

IIS "Denina - Rivoira - Pellico"
Sezione ITIS Giovanni Rivoira - A.S. 2023 - 2024
Informatica - 5L



Istituto Istruzione Superiore
Carlo Denina

Documentazione progetto Raspberry

Gruppo 2

Audisio Nicolò, Bracco Mattia, Galfrè Beniamino Maria

Contents

1	Introduzione	1
1.1	Cos'è Raspberry?	1
1.2	Per cosa si utilizza il Raspberry?	2
1.3	Scheda tecnica del Raspberry PI 3 Model B+	5
2	Installazione del Sistema Operativo	6
2.1	Procedure pre-installazione	6
2.2	Installazione del Sistema Operativo sulla macchina	8
3	Configurazione del Raspberry	8
3.1	Configurazione di base	8
3.1.1	Creazione dell'utente	9
3.1.2	Negare l'aggiornamento	10
3.2	Primo aggiornamento del Sistema Operativo	11
3.3	Modifiche da effettuare	13
3.3.1	Modifica impostazioni di sistema	13
3.3.2	Modifica impostazioni dello schermo	13
3.3.3	Modifica impostazioni delle interfacce	13
3.3.4	Conferma delle modifiche	13
3.4	Configurazione di rete	14
3.4.1	Impostazioni della nostra scheda di rete	15
3.5	Connessione SSH da un computer remoto	16
4	Installazione TigerVNC Server	20
4.1	Cosa succede se si riavvia il Raspberry?	23
4.2	Come ricollegarsi da remoto a TigerVNC?	23
5	Filezilla	24
5.1	Cos'è FileZilla e a cosa serve?	24
5.2	Installazione di FileZilla da terminale	24
5.3	Installazione di FileZilla dal <i>Sito Ufficiale</i>	25
5.4	Utilizzo di FileZilla	26
6	Protocollo FTP	29
6.1	Funzioni	29
6.2	Modalità di connessione client/server	29
6.2.1	Modalità attiva	29
6.2.2	Modalità passiva	30
6.2.3	Principali differenze	31
6.3	Modalità di accesso	31

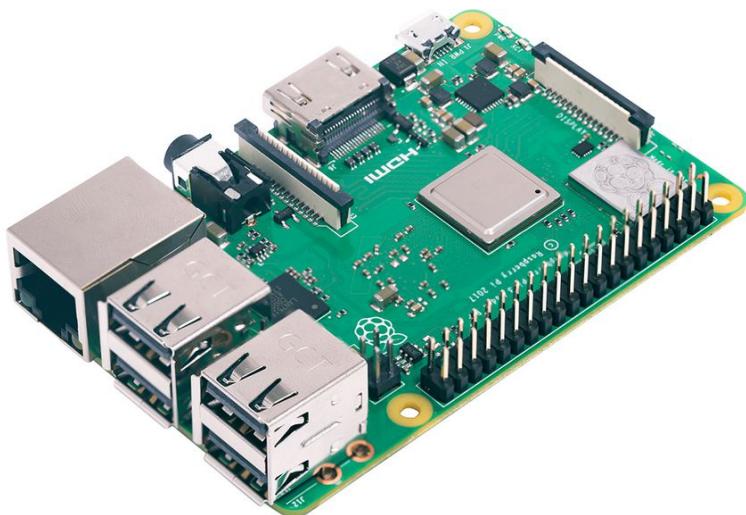
6.4	Comandi SFTP	31
6.4.1	Connessione	32
6.4.2	Gestione directory	32
6.4.3	Trasferimento file	32
6.4.4	Permessi e ownership	33
6.4.5	Informazioni sul file	33
6.4.6	Connessione e chiusura	33
6.5	Differenze tra FTP, TFTP, SFTP, FTPS	33
6.5.1	FTP (File Transfer Protocol)	33
6.5.2	Caratteristiche principali di FTP	33
6.5.3	Funzionamento di FTP	34
6.5.4	Casi d'uso di FTP	35
6.5.5	Vantaggi di FTP	35
6.5.6	Svantaggi di FTP	36
6.6	SFTP (SSH File Transfer Protocol)	36
6.6.1	Caratteristiche principali di SFTP	36
6.6.2	Funzionamento di SFTP	37
6.6.3	Casi d'uso di SFTP	37
6.6.4	Vantaggi di SFTP	37
6.6.5	Svantaggi di SFTP	38
6.7	FTPS (FTP Secure)	38
6.7.1	Caratteristiche principali di FTPS	38
6.7.2	Funzionamento di FTPS	39
6.7.3	Casi d'uso di FTPS	39
6.7.4	Differenze tra FTPS e SFTP	40
6.8	TFTP (Trivial File Transfer Protocol)	40
6.8.1	Caratteristiche principali di TFTP	40
6.8.2	Funzionamento di TFTP	41
6.8.3	Operazione TFTP	41
6.8.4	Casi d'uso di TFTP	41
6.8.5	Differenze tra TFTP e FTP	42

1 Introduzione

1.1 Cos'è Raspberry?

Il Raspberry Pi è una piattaforma rivoluzionaria che dal 2012 ha aperto nuove prospettive per maker di ogni livello e per chiunque voglia imparare a programmare senza grandi investimenti. È un minicomputer composto da una singola scheda, sviluppato nel Regno Unito dalla Fondazione Raspberry Pi, con l'obiettivo specifico di insegnare e promuovere le basi dell'insegnamento della Computer Science nelle scuole e nei paesi in via di sviluppo.

Ha le tipiche caratteristiche ricercate da professionisti ed amatori: costa pochissimo, è completo, piccolissimo e facile da configurare. Può essere considerato un “computer in miniatura”, un intero ecosistema hardware raccolto in un'unica scheda. Raspberry Pi può essere utilizzato per realizzare una vasta gamma di progetti, tra cui videogiochi, sistemi audio, computer e stampanti 3D, media center e smartphone. Inoltre, è possibile utilizzare la scheda per la smart home, la gestione di robot e persino per costruire una macchina fotografica artigianale.



1.2 Per cosa si utilizza il Raspberry?

- **Apprendimento della programmazione:** particolarmente adatto per imparare a conoscere le strutture hardware di un computer e iniziare a cimentarsi nella programmazione.

Può essere utilizzato per la realizzazione di programmi in qualsiasi linguaggio e utilizzando *PiBakery* per la programmazione di un progetto mediante la OOP in maniera grafica, come *Scratch*.



Figure 1: Immagine del software PiBakery

- **Progetti fai-da-te e domotici:** numerosi utenti ingegnosi e fai da te hanno realizzato svariati progetti di diversa natura. Un esempio può essere la realizzazione di una piccola stazione meteo dove attraverso dei sensori di temperatura e umidità i dati rilevati vengono stampati su un display LCD, oppure controllare attraverso la domotica l'illuminazione interna o esterna di un edificio.

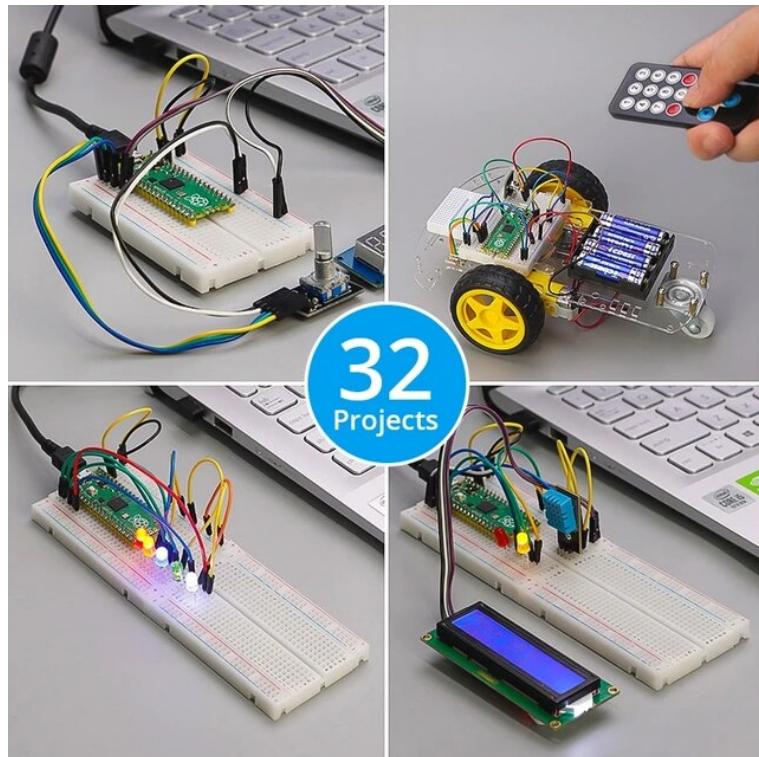


Immagine rappresentativa di alcuni progetti fai-da-te

- **Computer desktop di base:** può essere utilizzato come computer desktop di base per le attività quotidiane, come la navigazione sul web, l'elaborazione di testi e gestione di piccoli contenuti multimediali.

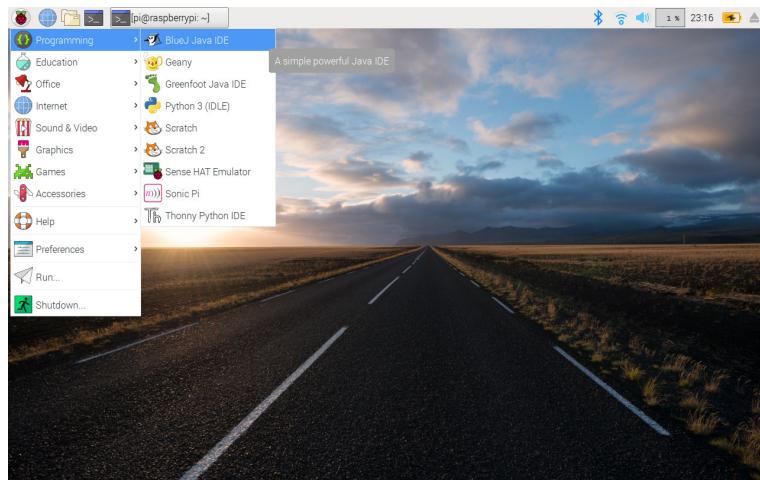


Immagine rappresentativa dell'interfaccia desktop di Raspberry Pi OS

1.3 Scheda tecnica del Raspberry PI 3 Model B+

Caratteristica	Dettaglio
Processore	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz
Memoria	1GB
Connettività	LAN wireless dual-band 2.4GHz e 5GHz IEEE 802.11b/g/n/ac Bluetooth 4.2, BLE Gigabit Ethernet 4 × interfaccia USB 2.0
Video e suono	1 x HDMI Porta di visualizzazione MIPI DSI Porta della fotocamera MIPI CSI Uscita stereo a 4 poli e porta video composita
Multimedia	Decodifica H.264, MPEG-4 (1080p30) Codifica H.264 (1080p30) Grafica OpenGL ES 1.1, 2.0
Supporto scheda SD	formato Micro SD per il caricamento del sistema operativo e lo storage dei dati
Alimentazione	5V/2.5A DC tramite connettore micro USB 5V DC tramite header GPIO Power over Ethernet (PoE) abilitato (richiede un PoE HAT separato)
Temperatura operativa	0-50°C

2 Installazione del Sistema Operativo

2.1 Procedure pre-installazione

In questo capitolo vengono mostrati i passaggi fondamentali per installare un sistema operativo sulla piattaforma Raspberry Pi.

Partendo dall'installazione del software Raspberry Pi Imager, che permetterà successivamente di preparare la scheda SD con il sistema operativo desiderato. Infine, verrà mostrato il processo di scaricamento e installazione del sistema operativo sulla scheda SD, preparando così la scheda per l'avvio e l'utilizzo del Raspberry Pi.

