Inserire comandi ; Vedere i risultati; Interagire direttamente con il sistema → quindi TTY è il terminale mentre, per esempio bash, è il programma che gira dentro il terminale.

PASSWORD POLICY

Per modificare le password policy bisogna fare
nano /etc/login.defs
Modifica apportata = valore di aging della password (PASS_MAX_DAYS) modificato a 30 giorni; PASS_MIN_DAYS modificata a 2 → devono passare almeno due giorni tra il cambio di password.
VANTAGGI → la scadenza ogni 30 giorni riduce il rischio che password compromesse possano essere utilizzate per periodi lunghi. Il periodo minimo impedisce che gli utenti aggiornino il sistema tornando immediatamente alla vecchia password. La limitazione a 3 tentativi protegge da attacchi brute force. E' facile da configurare attraverso il file di configurazione standard. L'esecuzione solo da terminale reale (requiretty) aumenta la sicurezza. Svantaggi: cambio frequente di password può portare gli utenti a usare password più semplici.
Per rinforzare la qualità della password installiamo altri pacchetti, come la libreria PAM(Pluggable Authentication Module) che fornisce controlli sulla qualità della password
sudo apt install libpam-pwquality
Successivamente si edita il file standard con:
nano /etc/pam.d/common-password
e si aggiungono dei parametri alla configurazione principale della password:
password requisite pam_pwquality.so
retry=3 minlen=10 ucredit=-1 dcredit=-1 reject_username difok=7 enforce_for_root
ci sono 3 tentativi per inserire la password valida
richiede una lunghezza minima di 10 caratteri
richiede almeno una lettera maiuscola (ucredit)
richiede almeno una cifra (dcredit)
la password non può contenere il nome utente (reject_username)
la nuova password deve differire dalla precedente per almeno 7 caratteri (difok=7)
applica le regole anche all'utente root (enforce_for_root)
Questa configurazione implementa una politica di password robusta che:

1. Assicura una complessità minima
2. Previene il riutilizzo di password simili
3. Impedisce l'uso di password facilmente indovinabili
4. Non fa eccezioni nemmeno per l'amministratore di sistema

HOSTNAME E PARTIZIONI

hostname = nome assegnato al computer all'interno di una rete.

Comando → **hostname** →visualizzare hostname

Comando → **hostname *nuovo_hostname*** → modifica hostname temporaneamente

Comando → **touch /etc/hostname** *modifica file con nuovo_hostname salva ed esci.*

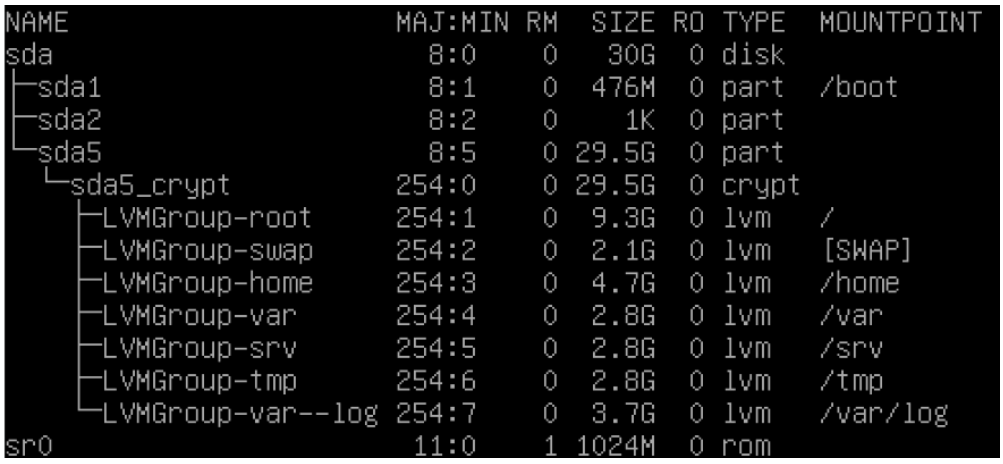
LVM – LOGICAL VOLUME MANAGEMENT

LVM (Logical Volume Manager) è un sistema che permette di gestire lo spazio su disco in modo più flessibile rispetto al partizionamento tradizionale. Invece di dividere fisicamente il disco in partizioni fisse, LVM crea un livello di astrazione che consente di ridimensionare le partizioni (chiamate volumi logici) senza dover riformattare o perdere dati.

Funziona organizzando i dischi fisici in "gruppi di volumi", dentro i quali si possono creare e modificare liberamente i volumi logici. Quindi abbiamo il Volume fisico(il disco su cui viene creato LVM), poi abbiamo il gruppo volume → un aggregazione di più volumi fisici che rappresentano uno spazio gestibile dinamicamente, infine abbiamo il volume logico, che è la partizione virtuale creata all'interno del gruppo volume. **Vantaggi?** *Si possono aumentare o ridurre le dimensioni di un volume senza dover formattare; si può creare copia istantanea dei dati per backup o test; permette di unire più dischi in un unico volume o di distribuirli per migliorare le prestazioni.*

Come vedere le partizioni della VM?

Lsblk



La partizione **root (/),** da 30.3GB, è la p***artizione principale del sistema operativo***. Contiene i file essenziali di sistema, programmi installati e librerie fondamentali per il funzionamento del sistema.

L'area **swap** di 2.3GB funziona come ***memoria virtuale aggiuntiva.*** Viene ***utilizzata quando la RAM fisica è esaurita***, permettendo al sistema di continuare a funzionare spostando dati meno utilizzati sul disco.

La directory **/home** di 5GB è lo s**pazio dedicato ai dati personali degli utenti**. Qui vengono salvati *documenti, configurazioni personali e file degli utenti del sistema.*

La directory **/var** di 3GB ***contiene dati variabili come database, cache e file che cambiano frequentemente durante l'operatività del sistema.***

La directory **/srv** di 3GB è dedicata ai ***dati dei servizi offerti dal sistema***. Per esempio, se il server ospita un sito web, i suoi file potrebbero essere memorizzati qui.

La directory **/tmp** di 3GB è utilizzata per i ***file temporanei creati sia dal sistema che dalle applicazioni. Questi file vengono tipicamente cancellati al riavvio del sistema.***

Infine, **/var/log** di 4GB è dedicata ai ***file di log del sistema. Qui vengono registrate tutte le attività del sistema, errori, accessi e altre informazioni cruciali per il monitoraggio e la diagnosi dei problemi.***

SCRIPT SYSTEM INFO

Uno **script** è come una **lista di istruzioni** che il computer segue passo dopo passo. È un file di testo che contiene comandi scritti in un linguaggio di programmazione (come Bash, Python, ecc.). Invece di scrivere ogni comando a mano ogni volta, lo scrivi una volta in uno **script** e lo fai eseguire al computer quando serve! Gli script possono essere utilizzati per raccogliere e visualizzare informazioni sul sistema, come:

- Il nome del sistema operativo
- La versione del kernel
- L'uso della CPU e della RAM
- Lo spazio disponibile sul disco
- Gli utenti attualmente connessi

Si crea il file monitoring.sh con un editor di testo
ex: **nano monitoring.sh**
#script
Si rende eseguibile lo script:
chmod +x monitoring.sh
si esegue lo script
./monitoring.sh
Output: stampa informazioni utili sul sistema.

Come funziona uno script?
Per prima cosa si crea il file di testo che conterrà tutti i comandi
#!/bin/bash
# ARCH
arch=$(uname -a) → *rileva architettura di sistema*
# CPU PHYSICAL
cpuf=$(grep "physical id" /proc/cpuinfo | wc -l) → *conta il numero di CPU fisiche*
# CPU VIRTUAL → *conta il numero di CPU virtuali*
cpuv=$(grep "processor" /proc/cpuinfo | wc -l)
# RAM → ottiene l'uso della ram
ram_total=$(free --mega | awk '$1 == "Mem:" {print $2}')
ram_use=$(free --mega | awk '$1 == "Mem:" {print $3}')
ram_percent=$(free --mega | awk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}') → *per ottenere la ram totale usata e la percentuale di utilizzo.*
# DISK → *calcola uso del disco*
disk_total=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_t += $2} END {printf ("%.1fGb\n"), disk_t/1024}')
disk_use=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_u += $3} END {print disk_u}')
disk_percent=$(df -m | grep "/dev/" | grep -v "/boot" | awk '{disk_u += $3} {disk_t+= $2} END {printf("%d"), disk_u/disk_t*100}') → *usa df -m per calcolare la dimensione totale, lo spazio usato e la percentuale di utilizzo del disco*
# CPU LOAD → *misura il carico della CPU. Usa **vmstat** per ottenere la percentuale di idle della CPU e calcola l'uso attivo*
cpul=$(vmstat 1 2 | tail -1 | awk '{printf $15}')
cpu_op=$(expr 100 - $cpul)
cpu_fin=$(printf "%.1f" $cpu_op)
# LAST BOOT
lb=$(who -b | awk '$1 == "system" {print $3 " " $4}') → *ottiene ultimo riavvio del sistema*
# LVM USE → *verifica LVM in uso*
lvmu=$(if [$(lsblk | grep "lvm" | wc -l) -gt 0]; then echo yes; else echo no; fi)
# TCP CONNEXIONS → *conta le connessioni TCP attive. Usa ss -ta per contare le connessioni TCP stabilite*
tcpc=$(ss -ta | grep ESTAB | wc -l)
# USER LOG → *conta gli utenti attualmente loggati.*
ulog=$(users | wc -w)
# NETWORK → *ottiene l'indirizzo IP e MAC*
ip=$(hostname -I)
mac=$(ip link | grep "link/ether" | awk '{print $2}')
# SUDO → *conta i comandi eseguiti con sudo con journalctl*
cmnd=$(journalctl _COMM=sudo | grep COMMAND | wc -l)
wall " Architecture: $arch
CPU physical: $cpuf
vCPU: $cpuv
Memory Usage: $ram_use/${ram_total}MB ($ram_percent%)
Disk Usage: $disk_use/${disk_total} ($disk_percent%)
CPU load: $cpu_fin%
Last boot: $lb
LVM use: $lvmu
Connections TCP: $tcpc ESTABLISHED
User log: $ulog
Network: IP $ip ($mac)
Sudo: $cmnd cmd" → ***mostra tutto a schermo***
📌 *Il comando `wall` invia il messaggio a tutti gli utenti connessi al sistema.*

architettura di sistema

L'architettura si riferisce alla progettazione e all'organizzazione dei componenti di un computer, come il processore, la memoria e il sistema operativo.
1. Cos'è l'architettura di un computer?
L'architettura di un computer definisce il modo in cui le sue parti hardware e software interagiscono.
Per ottenere informazioni sull'architettura di sistema si usa il comando uname -a

CRONTAB

Crontab è un servizio di pianificazione dei processi nei sistemi Unix/Linux, esso permette di eseguire automaticamente script, comandi o programmi in momenti specifici.
Per configurare crontab
sudo crontab -u root -e

> Usato per modificare il file di **cron(**che gestisce i compiti pianificati) per l'utente root.
> -e → serve a modificare il file *crontab* esistente.
> -u root → indica che il file *crontab* da modificare è quello dell'utente root, senza questa specifica il comando agirebbe sul crontab dell'utente che inserisce il comando.

nel file bisogna aggiungere
***/10 * * * * sh /path_to_file.sh**
*/10 → Ogni 10 minuti
* → ogni ora
* → ogni giorno del mese
* → ogni mese
* → ogni giorno della settimana

SIGNATURE .TXT

-spegnere la virtual machine.
-trovare il file .vdi della macchina virtuale sul computer fisico → contiene l'immagine del disco virtuale.
- si usa il comando **shasum [nome_della_macchina].vdi** per generare la signature. Il risultato di questo comando va salvatoe nel file signature.txt e caricato nella repo del progetto da consegnare.

shasum Born2beroot.vdi

Il comando **shasum** è uno strumento che genera una checksum SHA-1 del file, permettendo di verificarne l'integrità
Non riavviare la macchina dopo aver generato la signature, altrimenti cambierà.

SERVIZI BONUS

LIGHTTPT

E' un server web, ovvero un programma che permette di gestire siti e pagine web.
Installare → **sudo apt install lighttpd**
abilita l'avvio automatico → **sudo systemctl enable lighttpd**

WORDPRESS

E' un sistema di gestione dei contenuti progettato per la creazione di qualsiasi tipo di sito web.
Per installare l'ultima versione → **sudo apt install wget zip**
wget → download files from the web
zip → command line utility for compressing and decompressing files in zip format.
Bisogna navigare nella cartella /var/ww/ e dobbiamo scaricare l'ultima versione di WP. Dopo aver scaricato il file zip lo decomprimiamo con **sudo unzip [nome_file.zip].**
Rinominiamo la cartella di WP in html per fare in modo che il server web la riconosca come la directory principale del sito. Per garantire che il server web possa accedere correttamente ai file, dobbiamo impostare i permessi sulla cartella html → **sudo chmod -R 755 html**
permesso di lettura, scrittura ed esecuzione al proprietario, e per il gruppo e altri utenti solo lettura ed esecuzione.

