

Spectrum Survival Admin Guide

A basic guide created for a "Spectrum Survival" server admin

The guide is written for version 1.1

- Hosting the server
- Configuring The Server
 - Basic Setup
 - Additional Setup
 - Multi-version/Bedrock support

Hosting the server

How to download the Server?

To download the server you need to:

1. Have Java 21 installed (See FAQ below)
2. Download the latest release from [Github Releases](#) (Select "Server.zip", not "Source Code")
3. Unzip it

How to run the Server?

On a PC/Server:

1. Run the server inside a Terminal using:

```
java -jar SpectrumSurvival.jar
```

2. Join the server on the Minecraft version it's showing in the terminal output

With a Minecraft Hosting Provider:

Most providers allow you to upload something called "Custom JAR". This is exactly what Spectrum Survival is. Follow your Hosting's instructions on how to do it. [Here's Shockbyte for example.](#)

FAQ:

How to install Java 21?

To download and install Java 21 you need to:

1. Go to [this URL](#) and download the version for your system (Linux, macOS, Windows | x64, ARM64)

Installer will probably be the best option (So for Windows it will be the "x64 Installer" option)

2. Run the downloaded installer and follow the instructions
3. After installer is done you *should* reboot your machine (PC/Server/VM/etc.)

How to check my IP?

On Linux/macOS:

Use this command inside a Terminal:

```
ifconfig
```

On Windows:

```
ipconfig
```

On a (Minecraft) Hosting:

↓ See below ↓

How should my IP look like:

It depends.

1. If you're running the server at home you need to look for a "Private IP". On Windows it will be **IPv4 Address....: YourIP** and on Linux/macOS **inet addr: YourIP**. It will most probably be something like **192.168.X.X** (but it can be different).
2. If you're running the server at some Minecraft Hosting, they'll have the IP somewhere in the admin panel.
3. If you're running the server somewhere else (Like a Linux VM Hosting), TODO

Configuring The Server

Basic Setup

To configure the server you can edit the `server.properties` file.

Don't write the values in quotes. For example write `server.mode=online` instead of `server.mode="online"`

Set listening IP and Port:

The IP and Port can be set using `server.ip` and `server.port` properties in the config file. In most cases you shouldn't change server's IP.

Set server's mode:

To set the server's mode you can use `server.mode`. There are 3 values that can be used:

1. **"online"** - The server will "ask" Mojang servers if the player joining the server actually bought Minecraft.
2. **"offline"** - The server will let anyone join. So player can have a "cracked"/"non-premium" account and still join the server.
3. **"velocity"** - This mode is used only with a Velocity Proxy. You can set `server.proxysecret` to **Velocity's secret**.
4. **"bungeecord"** - This mode is used only with a Bungeecord Proxy.

Additional Setup

Setting maximum server slots:

By default the server doesn't limit the player count, but you can change it if you want by setting `server.slots` to a `number`. (`0` is for unlimited)

Setting minimum/maximum game slots:

This settings allow to change the minimum and maximum amount of players in one lobby. To change them set:

1. `game.minslots` to a `number` of minimum amount of players.
2. `game.maxslots` to a `number` of maximum amount of players.

Setting wait time for game to start:

To set the wait time you need to set the `game.waittime` property to a `number`. (in seconds)

Bypassing minimum game slots requirement:

As an admin you can enable/disable the `/queue force` command. This command allows players to force start the game even when there's not enough players (`game.minslots`). To change it set `game.allowforce` to `"true"` or `"false"`

Saving player stats:

To save player statistics you'll need to set **stats.saving** to **"true"**.

Then you'll need to select a way of saving those statistics. The solutions are:

1. **A MariaDB Database** - *Server will save player data in an external MariaDB Database* - Set:
 1. **stats.type** to **"mariadb"**
 2. **stats.ip** to **database's IP**
 3. **stats.port** to **database's port**
 4. **stats.db** to the **database name (made specifically for Spectrum Survival)**
 5. **stats.user** to the **username** set in the database.
 6. **stats.pass** to the **password** set in the database.
2. **A MongoDB Database** (Added by: [whynotmax](#)) - *Server will save player data in an external MongoDB Database* - Set:
 1. **stats.type** to **"mongodb"**
 2. **stats.mongo-uri** to **"mongodb://username:password@databaseIP:databasePort/database"**
3. **A SQLite Database** - *Server will save player data in a .db file* - Set:
 1. **stats.type** to **"sqlite"**
 2. **stats.db** to a **filename** (like "players.db", the file doesn't need to exist, the server should create it by itself)
 3. Leave **stats.ip**, **stats.port**, **stats.user** and **stats.pass empty**.
4. **A Directory** - *Server will save player data in a directory in the server's root directory* - Set:
 1. **stats.type** to **"dir"**
 2. **stats.db** to a **directory name** (like "players", the directory doesn't need to exist, the server should create it by itself)
 3. Leave **stats.ip**, **stats.port**, **stats.user** and **stats.pass empty**.

Multi-version/Bedrock support

This was only **somewhat** tested but it should be fine

Multi-version support (ViaVersion):

By default Spectrum Survival supports only one version of Minecraft just like Vanilla servers. You can change it by using ViaVersion (either as a standalone proxy (ViaProxy) or on Velocity).

Bedrock support (GeyserMC):

Spectrum Survival is a **Java Edition only** server. But by using GeyserMC (either standalone, Geyser-ViaProxy, or on Velocity) you can change that.