# Raspberry Pi 3 Configuration Log

**Subject:** Server Environment Setup for Custom Cloud Application

**Hardware Components:** Raspberry Pi 3, Micro SD Card, USB Flash Drive

## Overview

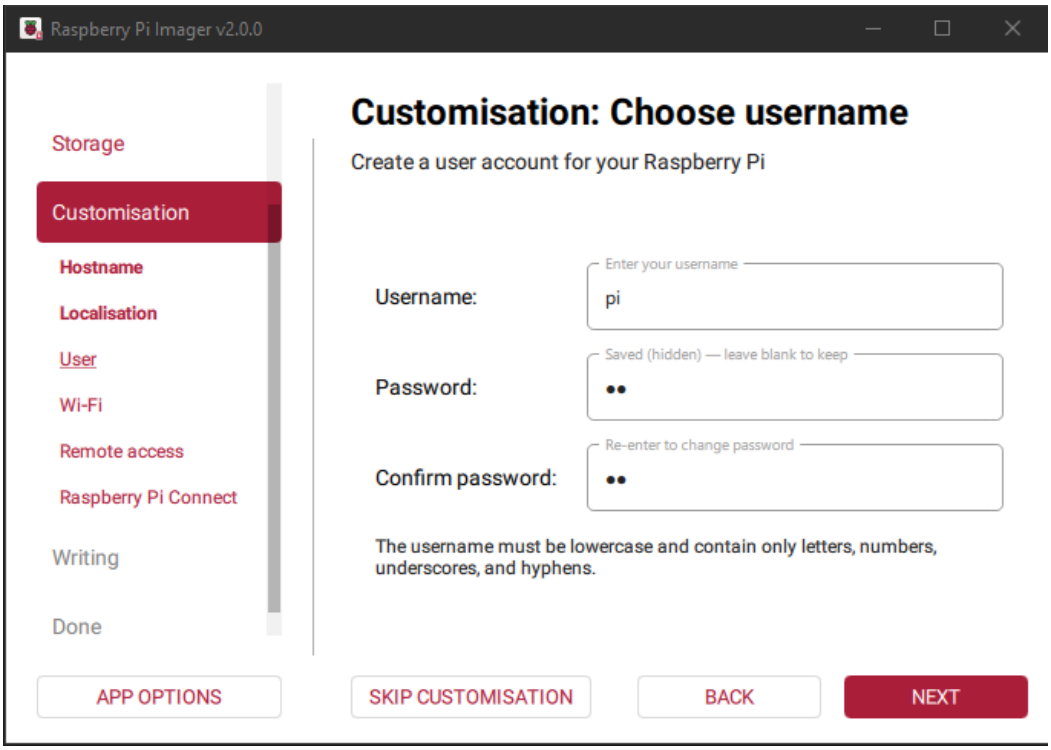
This document outlines the step-by-step procedure executed to configure a Raspberry Pi 3 as a headless server. The goal of this configuration is to prepare the environment for hosting a custom-built cloud storage service. The setup includes OS installation, remote access configuration, web server deployment, external storage mounting, and memory optimization via ZRAM.

## 1. Operating System Flashing

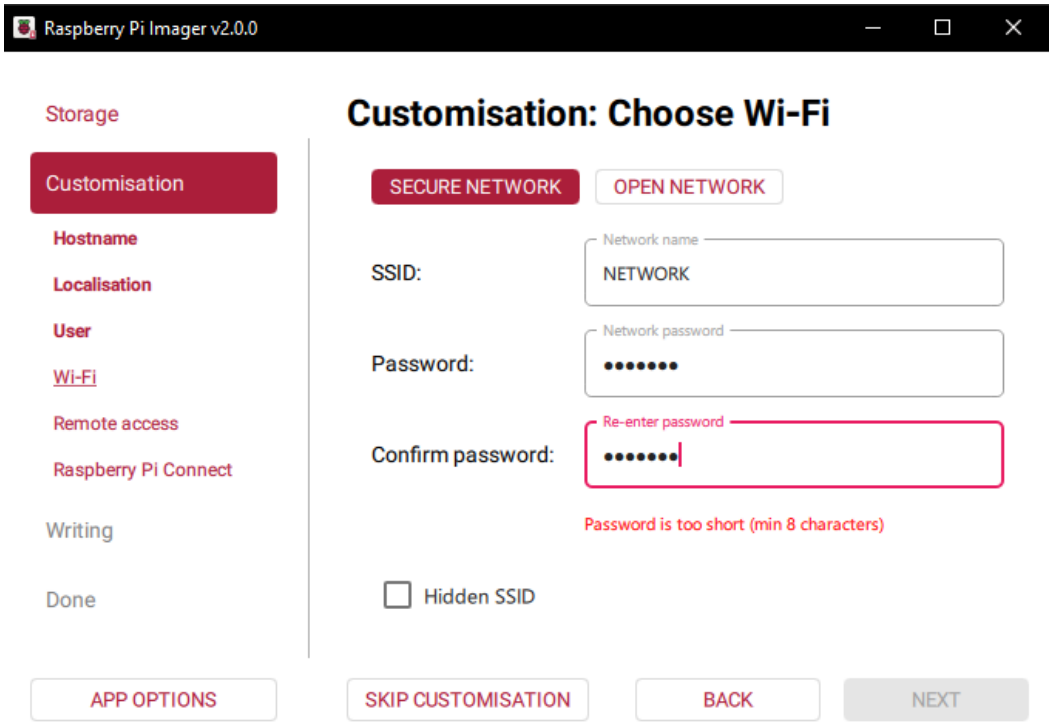
**Tool Used:** Raspberry Pi Imager **OS Selected:** Raspberry Pi OS Lite (64-bit)

