Set up Simple Voice Chat with an allocated UDP port

Simple Voice Chat adds proximity audio to a compatible Minecraft Java server. The Bukkit plugin runs on supported Paper-style servers, while each player who wants voice needs the matching client mod. Installing the plugin alone does not put voice controls into an unmodified Minecraft client.

Match the platform before installing

Record the Minecraft version, server platform, Java runtime and plugin release. Choose the server download for that platform; a Fabric or Forge server needs its corresponding mod rather than the Bukkit JAR. Check the project's current compatibility notes instead of assuming that a recent game update is supported.

Use the plugin installation guide for a Paper installation. Keep a completed backup and the previous voice configuration before replacing an existing build. Ask two players to install compatible client builds on their own supported loaders, then join with the ordinary game address.

Assign a usable voice endpoint

1. Find a UDP allocation explicitly assigned to this service. Ask support if you cannot identify one. A number that works for Minecraft TCP is not automatically available for another protocol.
2. Start once to generate the plugin configuration, then stop normally before editing. Open the generated voice-chat server properties file; the exact location depends on the installed platform.
3. Set its `port` to the confirmed UDP allocation. The documented default is `24454`, but it is only usable when assigned and reachable for your service.
4. Leave binding and advertised-address settings at their generated values unless the service's network arrangement requires a documented change. A public address is not necessarily an address that exists inside the server container.

5. Start and inspect Console for a successful listener and any bind error. Keep other services off that same UDP allocation.

The plugin's voice traffic and Minecraft login traffic are separate. Reusing the same numeric TCP and UDP port can work when both allocations are available, but matching numbers alone does not establish that. Do not open every port or replace the game port to test voice.

Test audio in both directions

Open the client voice menu, complete microphone setup, select the intended input/output devices and check mute or push-to-talk state. Have two players stand nearby and speak in turns. A remote third player provides a useful check that the test is not only working on one local network.

Use the installed version's documented voice connection test when diagnosing a failure. Confirm operating-system microphone access on the client. A working text chat does not prove the UDP listener is reachable, and a normal website-based TCP port checker cannot validate this audio path.

Separate network failures from client problems

If everyone sees a disconnected voice icon, check the assigned endpoint, listener startup and UDP reachability with support. If only one player fails, compare client versions, microphone permissions and local security software. Proxy networks need the project's separate proxy instructions; do not copy a standalone endpoint into every backend.

Send support the server ID, voice version, exact error and test time. Redact tokens and private addresses from public logs. See collecting useful Minecraft logs. Further reading: server setup, server configuration and troubleshooting.

Looking for another plugin? Browse the Minecraft plugin guide directory.

Revision #2
Created 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin
Updated 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin