

# Connect to MCP Server Streamable HTTP with MCP Inspector

## Overview

In this lesson, we wrapped up our exploration on how to connect a **locally installed MCP client** to a **remote MCP server** using **streamable HTTP transport**. We ensured that the connection was smooth and verified by using the **MCP Inspector**.

## Setting Up the MCP Server

- **Ensure MCP Server is Running:**
  - We verified that our MCP server is up and running. This involved checking the virtual machine to confirm the server's core is active on a specific host and port.

## Using the MCP Inspector

- **Open MCP Inspector:**
  - We opened the **MCP Inspector** by using the command `MCP dev` to get a host and port to run in our local browser.
- **Verify Server Configuration:**
  - Confirmed the transport type as **streamable HTTP**.
  - Updated the server's URI to the one used by the virtual machine.
  - Adjusted the port number to match what was shown in the terminal.
  - Added the suffix `/MCP` to the URI for accurate connections.

## Connecting to the Server

- **Making the Connection:**
  - We clicked 'connect' in the inspector and successfully connected to the remote MCP server.

## Interacting with the Server

- **Running Tools:**
  - We explored some interactive functionalities:
    - Listed available tools.
    - Ran a tool to send a greeting as "Henry" and received the response "hi Henry."

## Benefits of Remote MCP Servers

- **Distributed Load:**
  - The major advantage of setting up an MCP server on a remote virtual machine is offloading processing from the local machine, which is beneficial when the server is accessed by multiple MCP clients.

## Final Steps

- **Ensure Client Connection:**

- Besides verifying the inspector connection, we also aimed to ensure actual clients could connect to the MCP server seamlessly.

This lesson highlighted the practical steps and advantages of connecting and interacting between local clients and remote MCP servers, ensuring efficient and distributed processing.