# Connect MCP Client to Streamable HTTP MCP Server

## Overview

In this lesson, we tackled the process of linking an MCP client to an MCP server using **HTTP streamable transport**. Here’s the catch: many MCP clients aren't equipped for this. Our client, **Claude**, is one such example; it only connects to local servers via `STDIO`. But fear not! We found a solution.

## The Solution: `mcp-remote`

- A nifty utility exists, called `mcp-remote`, available via **NPM**.
- It acts as a bridge to connect local-only MCP clients, like **Claude**, to remote MCP servers.

## How It Works

- **Streams Calls:** It streams tool and resource calls from the server to the client.
- **Simple Setup:** Use the command `npx mcp-remote`, followed by your server URL.

## Step-by-Step Process

1. **Setup `mcp-remote`:**
    - Utilize `npx` with `mcp-remote` and the server URL.
    - **Example:** `npx mcp-remote [Your Server URL]`.
2. **Configure `Claude`:**
    - Update **Claude’s desktop config JSON** with the server URL.
    - Add `/MCP` to the server URL for **streaming purposes**.
3. **Handling HTTP and SSL:**
    - Use **HTTPS** URLs ideally.
    - If using HTTP, don’t forget the extra argument `allow HTTP`.
4. **Testing the Connection:**
    - Restart **Claude** to apply changes.
    - Verify connectivity by performing a simple task like sending a greeting.

## Results and Benefits

- **Seamless Connection:** Our client can now connect to remote servers without hassle.
- **Secure Communication:** Supports HTTPS for enhanced security.

## Final Thoughts

In the next segment, we’ll deploy the server online on a **virtual machine**. This will enable global accessibility without local installs. The future is bright with

**streamable HTTP MCP servers** becoming the norm, simplifying how clients and servers interact at a distance. Stay tuned!