# Set up a Velocity network with protected backend servers

Velocity runs as a separate proxy in front of your Minecraft servers. Players join the proxy; it forwards them to a lobby or game server. This guide covers a modern Velocity and Paper network. Legacy clients and other server implementations need their own compatibility checks.

# Plan services and protect the backend path first

Allocate resources for the proxy and each backend. Server splits share your parent service's allowance; creating a split does not add RAM or storage. Record each server ID and its assigned address and port.

**Ask CloudExa support to confirm backend access restrictions before exposing an offline-mode backend.** Only your authorised proxy should reach backend game ports. Modern forwarding adds identity protection, but it does not replace network isolation. On shared hosting, a source-IP rule may not distinguish other containers using the same outbound IP. Support must check the actual path.

Do not copy localhost recipes between separate game containers. Each container has its own network context; `127.0.0.1` inside the proxy does not identify another split.

# Configure the proxy

1. Back up existing servers and obtain a Velocity service or split configured with the correct runtime. Ask support if the required server type is unavailable.
2. Start the proxy once to generate configuration, then stop it before editing `velocity.toml`.

3. Set its listening port to the allocation assigned to the proxy. Use the container-compatible bind address supported by your service; do not bind it to a public host IP that is absent inside the container.
4. Keep proxy `online-mode = true` for an authenticated Java network and set `player-info-forwarding-mode = "modern"`.
5. In the existing `[servers]` section, add each backend's reachable address and port. Set the `try` list to your lobby name. Use the addresses confirmed for proxy-to-backend traffic, not an example from this guide.
6. Keep `forwarding.secret` private. You will copy its value into the backend configuration; never share it in a ticket or screenshot.

# Configure each supported Paper backend

After support confirms the network protection, stop the backend and back up its configuration. For current Paper, enable `proxies.velocity.enabled` in `config/paper-global.yml`, set its secret to the proxy's forwarding secret, and set `proxies.velocity.online-mode` to match the proxy. Disable legacy BungeeCord forwarding in `spigot.yml` if previously enabled.

Set `online-mode=false` in the backend's `server.properties` only as part of this protected proxy setup. The proxy authenticates players. Do not apply that change to a standalone public Java server. Older Paper releases use different configuration paths; follow the matching upstream version instructions instead of adding duplicate settings.

# Start and test the network

1. Start the backends, then the proxy. Check every console for binding and forwarding-secret errors.
2. Join through the proxy's public address with an authenticated test account. Confirm the correct lobby and player identity, inventory and permissions.
3. Test switching to another backend and returning to the lobby.
4. Ask support to verify that an unauthorised connection cannot bypass the proxy and reach a backend. Do not open backend access as a shortcut for a failed proxy test.

If connection fails, provide the proxy/backend IDs, target port, timestamp and redacted error. For secret mismatches, compare values privately. For timeouts, ask support to check the permitted network path. A public server list or hidden backend address alone does not protect a backend.

References: Velocity configuration, modern forwarding and backend security.

Revision #2
Created 2026-11-08 20:06:27 UTC by CloudExa Hosting Admin
Updated 2026-11-08 20:06:28 UTC by CloudExa Hosting Admin