

The Moderator's Guide to Digital Ministry

Heart, Discernment & Technical Skills for Serving Your Streamer's Community

"Whoever wants to become great among you must be your servant."

— Mark 10:43 (ESV)

Introduction

A question has come up multiple times across Christian Twitch communities: "Can you do a training stream for moderators?" It's a great question, and it reveals something important actually.. Most mods are thrown into the role with a sword icon next to their name and very little guidance on what to actually do with it.

This document is designed to be a comprehensive training resource for moderators in digital ministry spaces, particularly Twitch streams and Discord servers that are focused on sharing Jesus Christ. It covers three essential areas: the heart and character a mod needs, the discernment required to handle difficult situations wisely, and the technical commands and tools available across the most common platforms and bots.

Whether you are a brand new mod who just got the sword, a seasoned veteran looking for a reference guide, or a streamer who wants to hand your mods a training manual.. this resource is for you! While this guide is written with Christian digital ministry in mind, the principles and technical references apply to any Twitch or Discord community.

For deeper study on many of the biblical principles covered here, visit forgeiblebot.com/community where you can download the full *Navigating Difficult Encounters in Digital Ministry* resource and other biblical examination documents referenced throughout this guide.

Part 1: The Heart of a Moderator

Before we talk about any commands or technical skills, we need to talk about something far more important: who you are when you hold that mod sword. Because the truth is, a mod with perfect technical knowledge, but a wrong heart will do more damage to a community than a mod who doesn't know a single slash command, but loves people well.

You Are an Extension of the Streamer

When you accepted the moderator role, you became a representative of that streamer and their community. Your words, your tone, your decisions.. they all reflect on the person who trusted you with that responsibility. In a Christian ministry stream, that weight is even heavier, because you are not just representing a content creator. You are representing Christ in a digital space.

Think about it this way: if the streamer is the shepherd of their community, the moderators are the sheepdogs. Your job is to protect the flock, keep order, and help guide things.. not to be another shepherd barking your own orders, and certainly not a wolf. You serve at the pleasure of the streamer. Their rules are your rules. Their vision is your vision. If you disagree with how they want something handled, have that conversation privately.. never in front of the community.

1 Peter 5:2–3 (ESV) — *“Shepherd the flock of God that is among you, exercising oversight, not under compulsion, but willingly, as God would have you; not for shameful gain, but eagerly; not domineering over those who are in your charge, but being examples to the flock.”*

Character Matters More Than Commands

The most important qualities of a good moderator have nothing to do with technology:

- **Self-control.** You will encounter people who are rude, provocative, and deliberately trying to get under your skin. If you cannot remain calm when provoked, you are not ready to moderate. James 1:19 is your operating manual: “Be quick to listen, slow to speak, and slow to become angry.”
- **Humility.** The mod sword is not a crown. It is a servant's towel. You are there to serve the community, not to be served by it. Never use your authority to win arguments, settle personal grudges, or show off.
- **Discernment.** Not every problem requires the same response. Wisdom is knowing when to engage, when to warn, when to timeout, and when to ban. We'll cover this extensively later in this guide.

- **Patience.** Some of the most fruitful conversations in digital ministry start with what looks like trolling. The person who comes in with a sarcastic question might be the person God is drawing. Don't slam the door before the Holy Spirit has a chance to work.
- **Consistency.** Enforce the rules the same way for everyone. Favorites and double standards destroy community trust faster than almost anything.
- **Confidentiality.** What happens in mod discussions stays in mod discussions. Period. If a streamer shares concerns about a viewer with you, or if you see sensitive information in prayer requests, that is not your information to share.

A Sound Mind in a Volatile Space

Digital ministry spaces can be emotionally taxing. You will see people express pain, anger, confusion, and hostility.. sometimes all in the same stream! If you are not in a healthy place mentally and spiritually, moderating can amplify your own struggles rather than serving the community.

Guard your own heart. Stay in the Word. Pray before every stream. If you're having a bad day and you know your patience is thin, it's okay to tell your streamer "Hey, I'm not in a great headspace tonight.. I might need to step back from actively modding this stream." That's not a weakness. That's wisdom.

