How To Make A Gmod Server

Introduction: Why Host Your Own Garry's Mod Server?

Garry's Mod (GMod) has remained a sandbox phenomenon for over a decade, allowing players to create everything from simple contraptions to full-blown roleplaying worlds. While joining existing servers is fun, nothing compares to the freedom and control of running your own. Whether you want to build a private playground for friends, launch a dedicated roleplay community, or experiment with custom gamemodes, learning **how to make a GMod server** is the first step. This guide walks you through every stage—from choosing a hosting method to configuring your server for smooth, lag-free play.

Running your own GMod server isn't just about technical know-how; it's about crafting an experience. You decide the rules, the mods, and the atmosphere. With the right preparation, even a beginner can have a server up and running in under an hour. Let's dive into the practical steps.

Understanding the Basics: What You Need Before You Start

Before you begin setting up your Garry's Mod server, you need to understand the core requirements. A GMod server is simply a dedicated process that hosts the game world for other players to join. Here’s what you'll need:

- **A copy of Garry's Mod on Steam** (the server files come free with the game, but you need it in your library).
- **A computer capable of running the server** – either your own PC (if you have a solid internet connection) or a rented virtual server (VPS/dedicated).
- **Port forwarding knowledge** – players need to connect to your IP, so your router must allow traffic on the required ports.
- **Basic familiarity with command lines** – you'll be typing commands to start and manage the server.
- **Patience for troubleshooting** – mod conflicts and firewall issues are common.

If you plan to host a server for many players (say, 20+), a dedicated hosting provider is strongly recommended. For a small group of friends, hosting from your own machine is perfectly viable.

Method 1: Hosting a GMod Server on Your Own Computer

This is the most common way for newcomers. It's free and gives you full control. Here's a step-by-step breakdown of **how to make a GMod server** locally.

Step 1: Install the Dedicated Server Tools

Open your Steam library. In the "Tools" section, look for **"Garry's Mod Dedicated Server"** and install it. This downloads the server binaries separately from the game client. The installation folder is typically `C:\Program Files (x86)\Steam\steamapps\common\GarrysModDS` or similar.

Step 2: Configure Your Server

Inside the server folder, create a text file named `server.cfg`. This file contains all your server settings. A basic configuration might look like this:

```
hostname "My Awesome GMod Server"
sv_password ""
rcon_password "mypassword"
sv_contact "your@email.com"
sv_tags "sandbox,fun"
maxplayers 16
mp_forcecamera 0
sv_allowcslua 1
```

Save the file in `garrysmod/cfg/server.cfg`. You can edit it later to add gamemode-specific settings.

Step 3: Create a Launch Script

To start the server, you need a command line. Create a new text file in the main server folder (e.g., `start.bat`) and paste this line (adjust the path if needed):

```
"C:\Program Files (x86)\Steam\steamapps\common\GarrysModDS\srcds.exe" -console -game garrysmod +map gm_construct +maxplayers 16
```

Replace `gm_construct` with your preferred map. Double-click the batch file, and the server console will launch. It will take a minute to generate the map and become visible on the server list.

Step 4: Port Forwarding

For players outside your home network to join, you must forward ports on your router. GMod server uses UDP ports **27015** (game) and **27020** (SourceTV). Log into your router's admin panel, find port forwarding, and create rules for these ports pointing to your computer's local IP address (e.g., 192.168.1.10). If you have a firewall, allow `srcds.exe` through.

Method 2: Using a Game Server Hosting Provider

If you want 24/7 uptime, DDoS protection, and no strain on your home internet, renting a server is the best route. Many providers offer one-click setups for Garry's Mod. The process is simpler:

1. Choose a hosting company (e.g., GTXGaming, Shockbyte, OVH).
2. Select a plan based on player slots (4-64).
3. After payment, they'll give you a control panel (like TCAdmin or Pterodactyl).
4. Use the panel to install GMod, select your gamemode (Sandbox, DarkRP, TTT, etc.), set server.cfg variables, and start the server.
5. Share the IP and port (usually `ip:27015`) with your friends.

With hosting, you skip port forwarding and hardware management entirely. This is ideal for **how to make a GMod server** that stays online while you sleep.