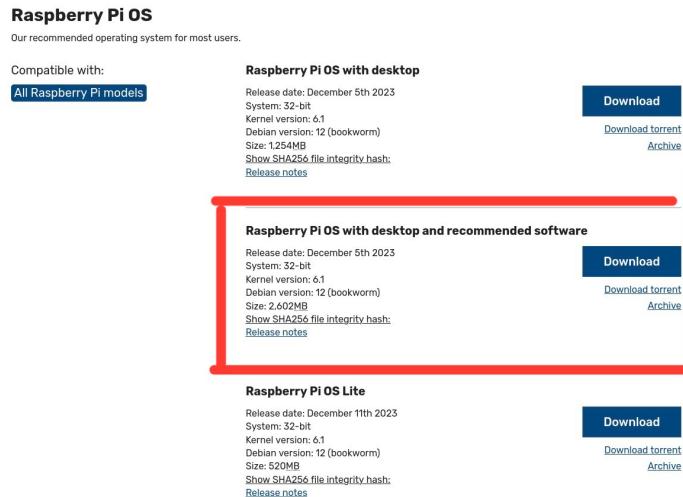
Passaggi per l'installazione del sistema operativo:

1. **Installazione del Raspberry Pi Imager:** Utilizzare il comando seguente per installare il Raspberry Pi Imager sul vostro sistema:

```
sudo apt install rpi-imager
```

In alternativa, è possibile scaricare l'immagine del sistema operativo direttamente dal *Sito Ufficiale*.

2. **Selezione del Sistema Operativo:** Selezionare il sistema operativo preferito. Per questo esempio, abbiamo scelto il “*Raspberry Pi OS with desktop and recommended software* (x32bit)”.



3. **Scaricamento del Sistema Operativo:** Dopo aver selezionato il sistema operativo, scaricare il file del sistema operativo sul proprio PC.
4. **Caricamento del Sistema Operativo sulla Scheda SD:** Utilizzare il Raspberry Pi Imager, precedentemente installato, per caricare il file del sistema operativo all'interno della scheda SD.



5. **Installazione del Sistema Operativo:** Dopo aver copiato il file .img del sistema operativo sulla microSD, procedere con l'installazione del sistema operativo sulla macchina.

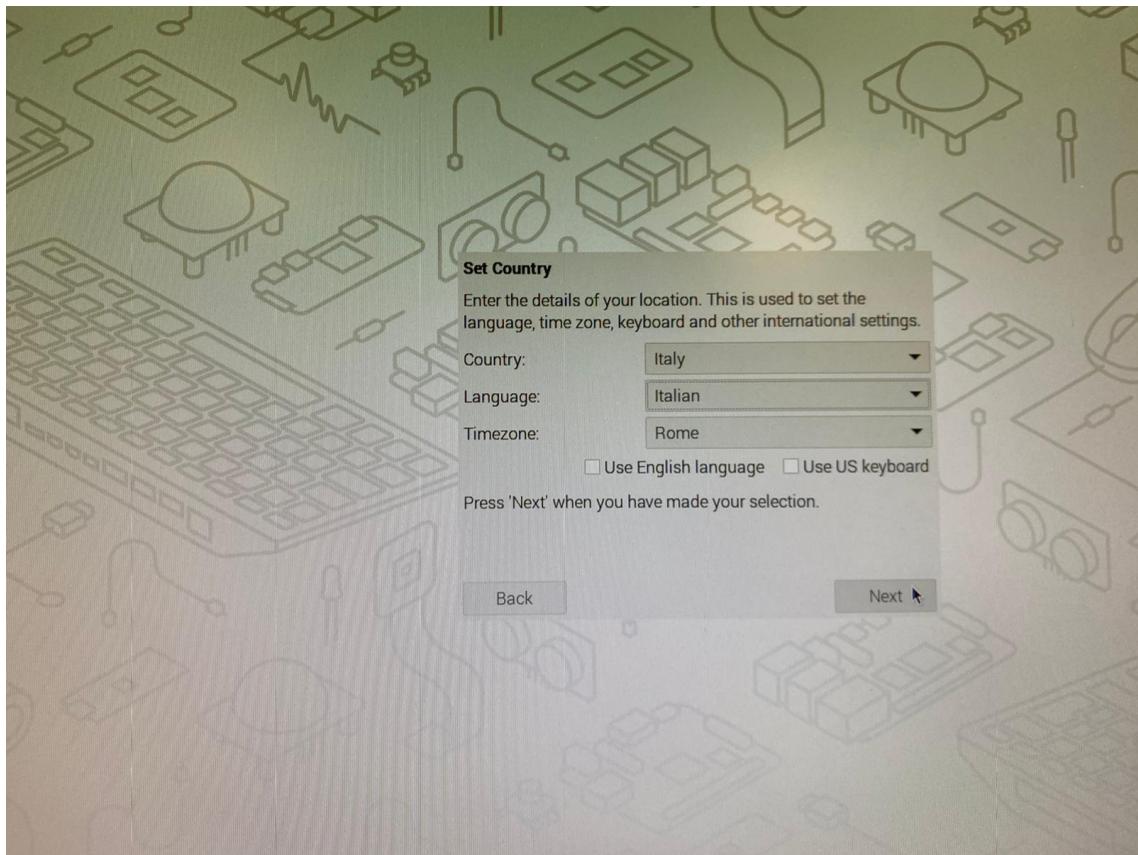
2.2 Installazione del Sistema Operativo sulla macchina

Dopo l'installazione sulla scheda SD, procedere ad inserirla nell'apposito slot SD presente sul retro della scheda e avviare il dispositivo accendendo l'alimentazione.

