Built-in tools › Elevated Mode

Built-in tools

Elevated Mode

What it does

- `/elevated on` runs on the gateway host and keeps exec approvals (same as `/elevated ask`).
- `/elevated full` runs on the gateway host **and** auto-approves exec (skips exec approvals).
- `/elevated ask` runs on the gateway host but keeps exec approvals (same as `/elevated on`).
- `on` / `ask` do **not** force `exec.security=full`; configured security/ask policy still applies.
- Only changes behavior when the agent is **sandboxed** (otherwise exec already runs on the host).
- Directive forms: `/elevated on|off|ask|full`, `/elev on|off|ask|full`.
- Only `on|off|ask|full` are accepted; anything else returns a hint and does not change state.

What it controls (and what it doesn’t)

- **Availability gates**: `tools.elevated` is the global baseline. `agents.list[].tools.elevated` can further restrict elevated per agent (both must allow).
- **Per-session state**: `/elevated on|off|ask|full` sets the elevated level for the current session key.

- **Inline directive**: `/elevated on|ask|full` inside a message applies to that message only.
- **Groups**: In group chats, elevated directives are only honored when the agent is mentioned. Command-only messages that bypass mention requirements are treated as mentioned.
- **Host execution**: elevated forces `exec` onto the gateway host; `full` also sets `security=full` .
- **Approvals**: `full` skips exec approvals; `on` / `ask` honor them when allowlist/ask rules require.
- **Unsandboxed agents**: no-op for location; only affects gating, logging, and status.
- **Tool policy still applies**: if `exec` is denied by tool policy, elevated cannot be used.
- **Separate from `/exec`** : `/exec` adjusts per-session defaults for authorized senders and does not require elevated.

Resolution order

1. Inline directive on the message (applies only to that message).
2. Session override (set by sending a directive-only message).
3. Global default (`agents.defaults.elevatedDefault` in config).

Setting a session default

- Send a message that is **only** the directive (whitespace allowed), e.g. `/elevated full` .
- Confirmation reply is sent (`Elevated mode set to full...` / `Elevated mode disabled.`).
- If elevated access is disabled or the sender is not on the approved allowlist, the directive replies with an actionable error and does not change session state.

Send `/elevated` (or `/elevated:`) with no argument to see the current elevated level.

Availability + allowlists

Feature gate: `tools.elevated.enabled` (default can be off via config even if the code supports it).

Sender allowlist: `tools.elevated.allowFrom` with per-provider allowlists (e.g. `discord`, `whatsapp`).

Per-agent gate: `agents.list[].tools.elevated.enabled` (optional; can only further restrict).

Per-agent allowlist: `agents.list[].tools.elevated.allowFrom` (optional; when set, the sender must match **both** global + per-agent allowlists).

Discord fallback: if `tools.elevated.allowFrom.discord` is omitted, the `channels.discord.allowFrom` list is used as a fallback (legacy: `channels.discord.dm.allowFrom`). Set `tools.elevated.allowFrom.discord` (even `[]`) to override. Per-agent allowlists do **not** use the fallback.

All gates must pass; otherwise elevated is treated as unavailable.

Logging + status

Elevated exec calls are logged at info level.

Session status includes elevated mode (e.g. `elevated=ask`, `elevated=full`).

‹ apply_patch Tool Thinking Levels ›