Choosing and Configuring Game Modes

GMod's magic lies in its gamemodes. By default, the server runs "Sandbox," but you can switch to any custom mode. Here's how:

Installing Popular Game Modes

- **DarkRP** (roleplay) – Download from Steam Workshop or GitHub, place the folder in `garrysmod/addons` and set `gamemode darkrp` in server.cfg.
- **Trouble in Terrorist Town (TTT)** – Add the TTT addon similarly; then in console type `gamemode terrortown`.
- **Prop Hunt** – Requires the Prop Hunt gamemode addon.
- **Sandbox** – Already built-in.

Make sure to restart the server after adding new gamemodes and mods. Many developers include .cfg files for mode-specific settings.

Essential Server Commands for Control

Once the server is running, you can use the console for live management. Some useful commands:

- `kick [player name]` – Remove a player.
- `banid [steamid]` – Ban by Steam ID.
- `changelevel [map]` – Switch maps without restart.
- `exec server.cfg` – Reload your config file.
- `sv_gravity [value]` – Change physics gravity (default 600).

Optimizing Your GMod Server for Performance

A laggy server drives players away. Follow these tips to keep your server smooth:

Limit Addons and Mod Overlap

Too many addons create lag and conflicts. Use the Workshop or direct download only what's necessary. Check for duplicate content (e.g., two different weapon packs).

Tweak Performance cvars

Add these to your server.cfg to reduce CPU load:

```
sv_minrate 60000
sv_maxrate 100000
sv_mincmdrate 30
sv_maxcmdrate 66
sv_minupdaterate 30
sv_maxupdaterate 66
fps_max 300
```

These set bandwidth and tick rate. Lower values help slower connections.

Regular Map Rotation

Stick to optimized, workshop-friendly maps like `gm_flatgrass`, `gm_construct`, or custom roleplay maps. Avoid huge,

unoptimized maps for low-end players.

Hardware Recommendations

If self-hosting, ensure your PC has at least 4 GB of RAM free for 16 players. A wired Ethernet connection is mandatory. For hosted servers, choose a plan with sufficient CPU (at least 2 cores) and RAM (3-4 GB for modded games).

Securing Your Server: RCON, Whitelist, and Admin Mods

Unwanted troublemakers can ruin the experience. Protect your server with these measures:

- **RCON password** – Set in server.cfg; allows remote admin access via console.
- **Set a server password** (`sv_password "yourpassword"`) – for private servers.
- **Install ULX** – A powerful admin mod. Download it, add it to `addons`, then configure `ulx_groups` to assign admin privileges to specific Steam IDs.
- **Use a whitelist** – Some hosting panels allow IP whitelisting. For self-hosted, you can use server plugins to allow only certain Steam groups.
- **Ban evasion prevention** – Use Steam ID bans rather than IP bans, as IPs change often.

Common Issues and Troubleshooting

Even with careful setup, problems arise. Here are fixes for frequent pain points:

- **Server not showing in list** – Check port forwarding, ensure you're not behind a VPN, and verify the server uses `+sv_lan 0` in launch parameters.
- **"Connection failed after 4 retries"** – Often a firewall block. Disable Windows Firewall temporarily to test, then add exceptions for port 27015 UDP.
- **Addon "Failed to mount" errors** – Delete the corrupted addon folder and redownload; ensure addons are in the correct directory.
- **Server crashes on map change** – Usually a memory issue. Lower `maxplayers` or remove heavy addons.
- **High ping for players** – Move the server geographically closer to your players using hosting providers with multiple locations (New York, London, Sydney).

Advanced: Automating Server Restarts and Monitoring

For a 24/7 server, consider automation:

- **Batch restart scripts** – Use a .bat file that runs the server in a loop: `:start`
  `srcds.exe -console -game garrysmod +map gm_construct +maxplayers 16`
  `goto start`
- **Use a watchdog program** – Tools like "EasyCraft" or "Watchdog" can auto-restart the server if it crashes.
- **Set up a website or Discord bot** – Use APIs to show server status, player counts, and last restart time. This adds professionalism to your community.

Conclusion: Your GMod Server Awaits

Learning **how to make a GMod server** opens up a world of creative hosting possibilities. Whether you choose to host locally for a handful of friends or rent a dedicated machine for a bustling community, the process is rewarding. Remember to start simple: get a basic sandbox server running first, then gradually add gamemodes, mods, and admin tools. With the steps outlined above—installation, configuration, port forwarding, performance tuning, and security—you have everything needed to launch a stable, enjoyable Garry's Mod server.

Don't be afraid to experiment. The GMod community thrives on innovation, and your server could become the next favorite hangout. So fire up that