MARIA DB

MariaDB è un sistema di gestione di database relazionale, il che significa che permette di organizzare e gestire dati in modo strutturato attraverso tabelle correlate tra loro. La sua natura open-source significa che il codice è pubblicamente accessibile, gratuito e può essere modificato dalla comunità di sviluppatori. Per installare → **sudo apt install mariadb-server**
Una volta installato, la configurazione predefinita lascia il sistema non sicuro, quindi è necessario eseguire uno script per restringere l'accesso al server e rimuovere eventuali account inutilizzati. Questo script è fornito con il pacchetto `mariadb-server` e si esegue con il comando: **sudo mysql_secure_installation**
Lo script ci chiederà alcune configurazioni per migliorare la sicurezza del server MariaDB:
switch to unix_socket autentication? -N → non vogliamo abilitare l'autenticazione tramite socket UNIX, perché abbiamo già un account root protetto
change the root password? -N → non vogliamo cambiare la password dell'utente root.
Remove anonymous users?-Y → rimuove utenti anonimi. Per impostazione predefinita mariadb crea un utente anonimo che consente a chiunque di accedere al database senza bisogno di un account
disallow root login remotely?-Y → rimuove il database di test e gli accessi ad esso. Mariadb crea un database di test per scopi di sviluppo ma non è sicuro lasciarlo attivo in ambiente di sviluppo
remove test database and access to it?-Y → per ricaricare le tabelle di privilegi. Questo applicherà immediatamente tutte le modifiche alla configurazione di sicurezza.
Una volta completata l'installazione di MariaDB, il passo successivo è creare il database e l'utente necessari per far funzionare WordPress. Per iniziare, dobbiamo accedere a MariaDB utilizzando il comando: **mariadb**
Una volta all'interno dell'ambiente creiamo il database per WP.

CREATE DATABASE wp_database;

#per verificare che il database sia stato creato correttamente

SHOW DATABASES;

#*Successivamente, dobbiamo creare un utente che avrà accesso al database appena creato. Per farlo, usiamo il comando*:

CREATE USER 'vhacman'@'localhost' IDENTIFIED BY '12345';

In questo caso, l'utente che stiamo creando si chiama `vhacman` e la password è `12345`. Ora, dobbiamo associare l'utente al database, in modo che abbia i permessi necessari per interagire con il database. Per farlo, eseguiamo il comando:

GRANT ALL PRIVILEGES ON wp_database.* TO 'vhacman'@'localhost';

Per far sì che le modifiche ai permessi abbiano effetto, eseguiamo il comando:

FLUSH PRIVILEGES;

exit

PHP

Linguaggio di programmazione principalmente utilizzato per sviluppare applicazioni web dinamiche e siti web interattivi. E' eseguito sul lato server, il che significa che il codice PHP viene eseguito dal server web e non dal browser dell'utente. Per poter eseguire applicazioni web scritte in PHP che necessitano di connettersi a un database MySQL (come nel caso di WordPress), dobbiamo installare i pacchetti necessari. Questi pacchetti permettono al server web di eseguire il codice PHP e di interagire con il database MySQL.

#per installare pacchetti necessari:

sudo apt install php-cgi php-mysql

#php-chi → è il modulo che consente di eseguire script PHP in modalità CGI (Common Gateway Interface), che permette di gestire le richieste PHP inviate dal server web.

#php-mysql → è l'estensione che permette a PHP di interagire con un database MySQL o MariaDB, facilitando la connessione e l'esecuzione di query sul database.

PER CONFIGURARE WORDPRESS SUL SERVER

Accedi alla directory di WP

cd /var/www/html

Copia e rinomina il file di configurazione di WP

cp wp-config-sample.php wp-config.php

Apri il file **wp-config.php** per modificarlo e configurarlo con le informazioni del database:

nano wp-config.php

DB_NAME: wp_database

DB_USER: vhacman

DP_PASSWORD: 12345

DB_HOST: localhost

Abilita il modulo fastcgi in Lighttpd
Per migliorare le prestazioni e la velocità delle applicazioni web PHP sul server, abilitiamo il modulo `fastcgi` e `fastcgi-php` di Lighttpd:

sudo lighty-enable-mod fastcgi

sudo lighty-enable-mod fastcgi-php

Ricarica la configurazione di Lighttpd

sudo service lighttpd force-reload

Avvia l'installazione di WordPress
A questo punto, puoi aprire il tuo browser e andare su `localhost` per avviare l'installazione di WordPress.