Proverbs 4:23 (ESV) — *"Keep your heart with all vigilance, for from it flow the springs of life."*

The Mod's Prayer

Before every stream, consider praying something like this:

"Lord, help me serve this community well today. Give me eyes to see what You see in the people who show up. Give me patience when I want to react. Give me wisdom to know the difference between someone who needs grace and someone who needs a boundary. Help me represent You and this streamer faithfully. Let my words and actions bring You glory. Amen."

Part 2: Discernment in Action

This section draws heavily from the *Navigating Difficult Encounters in Digital Ministry* resource available at forgebiblebot.com/community. If you haven't read that document, do so. What follows here is a summary tailored specifically for moderators.

Know Who You're Dealing With

Not everyone who challenges the stream has bad intentions. Your first job as a mod is to correctly identify what you're dealing with before deciding how to respond. Here are the four main types you'll encounter:

Genuine Seekers are people who are curious about faith. Their questions may sound blunt or even confrontational, but they're actually listening. They engage with answers, ask follow-up questions, and are respectful even if skeptical. These are your opportunities. Do not shut them down.

Confused or Misinformed Believers are fellow Christians who hold positions they can't defend well or apply Scripture incorrectly. They are not the enemy, they are family. Correct the theology, not the person. Point to Scripture, not your opinion.

Provocateurs and "Gotcha" People come in with an agenda to trip the streamer up or catch them in a contradiction. They rapid-fire questions without engaging with answers. They use Bible verses as weapons, not as genuine inquiry. Respond with calm truth. Do not take the bait.

Trolls with No Interest in Dialogue are purely there to disrupt. Their messages have no substance. They escalate regardless of your response. They spam, use profanity, or make personal attacks. This is where boundaries are appropriate.

The Escalation Framework

This is a simple, biblically grounded process for handling difficult encounters. As a moderator, you will likely be the first person to engage with these situations.. often before the streamer even notices! Communicate with your streamer about this framework so you are aligned on how to handle things.

Step 1: Engage with Grace. Assume good faith unless there is clear evidence otherwise. Respond to the question or comment with kindness and truth. Many situations resolve here. The streamer may address it themselves, or a simple friendly response from a mod can set the right tone.

Step 2: Set Clear Expectations. If behavior shifts from questioning to disrupting—hostility, dismissiveness, or clear disinterest in dialogue.. set a calm

expectation. Something like: “Hey, we’re happy to discuss that, but let’s keep it respectful.”

Step 3: Give a Fair Warning. If the behavior continues, issue a direct and compassionate warning. This is the Titus 3:10 principle: “Warn a divisive person once, and then warn them a second time. After that, have nothing to do with them.” Let the person know that continued behavior will result in a timeout or removal.

Step 4: Follow Through with Boundaries. If the person persists after clear warnings, follow through. A timeout is usually the appropriate first enforcement action, it communicates “that **behavior** is not welcome here” without communicating “**you** are not welcome here.” Reserve permanent bans for extreme cases or repeated offenses.

Important: If someone is posting illegal content, explicit material, doxxing, or making threats, skip the escalation framework entirely. Immediate removal is appropriate and necessary. The framework above applies to difficult conversations and disruptive behavior, not dangerous behavior.

Common Mistakes Mods Make

- **Banning first, asking never.** An immediate ban without any engagement sends the message: “We don’t actually practice the grace we preach.” A ban may be necessary eventually, but it should not be the opening move.
- **Treating every challenge as an attack.** Not every hard question is spiritual warfare. Not every rude comment is the enemy at work. Sometimes people are just having a bad day. Sometimes they’re testing your authenticity.
- **Responding in anger.** The moment you lose your temper in chat, the message shifts from the gospel to your reaction. You are representing Christ. Guard that witness.
- **Trying to win the argument.** Your job is not to win debates. Your job is to maintain a healthy environment where truth can be shared. Sometimes the most Christ-like move is to say nothing and let the streamer handle it.
- **Going rogue.** Making unilateral decisions that don’t align with the streamer’s philosophy will undermine trust, both the streamer’s trust in you and the community’s trust in the mod team. When in doubt, check with the streamer.