The operating system was flashed onto the Micro SD card using the Raspberry Pi Imager. Before writing the image, the "Advanced Options" menu was accessed to pre-configure the following settings for a headless boot:

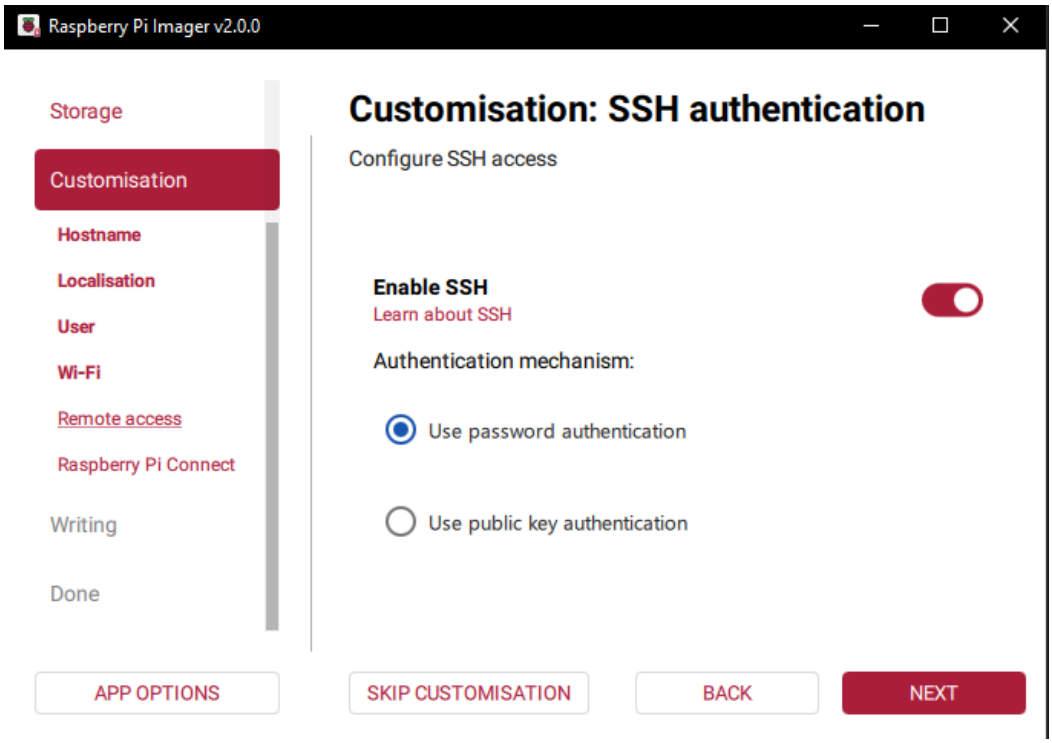
- **Hostname:** Set to a unique identifier for the network.



- **Network:** Configured Wi-Fi credentials (SSID and Password) to ensure immediate network connectivity upon boot.




**SSH:** Enabled SSH explicitly with password authentication.



## 2. Initial Boot and Remote Connection

After flashing, the SD card was inserted into the Raspberry Pi 3, and the device was powered on.

- **Connection Method:** SSH (Secure Shell) via laptop terminal.
- **Command Executed:** ssh (Ip address of the Raspberry Pi)
- **Result:** Successfully established a secure remote connection to the Raspberry Pi command line interface.

## 3. System Maintenance

To ensure system security and software compatibility, the package repositories and installed packages were updated to their latest versions immediately after login.

**Commands Executed:**

- sudo apt update
- sudo apt upgrade -y

## 4. Web Server Deployment (Nginx)

Nginx was selected and installed to serve the custom cloud application.

- **Installation:** Installed via the package manager using sudo apt install nginx -y.
- **Service Status:** Verified that the Nginx service was active and enabled to start on boot using systemctl status nginx.

## 5. External Storage Configuration

A USB flash drive was mounted to serve as the primary data store for the cloud application, separating user data from the boot OS.

- **Identification:** Used lsblk and sudo blkid to identify the partition name and UUID of the USB drive.
- **Mount Point:** Created a dedicated directory for the mount (e.g., /mnt/cloud_drive).
- **Persistence:** Edited the filesystem table (/etc/fstab) to ensure the drive mounts automatically upon system reboot.
    - *Configuration added:* /dev/sda1 /mnt/usb exfat defaults,nofail 0 0
- **Verification:** Mounted the drive using sudo mount -a and verified accessibility.

# 6. Memory Optimization (ZRAM)

To address the 1GB RAM limitation of the Raspberry Pi 3, ZRAM was configured to provide compressed swap space in memory.

- **Installation:** Installed the zram-tools package.
- **Configuration:** Edited /etc/default/zramswap to set the memory allocation limit.
    - *Setting changed:* PERCENTAGE=50 (Allocating 50% of total RAM for compressed swap).
- **Execution:** Restarted the ZRAM service.
- **Verification:** Confirmed the creation of the swap device /dev/zram0 and its priority using zramctl and cat /proc/swaps.

**Status:** The Raspberry Pi is now fully configured, optimized, and ready for the deployment of the custom cloud application code.