3 Configurazione del Raspberry

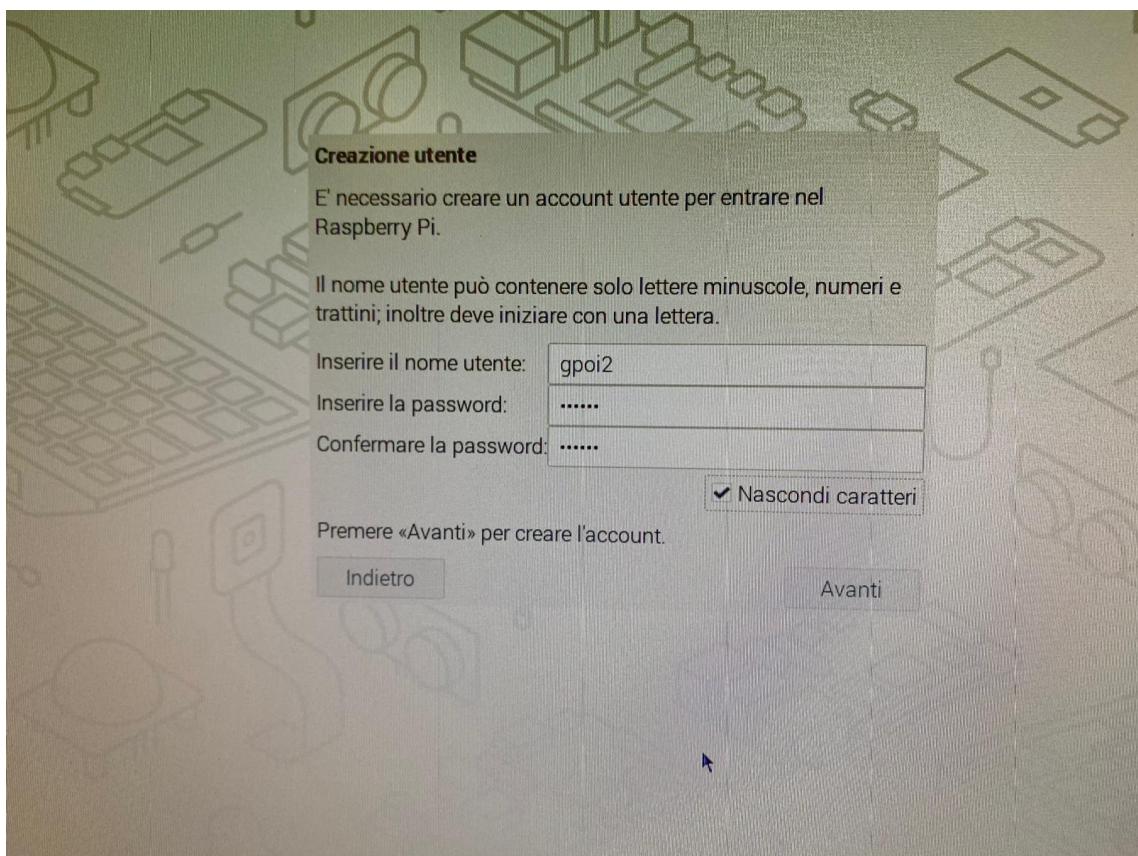
3.1 Configurazione di base

Parametro	Valore
Country	Italy
Language	Italian
Time Zone	Rome

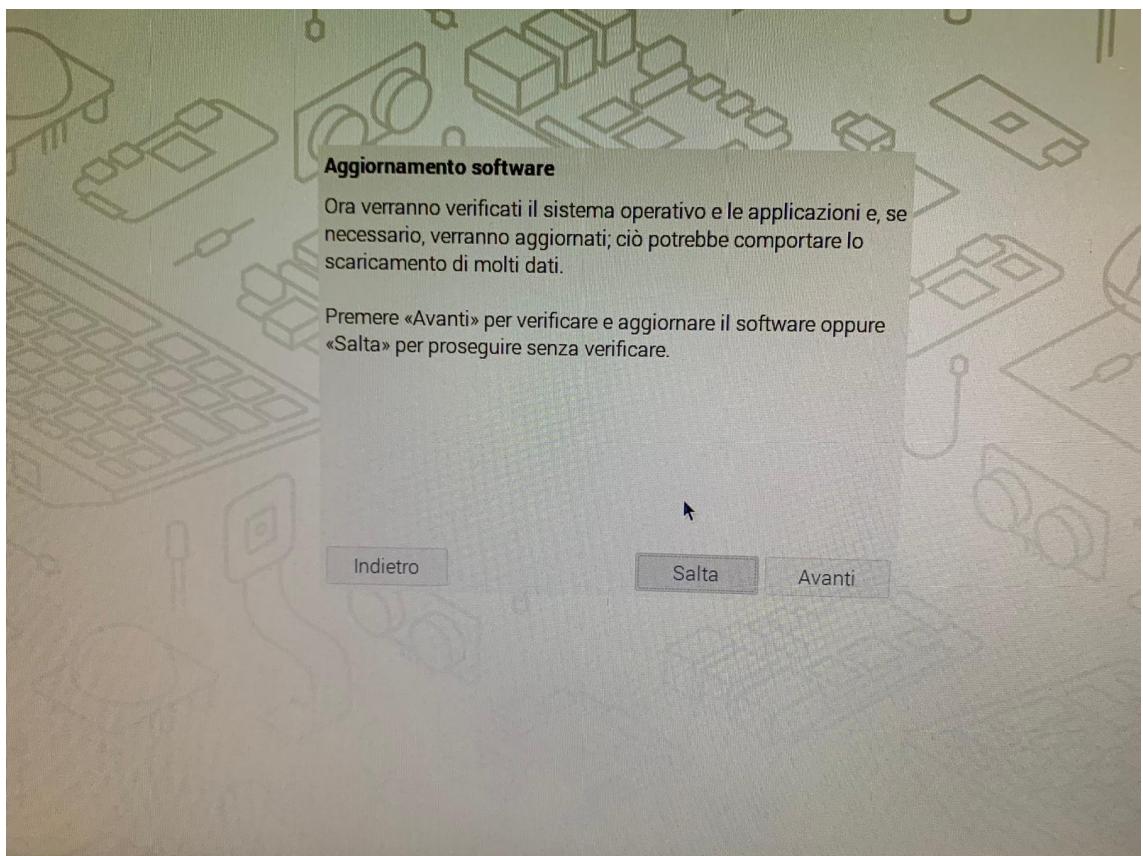


3.1.1 Creazione dell'utente

Parametro	Valore
Nome Utente	gpoi2
Password	pilota



3.1.2 Negare l'aggiornamento



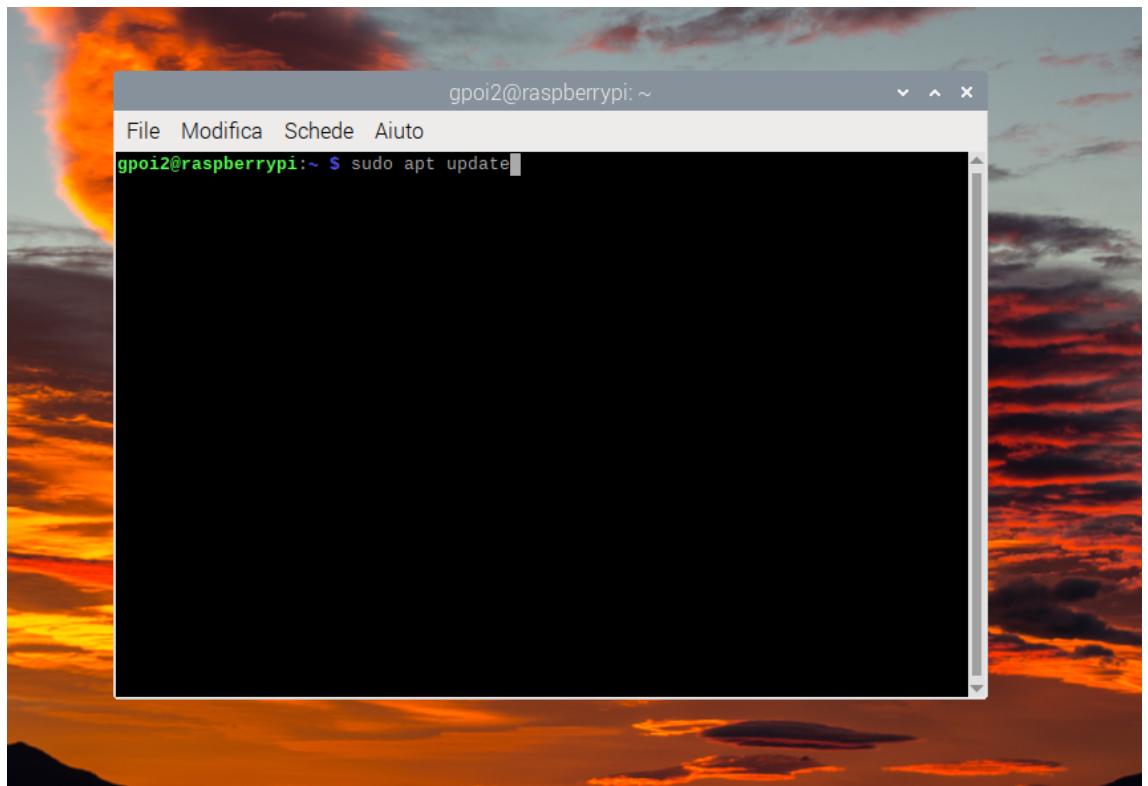
Si consiglia di saltare l'aggiornamento software in fase di installazione perché si potranno installare in un secondo momento a configurazione terminata.

3.2 Primo aggiornamento del Sistema Operativo

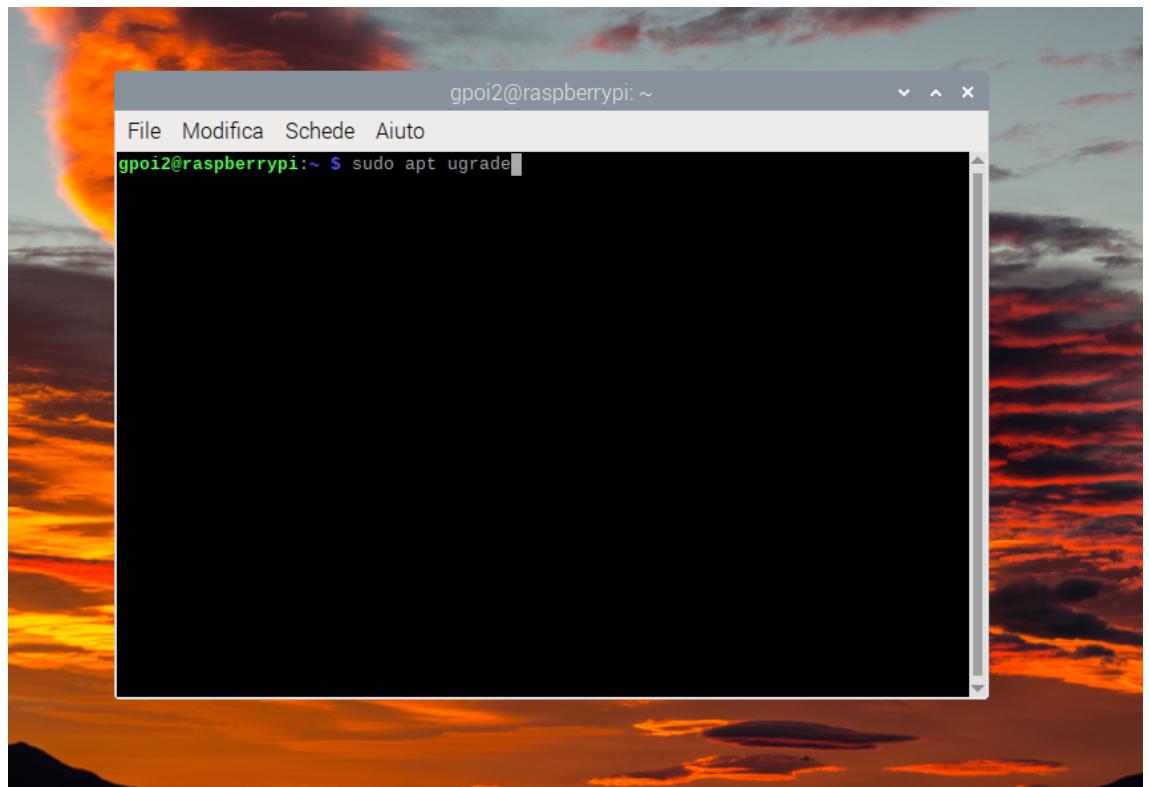
Al primo avvio bisogna procedere con l'aggiornamento del *Sistema Operativo* accedendo al terminale ed eseguendo i seguenti comandi.

Elenco dei comandi:

- `sudo apt update`
- `sudo apt upgrade`
- `sudo reboot`



Controllo aggiornamento dei pacchetti software



Installazione aggiornamenti dei pacchetti software

Successivamente si procede con il riavvio del sistema tramite interfaccia grafica oppure tramite il comando:

```
sudo reboot
```

3.3 Modifiche da effettuare

Successivamente, dopo aver riavviato il dispositivo, bisogna accedere nelle impostazioni di sistema per effettuare le seguenti modifiche:

3.3.1 Modifica impostazioni di sistema

All'interno delle impostazioni di sistema bisogna andare a **disabilitare accesso automatico**.

3.3.2 Modifica impostazioni dello schermo

All'interno delle impostazioni dello schermo **modificare la risoluzione dello schermo** per adattarlo meglio al proprio schermo.

3.3.3 Modifica impostazioni delle interfacce

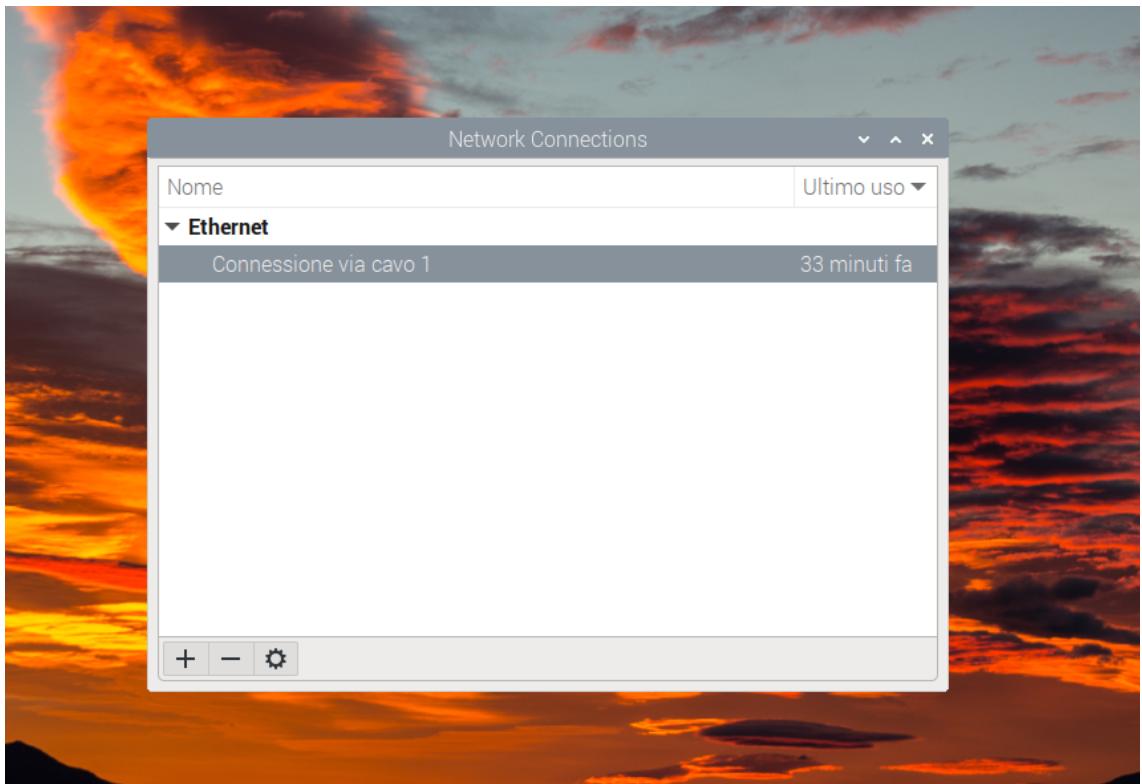
All'interno delle impostazioni delle interfacce andare ad **abilitare SSH e VNC**.