Communicating with Your Streamer

Good moderation requires communication between the mod team and the streamer. Here are practical ways to stay aligned:

- Have a private mod channel in Discord where you can discuss situations in real-time during streams.
- Before your first “shift”, ask the streamer about their moderation philosophy. How do they want hard questions handled? At what point do they want you to timeout or ban? Do they prefer to handle confrontations themselves?
- Debrief after streams where difficult situations occurred. What went well? What would you do differently?
- If you’re unsure about a situation in the moment, it’s always better to DM the streamer and ask than to act on your own.

Part 3: The Technical Toolkit

Now let's get into the practical tools at your disposal. This section covers native Twitch commands that mods can use, as well as commands for the most common bots in the Christian streaming space: StreamElements, Nightbot, and Forge Bible Bot.

Twitch Native Mod Commands

These are built into Twitch and available to all moderators. They use the / prefix and are typed directly into Twitch chat. These commands do not require any bots, they are part of the Twitch platform itself.

User Moderation

Command	What It Does
<code>/ban <username></code>	Permanently bans a user from the channel. They cannot chat, view chat, or interact. Use sparingly, this is the nuclear option.
<code>/unban <username></code>	Removes a permanent ban, allowing the user to chat again.
<code>/timeout <user> <seconds></code>	Temporarily restricts a user from chatting. The default is 600 seconds (10 minutes). Example: <code>/timeout trolluser 300</code> gives a 5-minute timeout.
<code>/untimeout <username></code>	Removes an active timeout, allowing the user to chat immediately.
Delete message	Click the trash icon next to any message in chat or Mod View to delete it. This is the easiest and most common way to remove individual messages. You can also hover over a message and click the three-dot menu for additional options.
<code>/warn <username></code>	Issues an official warning to a user. The user receives a notification.
<code>/restrict <username></code>	Restricts a suspicious user, only the mods and broadcaster can see their messages. Good for suspected ban evaders.
<code>/monitor <username></code>	Similar to restrict, but the user's messages are highlighted for mods to review while still visible to everyone.

Chat Mode Controls

Command	What It Does
<code>/slow <seconds></code>	Enables slow mode, users must wait the specified seconds between messages. Great for managing fast-moving chat.
<code>/slowoff</code>	Disables slow mode.
<code>/followers [duration]</code>	Restricts chat to followers only. Optional duration (e.g., <code>/followers 10m</code> means they must have followed for 10 minutes). Excellent anti-troll measure.
<code>/followersoff</code>	Disables followers-only mode.
<code>/subscribers</code>	Restricts chat to subscribers only.

<code>/subscribersoff</code>	Disables subscriber-only mode.
<code>/emoteonly</code>	Only emotes can be posted in chat, no text messages.
<code>/emoteonlyoff</code>	Disables emote-only mode.
<code>/uniquechat</code>	Prevents the same message from being sent repeatedly. Good against spam bots.
<code>/uniquechatoff</code>	Disables unique chat mode.
<code>/clear</code>	Clears the entire chat history for all viewers. Useful during mass trolling events.

Community & Recognition

Command	What It Does
<code>/shoutout <username></code>	Gives a streamer a shoutout with a viewer notification card. Also written as <code>/so</code> . Great for welcoming raiders or recognizing fellow streamers.
<code>/announce <message></code>	Sends a highlighted announcement message in chat. Also available with colors: <code>/announceblue</code> , <code>/announcegreen</code> , <code>/announceorange</code> , <code>/announcepurple</code> .
<code>/requests</code>	Opens the Channel Points request queue so you can manage pending redemptions.

Mod View & User Cards

Twitch's Mod View is a dedicated moderation dashboard that gives you everything you need in one place. There are two ways to access it: click the sword icon at the bottom of the chat box, or go directly to twitch.tv/moderator/<channelname>. Mod View includes customizable widgets for chat, mod actions, AutoMod queue, activity feed, and more. You can rearrange, resize, and toggle widgets by clicking the pencil icon in the lower-left corner.