LITESPEED →

LiteSpeed è conosciuto per le sue capacità di gestione del traffico ad alto volume e per il miglioramento delle prestazioni in termini di velocità, stabilità e sicurezza. LiteSpeed è il quarto server web più popolare a livello mondiale e viene utilizzato da circa il 10% dei siti web. LiteSpeed è spesso scelto per siti web che richiedono prestazioni elevate, come quelli di e-commerce o con traffico elevato.

OpenLiteSpeed è una scelta eccellente per chi cerca un server web veloce, sicuro e facile da configurare. È particolarmente utile per siti web che utilizzano PHP e richiedono un'alta capacità di gestione del traffico. La versione open-source mantiene molte delle caratteristiche avanzate di LiteSpeed, ed è una delle migliori opzioni gratuite per server web ad alte prestazioni.

Come?

Sudo apt update

sudo apt upgrade

wget -O - https://repo.litespeed.sh | sudo bash → official debian repo

sudo apt upsate *#again*

sudo apt install openlitespeed

sudo /usr/local/lsws/admin/misc/admpass.sh → *#change password from 123456 in something more secure*

sudo ufw allow 8088/tcp

sudo ufw allow 7080/tcp

sudo ufw reload

→ We configure the firewall to allow connections through ports 8088 and 7080. We then add the rules in the port forwarding

Once we have completed the previous step we can connect. We will put in the search engine of our browser `localhost:7080` we provide our login credentials and we will have access to everything.

User: idroot

*Password: ******

LITESPEED →

LiteSpeed è conosciuto per le sue capacità di gestione del traffico ad alto volume e per il miglioramento delle prestazioni in termini di velocità, stabilità e sicurezza. LiteSpeed è il quarto server web più popolare a livello mondiale e viene utilizzato da circa il 10% dei siti web. LiteSpeed è spesso scelto per siti web che richiedono prestazioni elevate, come quelli di e-commerce o con traffico elevato.

OpenLiteSpeed è una scelta eccellente per chi cerca un server web veloce, sicuro e facile da configurare. È particolarmente utile per siti web che utilizzano PHP e richiedono un'alta capacità di gestione del traffico. La versione open-source mantiene molte delle caratteristiche avanzate di LiteSpeed, ed è una delle migliori opzioni gratuite per server web ad alte prestazioni.

Come?

Sudo apt update

sudo apt upgrade

wget -O - https://repo.litespeed.sh | sudo bash → official debian repo

sudo apt upsate *#again*

sudo apt install openlitespeed

sudo /usr/local/lsws/admin/misc/admpass.sh → *#change password from 123456 in something more secure*

sudo ufw allow 8088/tcp

sudo ufw allow 7080/tcp

sudo ufw reload

→ *We configure the firewall to allow connections through ports 8088 and 7080. We then add the rules in the port forwarding*

Once we have completed the previous step we can connect. We will put in the search engine of our browser `localhost:7080` we provide our login credentials and we will have access to everything.

User: idroot

*Password: *****

VANTAGGI → Immagina di avere un sito e-commerce con molti visitatori. Utilizzando **OpenLiteSpeed**, il sito carica più velocemente grazie al **caching PHP**[3], migliorando l'esperienza utente. Inoltre, proteggi il sito da attacchi DDoS [4]con le sue funzionalità di sicurezza. La gestione tramite interfaccia web semplifica le configurazioni senza complicazioni (pannello di controllo che consente di gestire e configurare il server (come OpenLiteSpeed) facilmente tramite un browser).

PROTOCOLLO TCP

Il **protocollo TCP (Transmission Control Protocol)** è uno dei principali protocolli di comunicazione utilizzati in rete, facente parte del **modello TCP/IP**, che è alla base di Internet. TCP si occupa della **trasmissione affidabile** dei dati tra due dispositivi in rete.

3 Tecnica di memorizzare in una memoria temporanea (cache) i dati generati dinamicamente da PHP, come il contenuto di una pagina web → **riduce il tempo di elaborazione**

4 Litespeed può difendere il sito web da attacchi in cui molti computer tentato di sovraccaricare il server con richieste eccessive.