3.3.4 Conferma delle modifiche

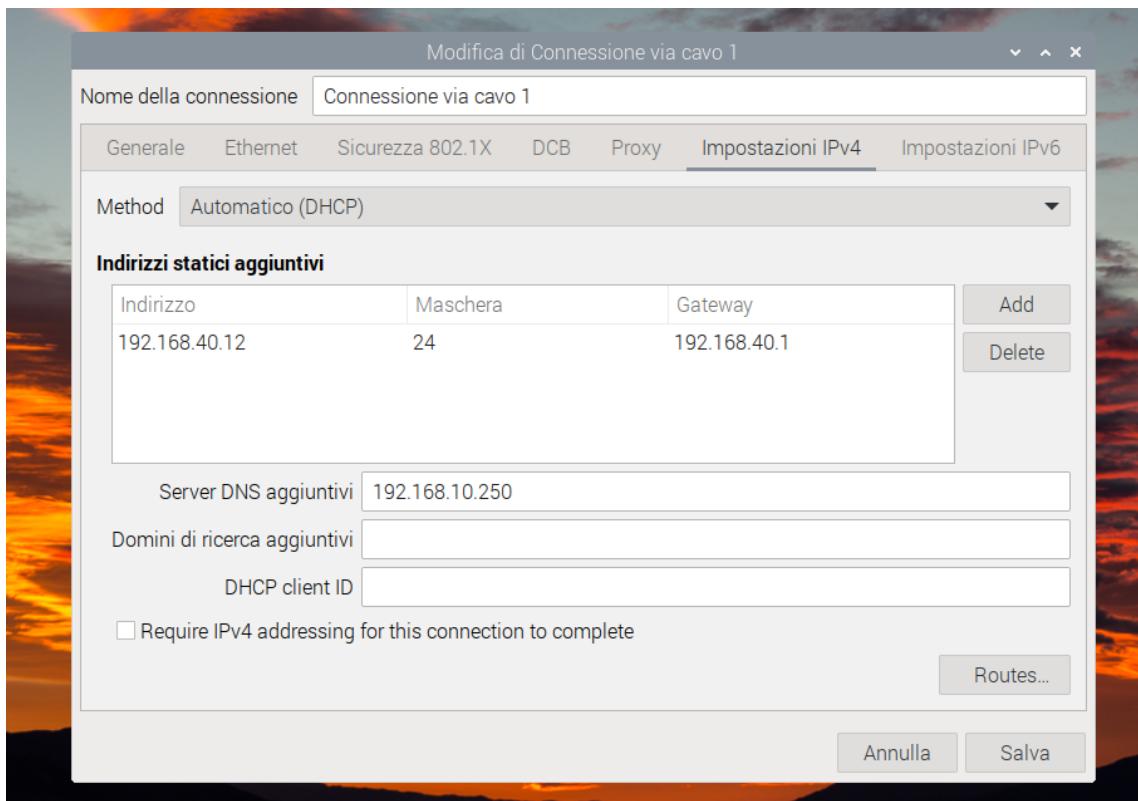
Per confermare le modifiche appena effettuate bisogna **riavviare il dispositivo**.

3.4 Configurazione di rete

Per andare a configurare le impostazioni di rete del Raspberry bisogna accedere nelle *Network Connections*



Successivamente bisogna recarsi nelle impostazioni, andare nella voce *Impostazioni IPv4* e aggiungere un nuovo indirizzo IP premendo sul pulsante *Add*.



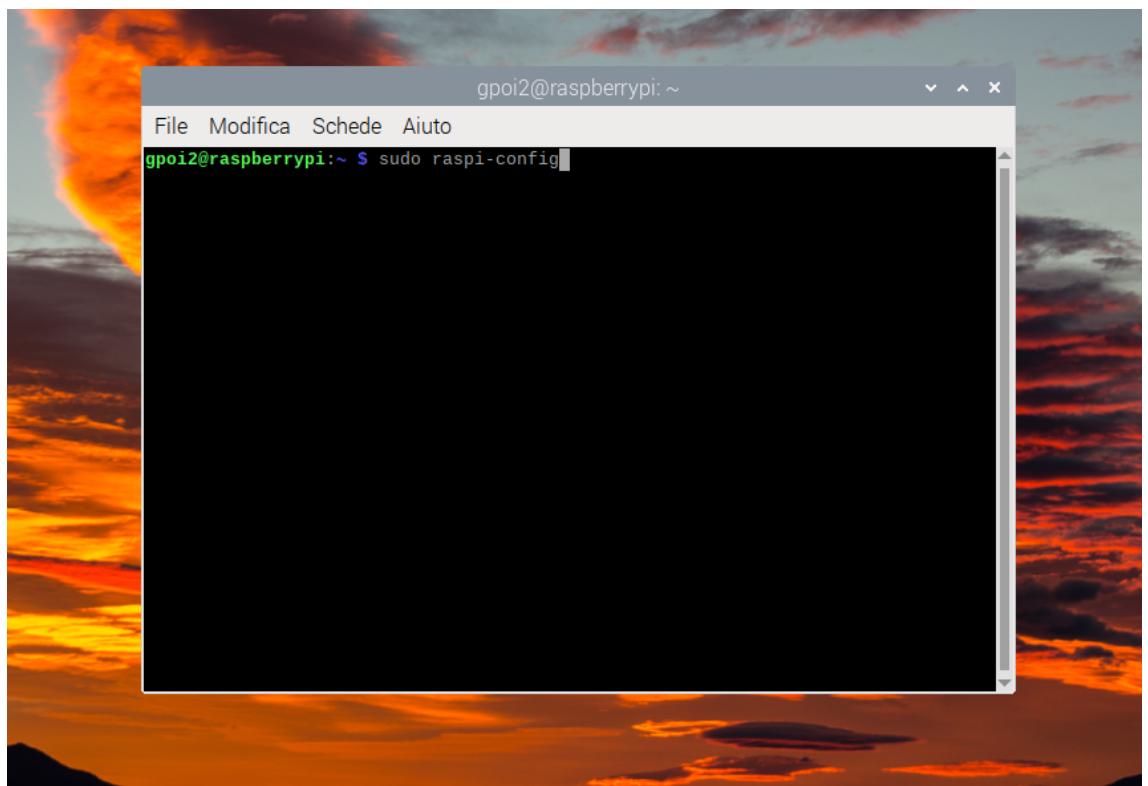
3.4.1 Impostazioni della nostra scheda di rete

Parametro	Valore
Indirizzo IP	192.168.40.12
Subnetmask	255.255.255.0
IP Default Gateway	192.168.40.1
IP DNS Server	192.168.10.250

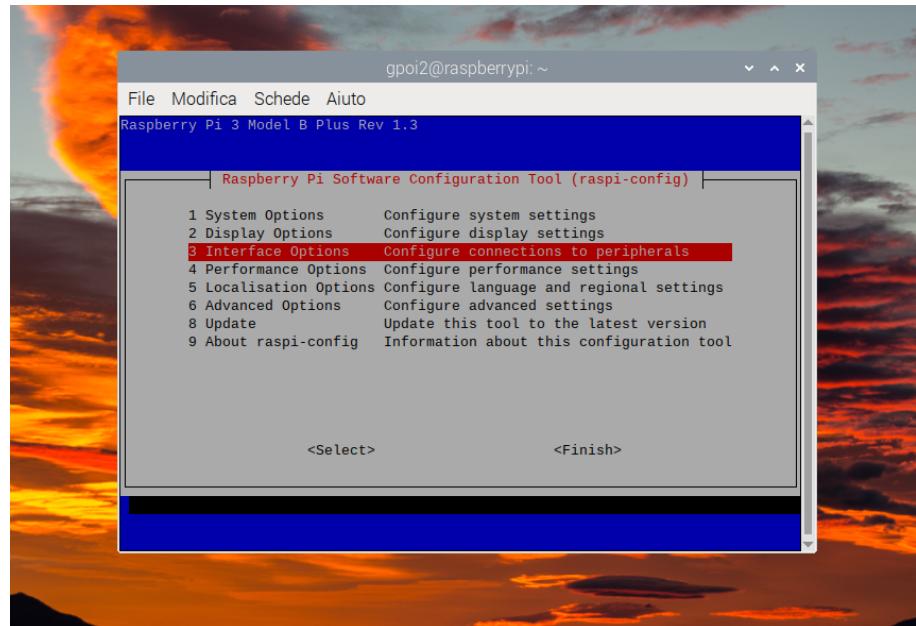
3.5 Connessione SSH da un computer remoto

Per verificare se la connessione SSH funziona, bisogna usare un computer remoto collegato sulla stessa rete del Raspberry.

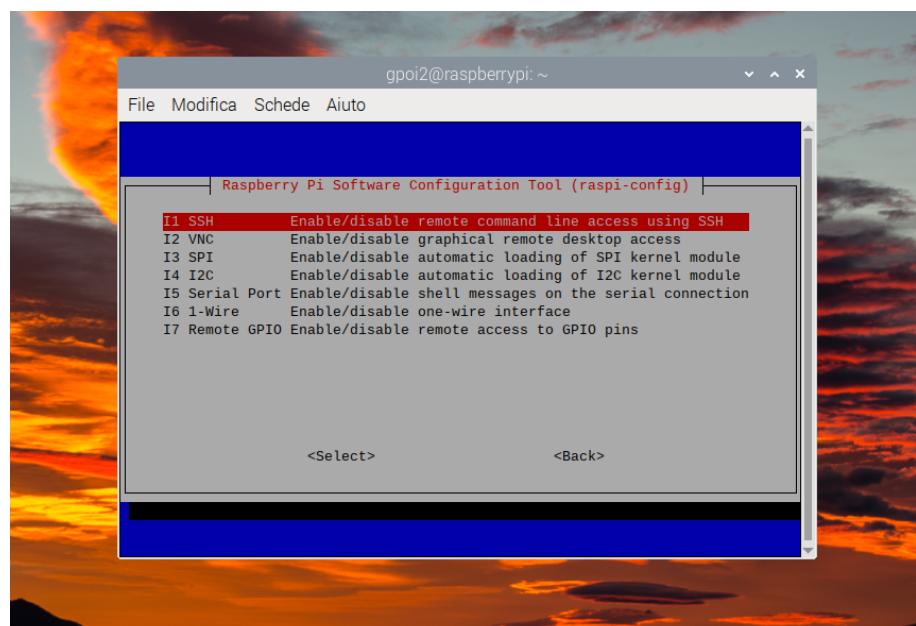
1. **Abilitare il server SSH dal Raspberry:** Accedere sul terminale e digitare il comando: `sudo raspi-config`



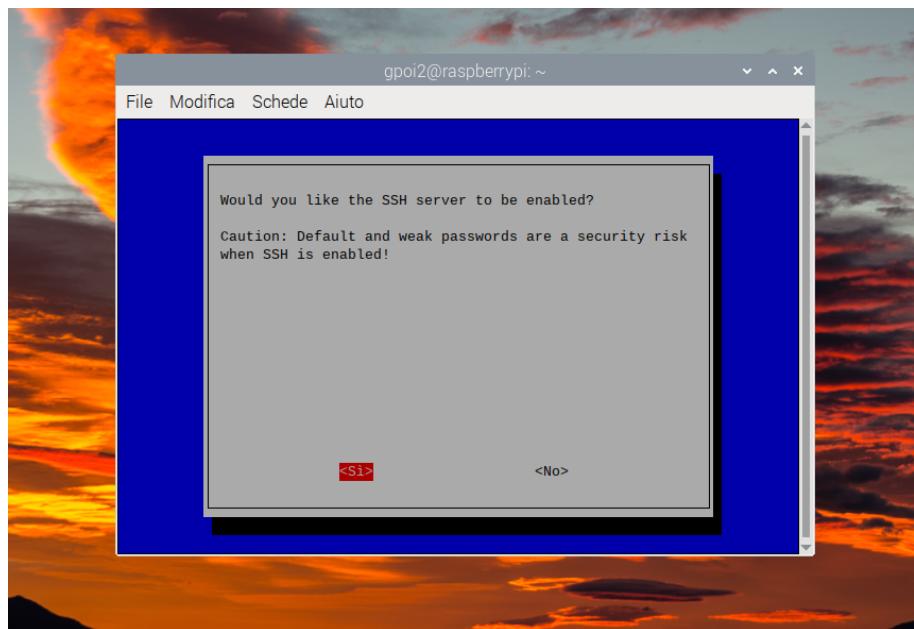
- Accedere a "Interface Options";