User Cards are one of the most powerful tools available to you. Click on any username in chat to open their user card. This shows you the user's account creation date (brand new accounts are often troll or bot accounts), their chat history in the channel, any previous bans or timeouts, and any mod comments that have been left by you or other moderators. This information helps you make informed decisions rather than reacting blindly.

Mod Comments (How to Use Them)

Mod Comments are channel-specific notes that moderators can leave on any user's card. They are visible to all moderators on that channel and persist between streams. This is an incredibly useful and underused feature. Here's how to use them:

- Click on a username in chat to open their user card.

- In the user card, look for the “Mod Comments” section (it may appear as a text field or expandable area below the user’s chat history and ban history).
- Type your note and submit it. The comment is now attached to that user’s card for this channel.
- All mods on the channel can see these comments when they open that user’s card.

Important note: As of early 2026, mod comments cannot be edited or deleted after they are posted. Be thoughtful about what you write. Keep notes factual and professional.. something like “Warned 4/10/26 for repeated off-topic disruption during Bible study” or “Genuine seeker, asked good questions about baptism on 3/15” rather than emotional reactions.

Pro tip: Make it a habit to leave a mod comment any time you warn or timeout a user. When another mod encounters that person in a future stream, they can see the history at a glance and respond appropriately. This is especially valuable in ministry contexts where you want to extend grace but also need to track patterns.

StreamElements Commands

StreamElements is one of the most popular cloud-based bots for Twitch. If your streamer uses StreamElements, here are the commands available to moderators. All StreamElements commands use the ! prefix.

Command Management

Command	What It Does
<code>!cmd add <name> <response></code>	Creates a new custom command. Example: <code>!cmd add !discord Join The Forge — our community for people pursuing Christ through worship, the Word, and sharpening each other. Jump in and grab your Forge Member role automatically! 🔥 https://discord.gg/4daR5acFFd</code>
<code>!cmd edit <name> <response></code>	Updates an existing custom command's response.
<code>!cmd remove <name></code>	Deletes a custom command.
<code>!cmd options <name> <flags></code>	Modify command settings (cooldown, user level, etc.). Example: <code>!cmd options !socials -cd=30 -ul=everyone</code>
<code>!cmd alias add <base> <alias></code>	Creates an alias for an existing command. Example: <code>!cmd alias add !socials !social !links</code>
<code>!cmd alias remove <base> <alias></code>	Removes an alias from a command.

Bot Control

Note: By default, these commands are available to moderators, but `!bot part` (which makes the bot leave entirely) is typically reserved for the broadcaster or super moderators. Your streamer may have customized permission levels differently on their StreamElements dashboard.

Command	What It Does
<code>!bot mute</code>	Silences the bot, it will not send any messages or respond to commands.
<code>!bot unmute</code>	Restores the bot to normal operation.
<code>!bot part</code>	Makes the bot leave the chat entirely. Typically broadcaster only, check with your streamer before using.
<code>!module [name] on/off</code>	Toggles specific modules (e.g., <code>!module roulette off</code>).

Community Engagement

Important: These commands only work if the streamer has the corresponding modules enabled and activated on their StreamElements dashboard. For example, if the streamer has not enabled the Raffle module, the `!raffle` command will not work. If a command is not responding, check with your streamer to see if that feature is turned on.

Command	What It Does
<code>!sraffle [duration]</code>	Starts a single-winner raffle. Viewers type <code>!join</code> to enter. Default 60 seconds.
<code>!raffle [duration]</code>	Starts a multi-winner raffle (winners scale with participants).
<code>!cancelraffle</code>	Cancels the current raffle.
<code>!queue open / close</code>	Opens or closes a viewer queue (great for prayer lines or Q&A).
<code>!queue next</code>	Selects the next viewer from the queue.
<code>!queue list / clear</code>	Lists or clears the queue.
<code>!giveaway</code>	Starts a giveaway with StreamElements' built-in system.
<code>!bingo <platform> <points></code>	Starts an emote bingo game.

