Pulse Sentinel

Advanced moderation and community protection for Telegram groups

Powered by Pulse Technologies

This manual explains how to use Pulse Sentinel inside Telegram groups, including setup, member tools, moderation commands, reporting, safety controls, and group operations.

- Setup-gated onboarding for new groups
- Member verification and reporting
- Moderator tools for warnings, mutes, bans, cases, and incidents
- Safety controls for raids, panic events, and spam bursts
- Operational visibility through logs, stats, and user investigation

User Edition

Prepared for Telegram group members, moderators, and group admins.

This manual is for people who use Pulse Sentinel inside Telegram groups.

It covers:

- private-chat onboarding
- group setup and activation
- member features
- moderator and group-admin commands
- safety, reporting, investigation, and analytics features

It does not cover:

- staff-only internal control-plane features
- deployment, hosting, or environment configuration

What Pulse Sentinel Does

Pulse Sentinel is a Telegram moderation and community protection bot for managed groups. It combines:

- moderation tools
- anti-spam and custom filters
- member verification
- reports, cases, and incidents
- safety controls such as raid mode and panic mode
- group analytics, logs, and moderation history

Before You Begin

Private Chat Onboarding

Open a private chat with the bot and send /start.

The private landing screen lets you:

- add the bot to a group
- read the setup guide
- review the public feature overview
- see where to use /help and /setup

Group Activation

Pulse Sentinel uses a setup-gated model.

- Any group can add the bot.
- The group becomes fully active only after a group admin completes /setup.
- Before setup is complete, advanced moderation and operations commands return a message telling you to run /setup first.

Safe entry points before setup:

- /help
- /setup
- /rules
- /id

Recommended Bot Permissions

For normal operation, promote the bot to admin and grant at least:

- Delete messages
- Restrict, ban or unban members
- Invite new users

Optional but useful:

- Pin messages

How Permissions Work

Group access is role-based.

- Members use member commands such as /help, /rules, /verify, and /report.
- Moderators use moderation and investigation commands.
- Group admins manage setup, settings, filters, safety controls, and log behavior.
- Some commands also respect internal bot roles assigned inside the group.

If a command is rejected, check:

- whether the group has completed /setup
- whether the bot has the right Telegram admin permissions
- whether your group role is high enough for that action

Common Usage Patterns

Reply-First Moderation

Most moderation commands are designed to be used as a reply to the target member's message. This is the safest workflow and reduces targeting mistakes.

Examples:

- reply with /warn spam
- reply with /mute 1h repeated links
- reply with /ban scam account
- reply with /purge

Duration Format

Commands that accept time windows use short duration formats such as:

- 30m
- 2h
- 7d
- 1w

Bulk Actions Need Confirmation

Bulk moderation commands require an explicit --confirm pass.

Examples:

- /massmute 1h 12345 23456 --reason raid cleanup --confirm
- /massban 12345 23456 --reason repeat scam --confirm

Public Output Is Compact

Pulse Sentinel keeps main group output compact by default.

- moderation confirmations are short
- operational commands return summaries instead of large dumps
- detailed audit information is sent to the configured log destination

Admins can tune this with /feedback.

Quick Start For Group Admins

1. Add Pulse Sentinel to the group.
2. Promote it to admin.
3. Run /setup.
4. Complete the guided configuration:
   - log destination
   - verification mode and timeout
   - anti-spam defaults
   - welcome message
   - rules text
   - link and media policy
   - raid defaults
5. Use /help in the group for the inline command center.

Member Features

/help

Open the guided inline help center.

Usage:

```
/help
```

/rules

Show the current group rules text.

Usage:

```
/rules
```

/verify

Use this to complete verification if the group requires it.

Behavior:

- members use it for their own verification
- moderators can approve a replied member when appropriate

Usage:

```
/verify
/verify <challenge_answer>
```

Moderator reply usage:

```
/verify
```

/report

Report a message to moderators by replying to it.

Usage:

```
/report
```

Notes:

- reports open a persistent review item
- moderators can act on reports through buttons and queue commands
- you cannot report yourself

Moderator Commands

These commands are typically for moderators, admins, or owners inside the group.

Warnings

/warn

Add a warning to the replied member.

Usage:

```
/warn <reason>
```

/warnings

List active warnings for the replied member.

Usage:

```
/warnings
```

/clearwarns

Clear active warnings for the replied member.

Usage:

```
/clearwarns <reason>
```

Mutes, Bans, and Removal

/mute

Mute the replied member, optionally for a duration.

Usage:

```
/mute
/mute 1h
/mute 1h <reason>
```

/unmute

Remove a mute from the replied member.

Usage:

```
/unmute
```

/ban

Ban the replied member, optionally temporarily.