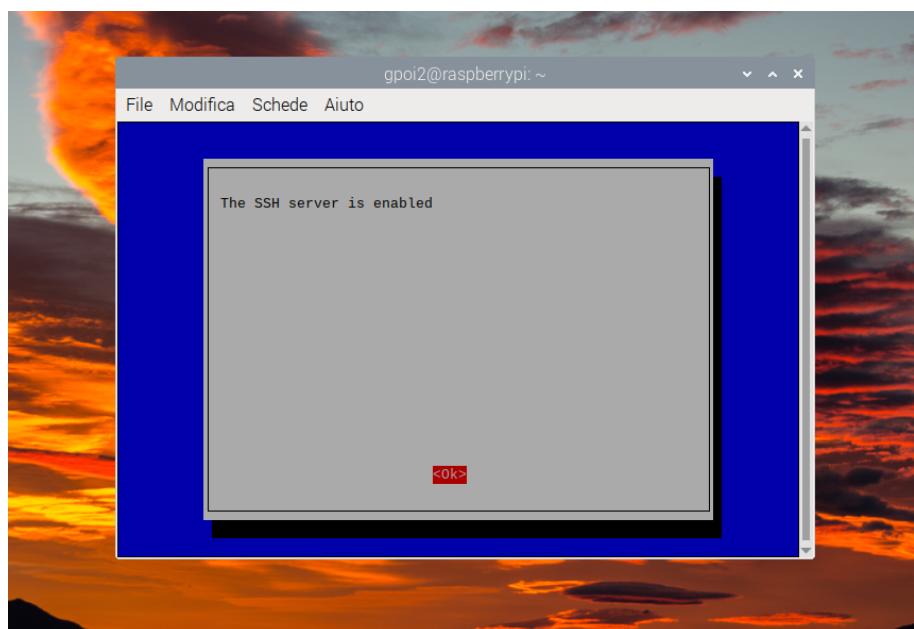
- Accedere a "SSH";



- Premere su ”Yes”: Approvazione per l’attivazione del server SSH;

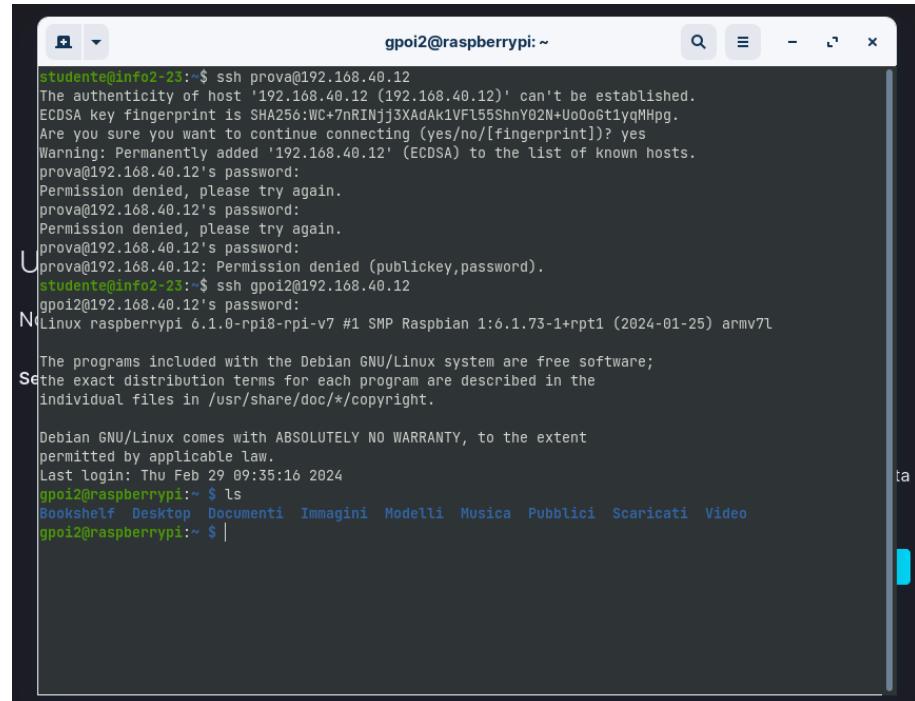


- Premere su ”OK”: Conferma dell’attivazione del server SSH;



2. **Collegamento al server SSH da un altro dispositivo:** Accedere da terminale e digitare il comando:

- **ssh nomeUtenteRaspberry@indirizzoIpRaspberry**
- Inserire la password dell'account Raspberry



```
studente@info2-23:~$ ssh prova@192.168.40.12
The authenticity of host '192.168.40.12 (192.168.40.12)' can't be established.
ECDSA key fingerprint is SHA256:WC+7nRINjj3XAdAk1VF155ShnY02N+UoOoGt1yqMHpg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.40.12' (ECDSA) to the list of known hosts.
prova@192.168.40.12's password:
Permission denied, please try again.
prova@192.168.40.12's password:
Permission denied, please try again.
prova@192.168.40.12's password:
prova@192.168.40.12: Permission denied (publickey,password).
studente@info2-23:~$ ssh gpoi2@192.168.40.12
gpoi2@192.168.40.12's password:
Linux raspberrypi 6.1.0-rpi18-rpi-v7 #1 SMP Raspbian 1:6.1.73-1+rpi1 (2024-01-25) armv7l

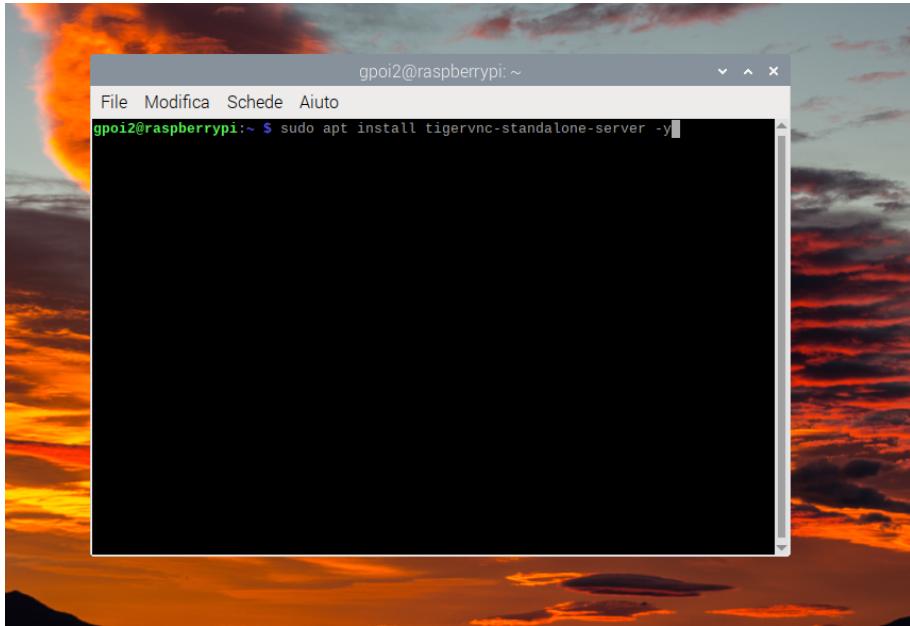
The programs included with the Debian GNU/Linux system are free software;
see the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Feb 29 09:35:16 2024
gpoi2@raspberrypi:~ $ ls
Bookshelf Desktop Documenti Immagini Modelli Musica Pubblici Scaricati Video
gpoi2@raspberrypi:~ $ |
```

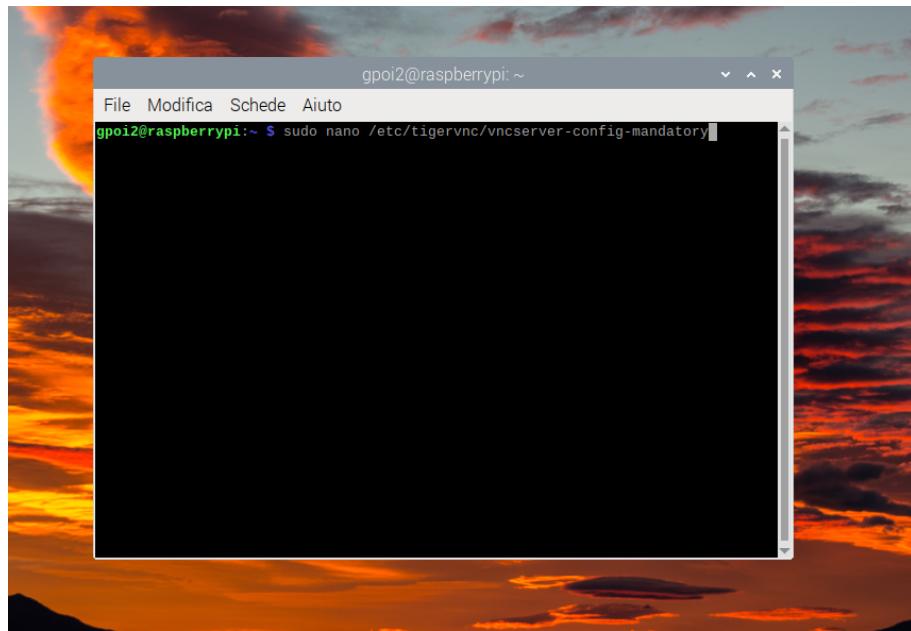
4 Installazioen TigerVNC Server

Per procedere con l'installazione di *TigerVNC Server*:

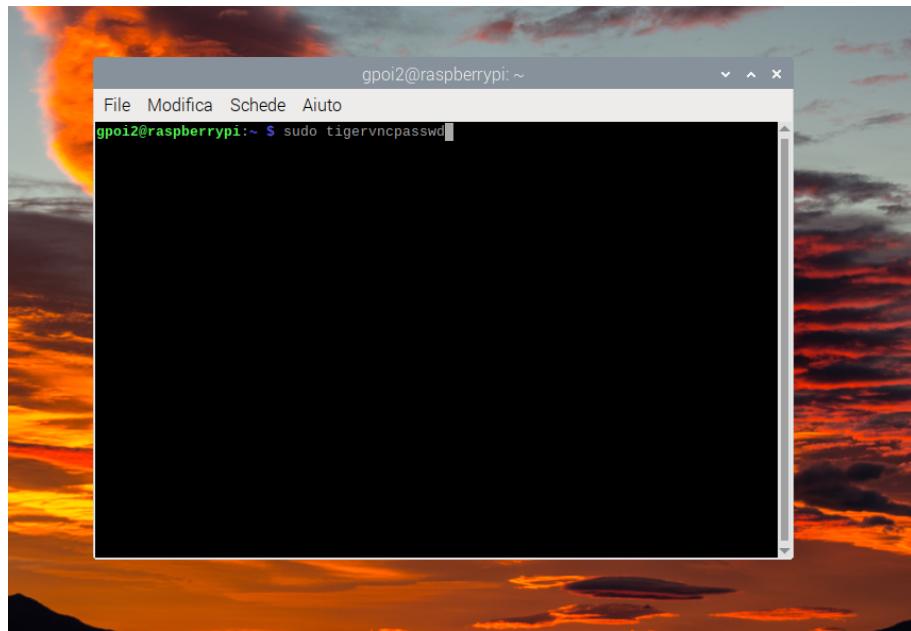
```
sudo apt install tigervnc-standalone-server
```



```
sudo nano /etc/tigervnc/vncserver-config-mandatory
```