Song Requests (if enabled)

Command	What It Does
<code>!songqueue</code>	Displays the current song request queue.
<code>!removesong</code>	Removes a specific song from the queue.
<code>!skipsong</code>	Skips the currently playing song.
<code>!volume <0-100></code>	Adjusts the song request volume.

Fun Custom Command Examples

StreamElements supports powerful variables in custom commands that let you create fun community interactions. Here are some examples your streamer might set up:

`!soup` — ``${sender} made a warm bowl of ${random.pick 'chicken noodle' 'tomato basil' 'loaded potato' 'French onion' 'clam chowder'} soup for ${1}!`

`!hug` — ``${sender} gives ${1} a big warm hug!`

`!coffee` — ``${sender} brewed a fresh ${random.pick 'espresso' 'latte' 'cold brew' 'cappuccino'} for ${1}!`

These community commands build camaraderie and make your chat feel like a warm, welcoming place. The ``${sender}`` variable inserts the username of whoever typed the command, ``${1}`` inserts the first word after the command (usually a target user), and ``${random.pick}`` selects randomly from a list you define.

StreamElements Variables for Custom Commands

When creating or editing custom commands, StreamElements supports powerful variables that make your responses dynamic:

- ``${sender}`` — The username of whoever typed the command.

- `{1}`, `{2}`, `{3}` — The first, second, third word the user typed after the command.
- `{channel}` — The channel name.
- `{random.pick 'opt1' 'opt2' 'opt3'}` — Picks a random item from a list you define.
- `{count}` — A counter that increments each time the command is used.
- `{uptime}` — How long the stream has been live.
- `{user.points}` — The user's current loyalty points.
- `{customapi.URL}` — Fetches data from an external URL.

Nightbot Commands

Nightbot is a classic, reliable cloud-based bot that many streamers use alongside or instead of StreamElements. If your streamer uses Nightbot, here are the key commands available to moderators. As with StreamElements, some features may need to be enabled by the streamer on the Nightbot dashboard before they will work.

Command Management

Command	What It Does
<code>!commands add <name> <response></code>	Creates a new custom command. Flags: -ul=userlevel, -cd=cooldown, -a=alias.
<code>!commands edit <name> <response></code>	Updates an existing command's response.
<code>!commands delete <name></code>	Deletes a custom command.

Built-in Default Commands (Mod-Accessible)

Command	What It Does
<code>!game</code>	Displays the current stream category/game.
<code>!game <title></code>	Changes the stream category.
<code>!title</code>	Displays the current stream title.
<code>!title <new title></code>	Changes the stream title.
<code>!permit <username></code>	Temporarily allows a user to post a link (bypasses link filter).
<code>!marker [description]</code>	Sets a stream marker from chat.
<code>!poll new <title> <opts></code>	Creates a Strawpoll for your community to vote on.
<code>!poll results</code>	Shows the results of the latest poll.
<code>!songs current / next</code>	Shows the currently playing or next song request.
<code>!songs delete</code>	Removes a song from the queue.
<code>!regulars add <user></code>	Adds a user to Nightbot's "regular" user level.
<code>!regulars delete <user></code>	Removes a user from the regular user level.
<code>!filters</code>	Manages spam protection filters (caps, links, symbols, emotes, repetition).

Nightbot Variables for Custom Commands

When creating custom commands, Nightbot supports powerful variables:

- `$(user)` — The username of whoever typed the command.
- `$(touser)` — The first word after the command, or the user's name if nothing is provided. Great for shoutouts.

- `$(count)` — A counter that increments each time the command is used.
- `$(query)` — Everything the user typed after the command name.
- `$(channel)` — The channel name.
- `$(twitch <user> "<template>")` — Fetches Twitch info like `{{game}}`, `{{url}}`, `{{name}}`.
- `$(eval <javascript>)` — Runs JavaScript for complex conditional responses.

Forge Bible Bot Commands

Forge Bible Bot is a purpose-built ministry platform for Christian Twitch and Discord communities. If your streamer uses FBB, these are the commands you should know. All commands use the ! prefix. For a complete and always up-to-date reference, visit forgebiblebot.com/guide.