Usage:

```
/ban
/ban 24h
/ban 24h <reason>
```

/unban

Unban a member by reply or by Telegram user ID.

Usage:

```
/unban <user_id>
```

Reply usage:

```
/unban
```

/kick

Remove the replied member from the group without a standing ban.

Usage:

```
/kick <reason>
```

Message Cleanup

/purge

Delete recent messages or purge from a replied anchor message.

Usage:

```
/purge <count>
/purge <all|links|media|stickers|gifs|forwarded> [count]
```

Reply usage:

```
/purge
/purge all
/purge links
/purge media
/purge stickers
/purge gifs
/purge forwarded
```

Notes:

- reply mode purges from the replied message to the command message
- media targets regular media posts and excludes stickers and GIFs
- filtered purge uses the stored recent message history

User Review and Investigation

/user

Show a compact moderation profile for a target user.

Usage:

```
/user <user_id>
```

Reply usage:

```
/user
```

What it shows:

- warnings and active moderation state
- risk, trust, and reputation summary
- incident and case summary

/investigate

Show a broader investigation view for a target user.

Usage:

```
/investigate <user_id>
```

Reply usage:

```
/investigate
```

What it includes:

- moderation history
- recent behavior signals
- cases and incidents
- moderator notes
- audit and timeline context

Moderator Notes

/note

Manage private moderator notes for a user.

Usage:

```
/note add <text>
/note list
/note delete <note_id>
```

Reply-first usage is recommended.

Cases

/case

Show a specific moderation case.

Usage:

```
/case <case_id>
```

/cases

List recent cases.

Usage:

```
/cases
/cases resolved
/cases dismissed
/cases all
```

Incidents

/incidents

List security incidents for the current group.

Usage:

```
/incidents
/incidents resolved
/incidents all
```

/incident

Inspect or manage a specific incident.

Usage:

```
/incident <incident_id>
/incident resolve <incident_id>
/incident assign <incident_id> <reply or user_id|unassign>
/incident stage <incident_id> <open|investigating|contained|resolved|postmortem>
/incident note <incident_id> <text>
/incident respond <incident_id> <raid|domain_lockdown|clear>
/incident propose <incident_id> <ban|panic|resolve> [reply or user_id] [reason]
/incident vote <proposal_id> <approve|reject>
```

What incidents track:

- suspicious domain activity
- correlated spam bursts
- response bundles and containment actions
- assignment, notes, and resolution timeline

Reputation

/reputation

Inspect or manage a user's reputation label.

Usage:

```
/reputation <reply or user_id>
/reputation set <reply or user_id> <trusted|neutral|watch|restricted> [reason]
/reputation clear <reply or user_id>
```

Notes:

- reputation is per group
- it complements the risk and trust profile
- it appears in /user and /investigate

Bulk Moderation

/massmute

Review and execute a bulk mute against explicit Telegram user IDs.

Usage:

```
/massmute 1h 12345 23456 --reason raid cleanup --confirm
```

/massban

Review and execute a bulk ban against explicit Telegram user IDs.

Usage:

```
/massban 12345 23456 --reason repeat scam --confirm
```

Notes:

- these commands do not use fuzzy matching
- explicit user IDs are required
- --confirm is required for execution

Group Admin Commands

These commands are usually for group admins or owners.

Setup and Core Configuration

/setup

Launch or re-run the guided setup wizard for the current group.

Usage:

```
/setup
/setup restart
```

/settings

Show current group settings and quick-toggle controls.

Usage:

```
/settings
```

/welcome

Enable, disable, or replace the welcome message.

Usage:

```
/welcome on
/welcome off
/welcome <custom text>
```

/setrules

Replace the rules text shown by /rules.

Usage:

```
/setrules <rules text>
```

/setlog

Set the moderation log destination. With no argument, the current chat is used.

Usage:

```
/setlog
/setlog <chat_id>
```

Custom Filters

/filter

Add or remove custom content filters.

Simple usage:

```
/filter add <term>
/filter remove <pattern>
```

Advanced usage:

```
/filter add <literal|wildcard|regex> <warn|delete|mute|ban|log_only> <pattern> [--priority N] [--context key=value]
```

Advanced notes:

- literal, wildcard, and regex matching are supported
- --priority controls evaluation order
- --context can target conditions such as:
  - modes=normal|raid|panic
  - message_kind=text|link|media|sticker|gif|forwarded
  - new_member=true|false
  - risk_level=<band>
  - trust_level=<band>
  - reputation=<band>

Examples:

```
/filter add literal warn scam phrase
/filter add wildcard delete free*gift --priority 50
/filter add regex mute (?i)crypto\\s+guarantee --context modes=raid,message_kind=text
```

/filters

List configured custom filters.

Usage:

```
/filters
```

Domain Rules

/domain

Manage explicit domain rules for the group.