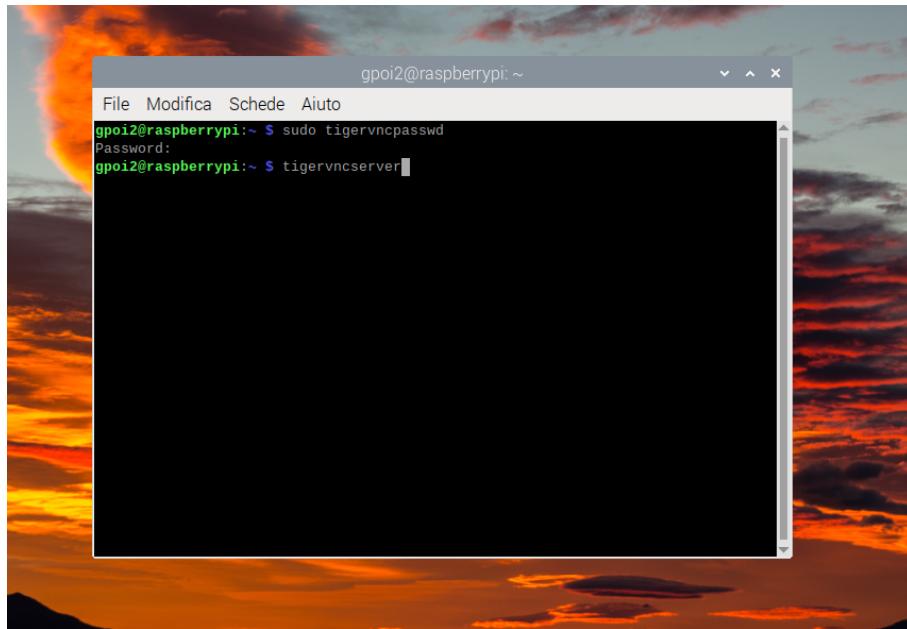


```
sudo tigervncpasswd
```



Per procedere bisogna inserire la password precedentemente configurata

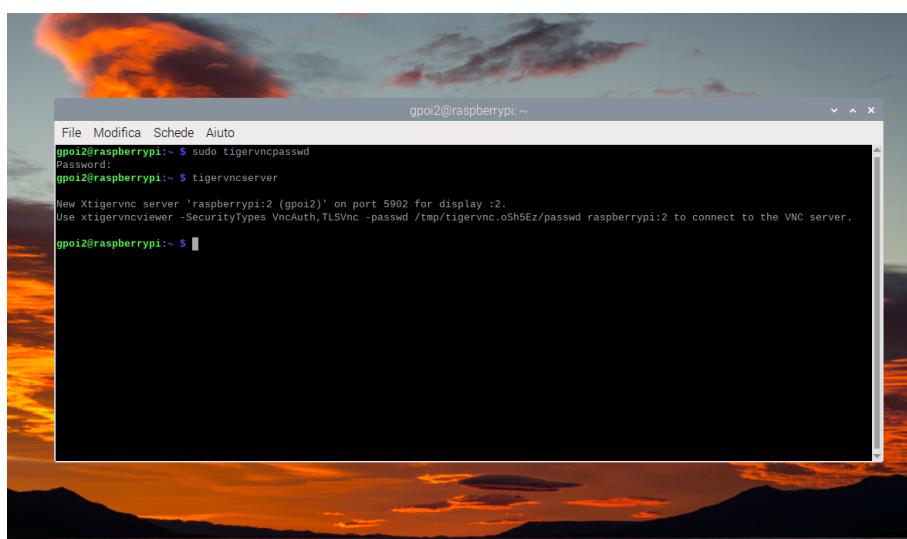
tigervncserver



Se si ha inserito correttamente la password e la configurazione è andata a buon fine, dopo l'esecuzione del comando l'utente si ritroverà all'interno del nostro dispositivo.

Lo si può notare dal terminale che riporta

```
gpoi2@raspberry: ~ $
```



4.1 Cosa succede se si riavvia il Raspberry?

Dopo l'installazione e la prima configurazione, se si procede a riavviare il dispositivo, non sarà possibile collegarsi da remoto attraverso il servizio TigerVNC.

4.2 Come ricollegarsi da remoto a TigerVNC?

Per ricollegarsi nuovamente da remoto è necessario riavviare il servizio di TigerVNC, per effettuare questa operazione ci si può collegare in SSH al Raspberry.

Bisogna effettuare i seguenti passaggi:

1. **Accendere il Raspberry Pi:** Assicurarsi che il Raspberry sia acceso e connesso alla rete.
2. **Avviare il server VNC sul Raspberry:** Accedendo tramite SSH da terminale avviare il servizio relativo a TigerVNC.
3. **Connettere il client VNC:** Sul computer, avviare il client di TigerVNC e inserire l'indirizzo IP del Raspberry seguito dal numero di porta mostrato a video nel terminale al momento dell'avvio del servizio sul Raspberry.

Se non si è a conoscenza dell'indirizzo IP del Raspberry lo si può ricavare utilizzando il seguente comando.

ifconfig

4. **Inserisci la password VNC:** Quando viene richiesto, inserire la password che impostata per il server VNC sul Raspberry.

5 Filezilla

5.1 Cos'è FileZilla e a cosa serve?

FileZilla è un programma gratuito e open source che facilita il trasferimento di file tramite il protocollo FTP (File Transfer Protocol). È disponibile per GNU/Linux, Microsoft Windows e macOS.

Ecco alcune informazioni chiave su FileZilla:

1. **FileZilla Client:** Questo è il programma principale che consente di trasferire file dal tuo computer locale a un server remoto tramite FTP. È uno dei programmi FTP più famosi grazie alla sua multipiattaforma e alla facilità d'uso. Puoi gestire più server FTP contemporaneamente e spostare i file trascinandoli da un punto all'altro.
2. **FileZilla Server:** Questo è un software aggiuntivo, anch'esso gratuito e open source, che ti permette di creare un server. Puoi accettare connessioni in entrata e comunicare con i client tramite protocollo FTP,SFTP o FTPS (SSL/TLS). In altre parole, puoi utilizzarlo come archivio per rendere disponibili alcuni file agli utenti che utilizzano un client.

In breve, FileZilla è uno strumento essenziale per gestire i trasferimenti di file tra il tuo computer e un server remoto. Se hai bisogno di trasferire file sul tuo spazio FTP, FileZilla è una scelta eccellente!

5.2 Installazione di FileZilla da terminale

1. Installazione di FileZilla:

```
sudo apt install filezilla
```

2. Avvio Firelilla:

```
filezilla
```

5.3 Installazione di FileZilla dal *Sito Ufficiale*

- Pagina di download:

The screenshot shows the official FileZilla website. On the left, there's a sidebar with links to Home, FileZilla (Features, Screenshots, Download, Documentation, FileZilla Pro), FileZilla Server (Download), Community (Forum, Wiki), and General (FAQ, Support). The main content area is titled "Overview". It welcomes visitors to the homepage of FileZilla®, the free FTP solution. It mentions the public license, offers for FileZilla Pro and FileZilla Server, support through forums, wiki, and bug reports, and documentation for compilation. Below this, a section titled "Quick download links" features two large buttons: "Download FileZilla Client" (All platforms) and "Download FileZilla Server" (All platforms).

- Download Client:

The screenshot shows the "Download FileZilla Client for Linux (64bit x86)" page. The sidebar on the left is identical to the homepage. The main content area is titled "Download FileZilla Client for Linux (64bit x86)". It states that the latest stable version is 3.66.5 and asks users to select the appropriate file for their platform. A green button labeled "Download FileZilla Client" with a red arrow is prominently displayed. Below it, there's an information icon and a note about the Debian 10.0 (Buster) 64bit edition being highly recommended. Another section titled "More download options" shows icons for other platforms: Linux 64-bit, Linux 32-bit, Mac OS X, and Windows.

5.4 Utilizzo di FileZilla

Di seguito è riportato un esempio di utilizzo di FileZilla per il trasferimento di una directory, e dei suoi relativi file contenuti, sul Raspberry utilizzando il protocollo *SFTP*.

1. Creazione della directory sul nostro dispositivo.

```
mkdir componentiGruppo
```



A screenshot of a terminal window titled "studente@info2-17: ~/Documenti". The window shows a black background with white text. At the top, there are icons for a plus sign, a dropdown arrow, a magnifying glass, a list icon, and close/minimize/maximize buttons. The terminal prompt is "studente@info2-17: ~/Documenti\$". Below the prompt, the command "mkdir componentiGruppo" is typed in green, followed by a new line character. The cursor is positioned at the end of the command line.

2. Entrare all'interno della directory.

```
cd componentiGruppo
```

3. Creazione dei file all'interno della directory.

```
echo "Testo a mio favore" > Audisio.txt
```



```
studente@info2-17: ~/Documenti/componentiGruppo
studente@info2-17:~/Documenti/componentiGruppo$ echo "Ciao io sono Audisio 21/03/2024" > Audisio.txt
```

4. Controllare tutti i file presenti nella directory.

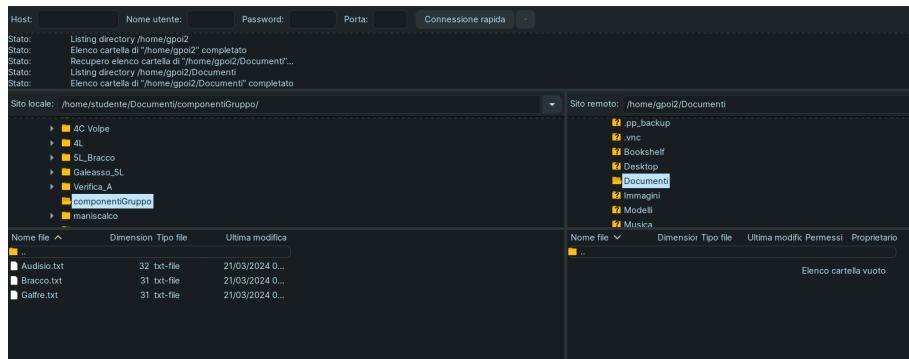
```
ls
```



```
studente@info2-17: ~/Documenti/componentiGruppo
studente@info2-17:~/Documenti/componentiGruppo$ ls
Audisio.txt Bracco.txt Galfre.txt
studente@info2-17:~/Documenti/componentiGruppo$ |
```

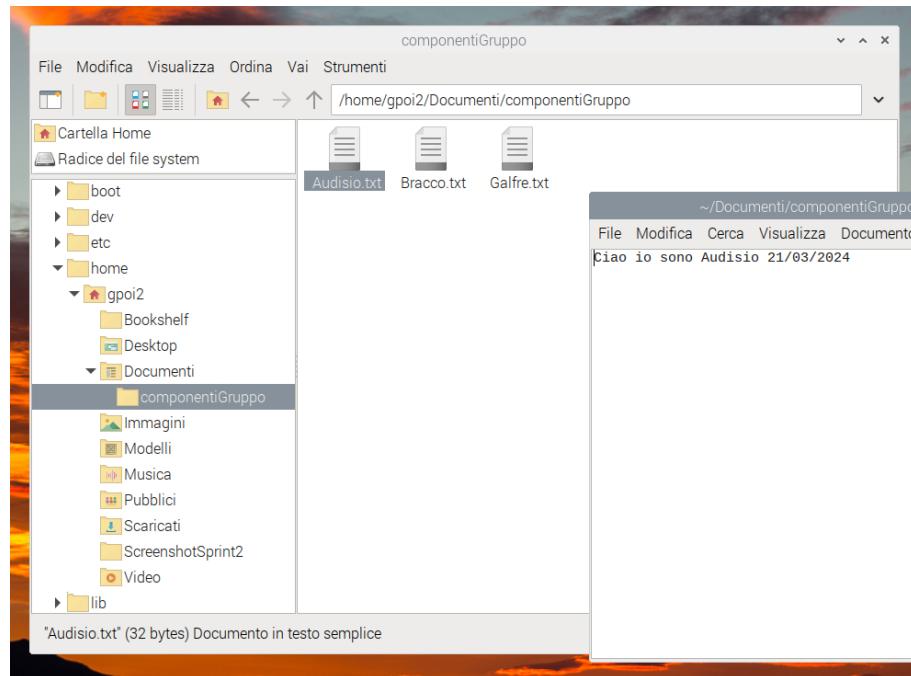
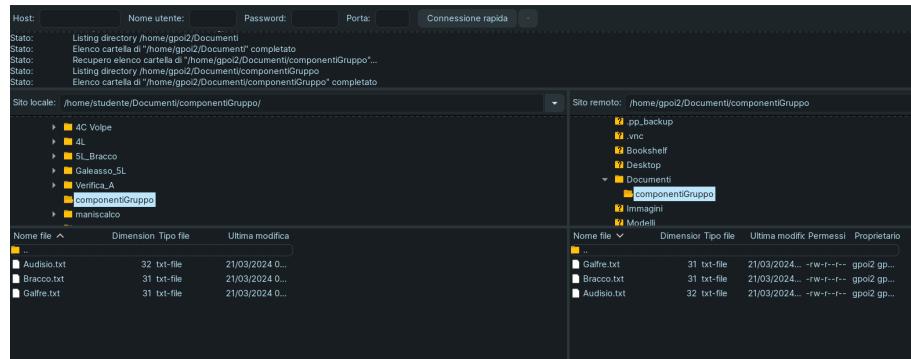
5. Avviare FileZilla e connettersi in *SFTP* (porta 22) al Raspberry.