Bible & Study Commands

Command	What It Does
<code>!verse <ref></code>	Look up any Bible verse or range (e.g., <code>!verse Romans 8:28-30</code>).
<code>!read <ref></code>	Read a full chapter.
<code>!search <words></code>	Search the Bible by keyword.
<code>!xref <ref></code>	Find cross-references for a verse.
<code>!random</code>	Get a random encouraging verse.
<code>!votd</code>	Today's Verse of the Day (builds streaks!).
<code>!next / !prev</code>	Navigate forward or backward through passages.
<code>!save / !saved</code>	Bookmark the last verse looked up / view your bookmarks.
<code>!translation <code></code>	Set your preferred Bible translation from 18 options (esv, niv, nkjv, nlt, nasb, kjv, amp, etc.).

Ministry & Community Commands

Command	What It Does
<code>!jesussaid</code>	Random red-letter quote from Jesus with text-to-speech on the OBS overlay.
<code>!proverb</code>	Biblical wisdom from Proverbs and wisdom literature.
<code>!gospel [lang]</code>	Share the Gospel message in 12 languages.
<code>!prayer <request></code>	Submit a prayer request (saved to the Prayer Wall and synced to Discord).
<code>!praise <message></code>	Share a praise report—celebrate what God is doing!
<code>!impact</code>	Community stats for the month: verses shared, trivia played, prayers, and praises.
<code>!trivia [difficulty]</code>	Start a trivia question (500 questions across easy, medium, and hard).
<code>!a <answer></code>	Answer the active trivia question.
<code>!leaderboard / !global</code>	Community trivia leaderboard or the global leaderboard across all FBB communities.

Stream Management (Mod & Broadcaster)

Command	What It Does
<code>!topic <ref></code>	Set a study topic that shows on the OBS overlay with reminders every 15 minutes.
<code>!topic clear</code>	Clear the current study topic.
<code>!q add <topic></code>	Add a discussion topic to the queue (great during Bible study for parking questions).
<code>!q next / !q done</code>	Activate the next queued topic or clear the current one.
<code>!q list / !q remove <#></code>	View the queue or remove a specific item.
<code>!q promote <#> / !q clear</code>	Move an item to position 1 or clear the entire queue.
<code>!timer <1-30></code>	Start a countdown timer on the OBS overlay (1–30 minutes).
<code>!timer stop</code>	Cancel the active timer.
<code>!toggle <feature> <setting></code>	Toggle features or set permissions during stream (on/off/mods/vips/subscribers/followers/everyone).

Auto-Detect: FBB also watches all chat messages for Bible references automatically, no command needed. If someone types “I love John 3:16” or just “Romans 8:28,” the bot detects the reference and responds with the verse. This works in both Twitch and Discord.

Team Access (For Mods Given Dashboard Permissions)

Streamers who use Forge Bible Bot can grant their moderators access to portions of the FBB dashboard through the Team Access feature. This is entirely optional and up to the streamer, not all streamers will choose to do this. If your streamer does invite you, you will receive a link to accept the invitation and will be given scoped permissions based on what the streamer has enabled for you. The four available permission levels are:

- Stream Controls — Access to the Stream Mode panel for toggling features on and off during the stream.
- Prayer Wall — Access to view, manage, and update the status of prayer requests submitted by the community.
- Settings — Access to configure bot settings, translations, and feature toggles.
- Scheduled Posts — Access to create and manage scheduled Discord posts.

If your streamer uses FBB and you are an active moderator, it may be worth asking if they would like to grant you Team Access—especially to the Prayer Wall, which allows you to help follow up on prayer requests submitted during streams.

Part 4: Best Practices for Moderators

Before the Stream

- Pray. Seriously. Ask God for wisdom, patience, and discernment.
- Check in with the streamer. Is there anything specific planned today? Any topics that might attract controversy?
- Open Mod View in Twitch and have your mod tools ready.
- If you use Discord for mod communication, have that channel open on a second monitor or device.
- Review any mod notes from previous streams for users you've had issues with.