Usage:

```
/domain <allow|block|suspicious|remove> <domain> [delete|warn|mute|ban|log_only] [reason]
```

Examples:

```
/domain block badsite.example delete phishing
/domain suspicious unusual.example warn review first
/domain allow trusted.example
/domain remove oldrule.example
```

/domains

List recent domain rules for the group.

Usage:

```
/domains
```

Reports Queue

/reports

List open reports for the group.

Usage:

```
/reports
```

Notes:

- report cards can include moderation action buttons
- reports feed into the case workflow

Internal Group Roles

/role

Assign, extend, clear, or list internal bot roles for trusted staff.

Usage:

```
/role set <reply or user_id> <owner|admin|moderator|helper> [duration] [--reason text]
/role extend <reply or user_id> <duration>
/role clear <reply or user_id>
/role list [active|all]
```

Notes:

- these roles are separate from Telegram’s built-in admin list
- temporary role grants can expire automatically

Feedback and Visibility

/feedback

Control how much the bot says in the main group.

Usage:

```
/feedback status
/feedback manual <full|compact|log_only>
/feedback automatic <full|compact|log_only>
/feedback cleanup <on|off> [delay_seconds]
```

What it controls:

- manual moderation confirmation detail
- automatic anti-spam visibility
- optional auto-delete of public moderation confirmations

Group Locks and Safety Modes

/lock

Restrict general posting in the group.

Usage:

```
/lock <reason>
```

/unlock

Reopen posting after a lock.

Usage:

```
/unlock <reason>
```

/raidmode

Manually enable or disable raid protections.

Usage:

```
/raidmode on
/raidmode off
```

/panic

Enable or disable emergency lockdown mode.

Usage:

```
/panic
/panic off
```

What these modes do:

- tighten verification and posting controls
- reduce spam tolerance
- restrict risky content types when needed
- help slow or contain raid-like activity

Utility Commands

/id

Show useful IDs for the current chat, user, or replied message.

Usage:

```
/id
```

/stats

Show a recent operational summary from persistent records.

Usage:

```
/stats
/stats 7d
/stats moderators 7d
```

What it can summarize:

- message activity
- moderation actions
- reports and incidents
- verification pass and failure counts
- moderator activity

/logs

Show recent moderation and audit history in compact form.

Usage:

```
/logs
```

How Reporting, Cases, and Incidents Fit Together

- /report creates a report for moderators.
- reports can create or update a moderation case
- /cases and /case help moderators review those items
- /incidents and /incident track broader security or coordinated abuse events

Use cases:

- use a report when a specific message or member needs moderator review
- use cases to follow moderation workflow over time
- use incidents to manage broader campaigns, suspicious domains, or coordinated abuse

Verification and Onboarding

Pulse Sentinel can keep new members restricted until they verify successfully.

Depending on group settings, verification may use:

- a button-based confirmation
- a challenge answer flow

Verification can be configured during /setup, including:

- verification mode
- timeout
- failure behavior

Risk, Trust, and Reputation

Pulse Sentinel keeps per-group user intelligence that moderators can review through /user, /investigate, and /reputation.

These profiles can reflect:

- moderation history

- anti-spam triggers
- verification outcomes
- suspicious domain activity
- upheld or dismissed reports

This helps moderators make more consistent decisions over time.

Public Output and Cleanup

In busy groups, Pulse Sentinel is designed to keep chat noise low.

- public moderation confirmations are compact by default
- richer detail is written to the configured log destination
- /feedback cleanup on [delay_seconds] can auto-delete public moderation confirmations after a short delay

This affects community-group output only.

Best Practices

- Complete /setup before relying on advanced commands.
- Use reply-based moderation whenever possible.
- Configure a dedicated log destination early with /setlog.
- Keep /feedback manual compact or /feedback manual log_only in busy groups.
- Use /reports, /cases, and /incidents together rather than relying on memory.
- Use /purge with scopes like links, media, or forwarded instead of broad deletion when you want precision.
- Use temporary internal roles when you need short-term delegated access.

Troubleshooting

“This group has not been initialized yet”

A group admin needs to run:

```
/setup
```

“Reply to the member's message and run the command again”

The command expects a reply target. Reply to the relevant message first, then run the command.

“This action is limited to admins or moderators”

Your group role is not high enough for that action, or the bot does not see the required Telegram admin permissions yet.

“The bot cannot complete this action”

Check that the bot has the required Telegram admin permissions, especially:

- delete messages
- restrict members
- invite users

Summary

Use Pulse Sentinel as a group-local moderation and protection tool:

- /help for the guided help center
- /setup to activate a new group
- /report to escalate member content
- moderation commands for day-to-day enforcement
- /cases, /incidents, /user, and /investigate for moderator workflow
- /stats and /logs for operational visibility