- 6.
- Nella finestra di sinistra selezionare la directory di origine (Computer) da cui trasferire i file.
 - Nella finestra di destra selezionare la directory di destinazione (Raspberry) in cui verranno trasferiti i file.



- Fare clic con il tasto destro sulla directory di origine e selezionare l'opzione *Upload*.

7. Dopo aver effettuato l'operazione sarà possibile trovare i file nella directory selezionata in precedenza.



8. Nel caso si volesse effettuare l'operazione inversa: selezionare nella finestra di destra la directory di origine (Raspberry) e in quella di sinistra la directory di destinazione (Computer); quindi premere il destro del mouse e selezionare l'opzione *scarica* dalla finestra di destra

6 Protocollo FTP

Il File Transfer Protocol (FTP) è un protocollo di rete che opera sul livello di applicazione del modello ISO/OSI ed è definito nella RFC 959. Il protocollo, definito originariamente nel 1971, consente di trasferire dati ripetutamente tra un dispositivo finale e un server ed è costruito su un'architettura client-server.

6.1 Funzioni

Il protocollo FTP definisce il modo in cui i dati devono essere trasferiti su una rete TCP/IP. Il suo obiettivo principale è consentire la condivisione di file tra macchine remote, permettere l'indipendenza tra i file di sistema del client e del server, e abilitare un trasferimento di dati efficiente. FTP viene utilizzato per comunicare e trasferire file tra computer su una rete. Le funzioni principali di FTP includono il caricamento, il download e la manipolazione di file tra diversi computer. FTP viene spesso utilizzato nella gestione di siti web. Attraverso l'accesso FTP è possibile trasmettere file HTML a un server.

6.2 Modalità di connessione client/server

FTP può operare in una modalità attiva o passiva, che determina come viene stabilita una connessione dati. In entrambi i casi, un client crea una connessione di controllo TCP alla porta 21 del server FTP.

6.2.1 Modalità attiva

Nella modalità attiva, il client FTP apre una porta casuale sul proprio sistema e comunica al server questa porta tramite il comando *PORT*. Il server quindi stabilisce una connessione dati separata a questa porta per trasferire i file. Ecco i passaggi nello specifico:

1. **Connessione di controllo:** Il client stabilisce una connessione di controllo con il server FTP utilizzando la porta TCP 21.
2. **Comando PORT:** Il client invia il comando *PORT* al server, specificando l'indirizzo IP e il numero di porta casuale su cui è in ascolto per la connessione dati.
3. **Apertura porta dati:** Il client apre la porta indicata nel comando *PORT* e attende la connessione dal server.

4. **Connessione dati:** Il server stabilisce una connessione dati separata alla porta specificata dal client.
5. **Trasferimento dati:** I file vengono trasferiti tra il client e il server tramite la connessione dati.
6. **Chiusura connessioni:** Al termine del trasferimento, entrambe le connessioni (controllo e dati) vengono chiuse.

6.2.2 Modalità passiva

Nella modalità passiva, il server apre una porta casuale sul proprio sistema e fornisce al client l'indirizzo IP e il numero di porta tramite il comando *PASV*. Il client quindi stabilisce una connessione dati separata a questo indirizzo e porta per trasferire i file.

Ecco i passaggi nello specifico:

1. Connessione di controllo: Il client stabilisce una connessione di controllo con il server FTP utilizzando la porta TCP 21.
2. **Comando PASV:** Il client invia il comando *PASV* al server, richiedendo l'indirizzo IP e il numero di porta che il server utilizzerà per la connessione dati.
3. **Risposta PASV:** Il server risponde con un messaggio *PASV* contenente l'indirizzo IP e il numero di porta casuale che ha aperto per la connessione dati.
4. **Apertura porta dati:** Il client apre una porta sul proprio sistema (diversa da quella utilizzata per la connessione di controllo) e memorizza l'indirizzo IP e il numero di porta.
5. **Connessione dati:** Il client stabilisce una connessione dati separata all'indirizzo IP e al numero di porta forniti dal server nel messaggio *PASV*.
6. **Trasferimento dati:** I file vengono trasferiti tra il client e il server tramite la connessione dati.
7. **Chiusura connessioni:** Al termine del trasferimento, entrambe le connessioni (controllo e dati) vengono chiuse.

6.2.3 Principali differenze

La differenza fondamentale tra le modalità attiva e passiva risiede in chi avvia la connessione dati:

- **Modalità attiva:** Il client apre una porta e il server si connette ad essa.
- **Modalità passiva:** Il server apre una porta e il client si connette ad essa.

La modalità passiva è generalmente preferita perché:

- **Supera i problemi di firewall:** La modalità attiva può avere problemi con i firewall, che potrebbero bloccare le connessioni in entrata dal server. La modalità passiva evita questo problema in quanto il client avvia la connessione dati in uscita.
- **Maggiore sicurezza:** La modalità passiva è considerata leggermente più sicura in quanto il server non è esposto a connessioni in entrata non richieste.

Tuttavia, la modalità attiva potrebbe essere necessaria in alcuni casi, ad esempio quando il client si trova dietro un firewall restrittivo che non consente connessioni dati in uscita.

Caratteristica	Modalità attiva	Modalità passiva
Inizio connessione dati	Client	Server
Specifiche porta	Client	Server
Problemi con firewall	Possibili	Generalmente minore
Sicurezza	Potenzialmente inferiore	Considerata leggermente superiore

6.3 Modalità di accesso

Per trasferire file su un server FTP, un utente autorizzato deve stabilire una connessione con il server e richiedere l'accesso ai file. Esistono due modalità di connessione FTP comunemente utilizzate: attiva e passiva.

6.4 Comandi SFTP

SFTP condivide alcuni comandi di base con la shell di Linux per la navigazione e la gestione di file, ma aggiunge funzionalità specifiche per il trasferimento di file su una connessione sicura. Ecco alcuni comandi SFTP comuni:

6.4.1 Connessione

sftp [username]@[host]

Stabilisce una connessione SFTP con un server specificando lo username e l'host.

6.4.2 Gestione directory

• **cd [directory]**

Cambia la directory di lavoro sul server remoto.

• **pwd**

Visualizza la directory di lavoro corrente sul server remoto.

• **ls [directory]**

Elenca i file e le directory nella directory specificata sul server remoto.

• **mkdir [directory]**

Crea una nuova directory sul server remoto.

• **rmdir [directory]**

Rimuove una directory vuota sul server remoto.

6.4.3 Trasferimento file

• **get [remoteFile] [localfile]**

Scarica un file dal server remoto alla directory locale. Se non viene specificato alcun nome di file locale, verrà utilizzato il nome del file remoto.

• **put [localFile] [remoteFile]**

Carica un file dalla directory locale al server remoto. Se non viene specificato alcun nome di file remoto, verrà utilizzato il nome del file locale.

• **lcd [directory]**

Cambia la directory di lavoro locale.

6.4.4 Permessi e ownership

- **chmod [mode] [file]**

Cambia le autorizzazioni di un file sul server remoto.

- **chown [owner] [file]**

Cambia il proprietario di un file sul server remoto.

6.4.5 Informazioni sul file

- **ls -l [file]**

Visualizza informazioni dettagliate su un file sul server remoto, inclusi permessi, proprietario, gruppo e dimensioni.

6.4.6 Connessione e chiusura

- **help**

Visualizza una lista di comandi SFTP disponibili.

- **exit o quit**

Chiude la connessione SFTP e termina il programma.

6.5 Differenze tra FTP, TFTP, SFTP, FTPS

6.5.1 FTP (File Transfer Protocol)

FTP (File Transfer Protocol) è un protocollo di rete standard per il trasferimento di file tra un client e un server su una rete TCP/IP. Viene utilizzato per condividere, aggiornare e gestire file su computer remoti. FTP è un protocollo relativamente semplice e ampiamente utilizzato, che lo rende una scelta popolare per il trasferimento di file in vari contesti.

6.5.2 Caratteristiche principali di FTP

- **Trasferimento file bidirezionale:** FTP consente il trasferimento di file bidirezionale tra client e server, ovvero i file possono essere scaricati dal server al client o caricati dal client al server.
- **Supporto per directory tree navigation:** FTP consente la navigazione nella directory tree del server, permettendo agli utenti di individuare, selezionare e trasferire file specifici.

- **Autenticazione e autorizzazioni:** FTP supporta meccanismi di autenticazione per consentire l'accesso solo agli utenti autorizzati. Può essere utilizzato il controllo dell'accesso basato su nome utente e password o su token di autenticazione.
- **Modalità di trasferimento:** FTP offre diverse modalità di trasferimento per ottimizzare le prestazioni in diverse situazioni. Le modalità comuni includono:
 - **Modalità attiva:** Il client apre una porta casuale e il server si connette ad essa per la connessione dati.
 - **Modalità passiva:** Il server apre una porta casuale e il client si connette ad essa per la connessione dati.
 - **Modalità attiva con NAT:** Utilizza tecniche specifiche per superare i problemi di NAT (Network Address Translation) quando si utilizza la modalità attiva.
- **Supporto per la ripresa del trasferimento:** FTP può riprendere i trasferimenti interrotti, evitando la necessità di ricominciare da capo il download o l'upload di un file parziale.
- **Tipi di file:** FTP supporta il trasferimento di una varietà di tipi di file, inclusi documenti, immagini, video, software e altro ancora.

6.5.3 Funzionamento di FTP

FTP utilizza due connessioni separate per la comunicazione:

- **Connessione di controllo:** Una connessione TCP viene stabilita sulla porta 21 per il controllo e la gestione del trasferimento dei file. Attraverso questa connessione, il client invia comandi al server e il server invia risposte e informazioni sullo stato.
- **Connessione dati:** Una seconda connessione viene stabilita su una porta diversa (definita durante la sessione di controllo) per il trasferimento effettivo dei dati del file. Questa connessione può essere in modalità attiva o passiva, a seconda della configurazione e delle esigenze.

6.5.4 Casi d'uso di FTP

FTP è ampiamente utilizzato in vari scenari, tra cui:

- **Download e upload di file:** FTP è una scelta comune per scaricare file da siti web, caricare file su server web, trasferire file tra computer e altro ancora.
- **Sviluppo web:** FTP viene spesso utilizzato per caricare file HTML, CSS, JavaScript e altri file di origine sui server web. Backup e ripristino di file: FTP può essere utilizzato per creare backup di file e cartelle importanti su un server remoto per scopi di ripristino in caso di guasti o perdita di dati.
- **Distribuzione di software:** FTP è comunemente utilizzato per distribuire file di installazione di software, aggiornamenti e patch.
- **Accesso a file remoti:** FTP consente di accedere e gestire file su un computer remoto come se fossero sul proprio computer locale.