During the Stream

- Stay engaged in chat. Your presence alone deters bad behavior.
- Welcome newcomers. A simple "Hey welcome in!" goes a long way in setting the tone.
- Watch for patterns, not just individual messages. A single off-color joke might be a lapse in judgment. A pattern of pushing boundaries is a problem.
- Let the streamer handle theological questions unless they ask for help or you have a clear green light to engage.
- If you need to timeout or ban someone during a sensitive moment (like a prayer segment), try to do it quietly and discreetly if possible.
- Use the discussion queue (!q add) to park good questions that come up so the streamer can address them when ready.

After the Stream

- Debrief with the streamer if anything notable happened.
- Leave mod notes on any users who were warned or timed out so the history is documented.
- Check the Prayer Wall if your streamer uses Forge Bible Bot! Prayer requests submitted during the stream deserve follow-up.
- Take care of yourself. If a stream was emotionally heavy, give yourself permission to decompress.

Handling Raids

Raids are when another streamer sends their viewers to your channel. The vast majority of raids are positive and friendly, especially in the Christian streaming community, raids are typically acts of love and support from fellow creators. Here's how to handle them:

- **Friendly raids (the norm):** Welcome the raiders warmly! You can use /shoutout or /so to recognize the raider and let the incoming viewers know what's happening on stream so they feel included. If the streamer uses Forge Bible Bot and has the Raid Welcome feature enabled (requires EventSub upgrade and the feature toggled on in their dashboard), FBB will also send an automatic welcome message.
- **Hostile raids (extremely rare):** In all honesty, hostile or "hate" raids are extremely uncommon, especially in the Christian streaming space. But it is good to know what to do just in case. If a wave of trolls or hateful messages floods chat after a raid, act quickly: use /clear to wipe the chat, enable /followers 10m or /subscribers to lock chat down temporarily, and report the raiding channel to Twitch. Stay calm! Your community is watching how you respond, and your composure will reassure them.

The Power of Your Presence

Here is something that might surprise you: some of the best moderation you'll ever do involves zero commands. Simply being present, engaged, and visible in chat as a moderator creates an environment where people feel safe and where bad actors think twice. Your consistent, kind, Christ-like presence is a more powerful moderation tool than any slash command.

When people see a mod who welcomes newcomers, answers questions with warmth, and handles conflict with grace, it sets the culture for the entire community. You're not just enforcing rules. You're modeling what it looks like to follow Jesus in a digital space.

Key Verses for Moderators

Mark 10:43 (ESV) — *“Whoever wants to become great among you must be your servant.”*

James 1:19–20 (ESV) — *“Everyone should be quick to listen, slow to speak and slow to become angry, because human anger does not produce the righteousness that God desires.”*

Proverbs 15:1 (ESV) — *“A gentle answer turns away wrath, but a harsh word stirs up anger.”*

2 Timothy 2:24–25 (ESV) — *“The Lord’s servant must not be quarrelsome but must be kind to everyone, able to teach, not resentful. Opponents must be gently instructed, in the hope that God will grant them repentance leading them to a knowledge of the truth.”*

1 Peter 3:15 (ESV) — *“Always be prepared to give an answer to everyone who asks you to give the reason for the hope that you have. But do this with gentleness and respect.”*

Titus 3:10–11 (ESV) — *“Warn a divisive person once, and then warn them a second time. After that, have nothing to do with them.”*

Proverbs 4:23 (ESV) — *“Keep your heart with all vigilance, for from it flow the springs of life.”*

Colossians 3:23 (ESV) — *“Whatever you do, work heartily, as for the Lord and not for men.”*

Matthew 5:16 (ESV) — *“Let your light shine before others, so that they may see your good works and give glory to your Father who is in heaven.”*

Romans 12:17–18 (ESV) — *“Do not repay anyone evil for evil. Be careful to do what is right in the eyes of everyone. If it is possible, as far as it depends on you, live at peace with everyone.”*

This resource was born from real conversations in the Christian Twitch streaming community.

For additional ministry resources, visit forgebiblebot.com/community

“As iron sharpens iron, so one person sharpens another.”

— Proverbs 27:17 (ESV)