6.5.5 Vantaggi di FTP

- **Trasferimento file bidirezionale:** Consente il trasferimento di file in entrambe le direzioni, facilitando la condivisione e la sincronizzazione dei file.
- **Supporto per directory tree navigation:** Offre una navigazione intuitiva nella struttura delle directory del server, facilitando la ricerca e la selezione dei file.
- **Autenticazione e autorizzazioni:** Consente di controllare l'accesso ai file e alle cartelle, limitando l'accesso agli utenti autorizzati.
- **Ripresa del trasferimento:** Può riprendere i trasferimenti interrotti, evitando la perdita di tempo e dati.
- **Supporta vari tipi di file:** Può trasferire una vasta gamma di tipi di file, inclusi documenti, immagini, video, software e altro ancora.
- **Semplice e ampiamente diffuso:** FTP è un protocollo relativamente semplice da comprendere e utilizzare, con un'ampia base di supporto per client e server.

6.5.6 Svantaggi di FTP

- **Mancanza di crittografia di default:** Le connessioni FTP in chiaro non sono crittografate, rendendo i dati vulnerabili a intercettazione e sniffing.

6.6 SFTP (SSH File Transfer Protocol)

SFTP (SSH File Transfer Protocol) è un protocollo di rete sicuro per il trasferimento di file, file system e operazioni di gestione directory su una rete TCP/IP. Si basa sul protocollo SSH (Secure Shell) per fornire un canale di comunicazione crittografato per il trasferimento dei file. SFTP è un'alternativa sicura a FTP (File Transfer Protocol) che presenta rischi per la sicurezza poiché le sue connessioni non sono crittografate.

6.6.1 Caratteristiche principali di SFTP

- **Sicurezza basata su SSH:** SFTP sfrutta l'infrastruttura crittografica SSH per proteggere la comunicazione. Tutte le comunicazioni, inclusi comandi di controllo e dati dei file, vengono crittografati, garantendo la riservatezza e l'integrità dei dati trasferiti.
- **Autenticazione sicura:** SFTP utilizza metodi di autenticazione SSH standard, come chiavi pubbliche/private e password, per verificare l'identità del client e del server.
- **Supporto per operazioni di file system:** Oltre al trasferimento di file, SFTP consente operazioni di gestione directory come la creazione, la ridefinizione, l'eliminazione e l'aggiornamento dei permessi di file e directory sul server remoto.
- **Compatibilità con i client SFTP:** La maggior parte dei client FTP moderni supportano anche SFTP, offrendo agli utenti la possibilità di trasferire file in modo sicuro senza dover utilizzare software aggiuntivo.
- **Modalità di trasferimento:** SFTP eredita le modalità di trasferimento di FTP, come la modalità attiva e passiva, per stabilire connessioni dati.

6.6.2 Funzionamento di SFTP

SFTP funziona stabilendo due connessioni separate su un singolo canale SSH:

- **Connessione SSH:** SFTP opera come un sottosistema SSH. La connessione SSH viene stabilita sulla porta TCP 22 e fornisce un tunnel sicuro per tutte le comunicazioni SFTP.
- **Connessione SFTP:** All'interno del tunnel SSH crittografato, viene stabilita una seconda connessione virtuale per trasferire comandi di controllo e dati dei file.

6.6.3 Casi d'uso di SFTP

SFTP è ampiamente utilizzato in scenari che richiedono trasferimenti di file sicuri, tra cui:

- **Sviluppo web:** SFTP è il metodo preferito per caricare file di codice sorgente, risorse web e altri file sensibili su server web.
- **Trasferimento di dati aziendali:** SFTP è utilizzato per trasferire dati sensibili tra aziende o filiali in modo sicuro e conforme.
- **Gestione di server remoti:** SFTP consente l'accesso e la gestione sicuri di file e directory su server remoti.
- **Backup e ripristino di file:** SFTP può essere utilizzato per creare backup crittografati di file e cartelle su server remoti per scopi di ripristino in caso di guasti o perdita di dati.

6.6.4 Vantaggi di SFTP

- **Sicurezza:** Le connessioni SFTP crittografate proteggono i dati da intercettazione e sniffing durante il trasferimento.
- **Autenticazione sicura:** I metodi di autenticazione SSH garantiscono l'accesso autorizzato ai server remoti.
- **Supporto per operazioni di file system:** Consente la gestione completa di file e directory sul server remoto.
- **Compatibilità client:** Molti client FTP supportano anche SFTP, offrendo agli utenti una soluzione sicura e versatile.
- **Integrità dei dati:** La crittografia assicura che i dati trasferiti rimangano inalterati durante il trasferimento.

6.6.5 Svantaggi di SFTP

- **Configurazione leggermente più complessa:** Potrebbe richiedere una configurazione iniziale aggiuntiva rispetto a FTP per stabilire connessioni SSH e gestire le chiavi.
- **Potenziale overhead:** La crittografia può introdurre un leggero overhead rispetto a FTP non crittografato, sebbene le differenze di prestazioni siano generalmente trascurabili per trasferimenti di file di dimensioni standard.

6.7 FTPS (FTP Secure)

FTPS (FTP Secure) è un'estensione del protocollo FTP (File Transfer Protocol) che aggiunge un livello di crittografia per proteggere i dati trasferiti. Utilizza il protocollo TLS/SSL (Transport Layer Security/Secure Sockets Layer) per crittografare la comunicazione tra client e server FTPS, garantendo riservatezza e integrità dei dati. FTPS offre un'alternativa sicura a FTP standard, che è vulnerabile all'intercettazione e alla manomissione dei dati poiché le sue connessioni non sono crittografate.

6.7.1 Caratteristiche principali di FTPS

- **Sicurezza basata su TLS/SSL:** FTPS utilizza TLS/SSL per crittografare tutte le comunicazioni, inclusi comandi di controllo e dati dei file, proteggendoli da intercettazione e sniffing durante il trasferimento.
- **Modalità implicite ed esplicite:** FTPS supporta due modalità di connessione:
- **Implicita (FTPES):** Il client si connette automaticamente alla porta TCP 990, presupponendo che il server FTPS sia in esecuzione su quella porta e abilitando la crittografia.
- **Esplicita (FTPSA):** Il client stabilisce una connessione iniziale non crittografata con il server FTPS (generalmente sulla porta TCP 21) e quindi invia un comando esplicito per richiedere la crittografia TLS/SSL.
- **Autenticazione:** FTPS eredita i meccanismi di autenticazione standard di FTP, basati su username e password o certificati client.

- **Compatibilità del client:** La maggior parte dei client FTP moderni supportano anche FTPS, consentendo agli utenti di trasferire file in modo sicuro senza dover utilizzare software aggiuntivo.
- **Modalità di trasferimento:** FTPS eredita le modalità di trasferimento di FTP, come la modalità attiva e passiva, per stabilire connessioni dati.

6.7.2 Funzionamento di FTPS

FTPS funziona stabilendo due connessioni separate:

- **Connessione di controllo:** In modalità esplicita, il client prima stabilisce una connessione iniziale non crittografata con il server FTPS (solitamente sulla porta TCP 21). Successivamente, invia un comando per richiedere la crittografia TLS/SSL. In modalità implicita, il client si connette direttamente alla porta TCP 990, presupponendo che il server FTPS sia in esecuzione e abilitato per la crittografia.
- **Connessione dati crittografata:** Una volta stabilita la crittografia TLS/SSL, viene stabilita una seconda connessione per trasferire comandi di controllo e dati dei file in modo crittografato.

6.7.3 Casi d'uso di FTPS

FTPS è utilizzato in scenari che richiedono trasferimenti di file sicuri, tra cui:

- **Sviluppo web:** FTPS è un'opzione per caricare file di codice sorgente, risorse web e altri file sensibili su server web, fornendo un'alternativa a SFTP.
- **Trasferimento di dati aziendali:** FTPS può essere utilizzato per trasferire dati sensibili tra aziende o filiali in modo sicuro e conforme. Gestione di server remoti: FTPS consente l'accesso e la gestione sicuri di file e directory su server remoti.
- **Backup e ripristino di file:** FTPS può essere utilizzato per creare backup crittografati di file e cartelle su server remoti per scopi di ripristino in caso di guasti o perdita di dati.

6.7.4 Differenze tra FTPS e SFTP

Caratteristica	FTPS	SFTP
Protocollo di base	FTPS	SFTP
Crittografia	TLS/SSL	SSH
Porte predefinite	990 (implicita)	21 (esplicita) 22
Modalità di connessione	Implicita o esplicita	Nessuna
Gestione file system	Limitata	Estesa
Compatibilità client	Ampia	Ampia (con supporto SFTP)

6.8 TFTP (Trivial File Transfer Protocol)

TFTP (Trivial File Transfer Protocol) è un protocollo di rete semplice e leggero progettato per il trasferimento di file su una rete locale. È una versione semplificata di FTP (File Transfer Protocol) che elimina funzionalità come autenticazione, crittografia e directory tree navigation. Di conseguenza, TFTP è più veloce e più efficiente di FTP per il trasferimento di file di piccole dimensioni all'interno di una rete locale.

6.8.1 Caratteristiche principali di TFTP

- **Semplicità:** TFTP ha un set di comandi ridotto e un'architettura minimalista, che lo rende facile da implementare e da utilizzare.
- **Velocità:** L'assenza di autenticazione e crittografia consente trasferimenti di file più rapidi rispetto a FTP.
- **Leggerezza:** TFTP ha un overhead minimo, rendendolo ideale per dispositivi con risorse limitate come router e stampanti di rete.
- **Trasferimento di file di piccole dimensioni:** TFTP è ottimizzato per il trasferimento di file di piccole dimensioni, come file di configurazione, immagini e firmware.
- **Trasmissione unidirezionale:** TFTP supporta solo il trasferimento unidirezionale di file, dal server al client o viceversa.
- **Nessun supporto per la directory tree navigation:** TFTP non consente la navigazione nella directory tree del server. I file devono essere specificati per nome completo.

6.8.2 Funzionamento di TFTP

TFTP utilizza la porta UDP 69 per la comunicazione tra client e server. Il protocollo segue un modello di richiesta-risposta, in cui il client invia una richiesta al server e il server risponde con i dati richiesti o con un messaggio di errore.

6.8.3 Operazione TFTP

- **Read request:** Il client invia una richiesta al server per leggere un file. Il server risponde inviando il file al client.
- **Write request:** Il client invia una richiesta al server per scrivere un file. Il server apre il file sul proprio sistema e attende i dati dal client. Il client quindi invia i dati al server, che li scrive nel file.

6.8.4 Casi d'uso di TFTP

TFTP è comunemente utilizzato in una varietà di scenari di rete locale, tra cui:

- **Configurazione di dispositivi di rete:** TFTP viene spesso utilizzato per trasferire file di configurazione a router, switch e altri dispositivi di rete.
- **Aggiornamenti del firmware:** TFTP può essere utilizzato per aggiornare il firmware di dispositivi come router, stampanti e telefoni IP.
- **Trasferimento di file di registro:** TFTP può essere utilizzato per trasferire file di registro da dispositivi di rete a un server centrale per l'analisi.
- **Download di file di avvio:** TFTP può essere utilizzato per scaricare file di avvio su dispositivi senza un sistema operativo installato.

6.8.5 Differenze tra TFTP e FTP

Caratteristica	TFTP	FTP
Autenticazione	Nessuna Richiede	username e password
Crittografia	Nessuna Supporta	crittografia SSL/TLS
Directory tree navigation	No	Sì
Trasferimento dati	Unidirezionale	Bidirezionale
Overhead	Basso	Alto
Dimensioni file	Piccole	Qualsiasi dimensione
Casi d'uso	Rete locale	Rete locale ed Internet