Official Model Context Protocol (MCP) Registry: Developer Guide

The Model Context Protocol (MCP) Registry serves as the central, community-driven catalog for discovering and managing MCP servers, acting as an "app store" for AI tools [1]. This guide provides the necessary instructions and links for both connecting to (consuming) the registry data and publishing your own MCP server to the official registry.

I. Connecting to the Official MCP Registry (Client Consumption)

For developers building AI hosts, IDE extensions (like GitHub Copilot or VS Code), or custom agents that need to discover available MCP servers, the official registry provides a stable, versioned API.

1. Registry API Base URL

The official MCP Registry is hosted at the following base URL:

> `https://registry.modelcontextprotocol.io` [2]

2. Key API Endpoints (v0.1 Specification)

The Registry API is currently in an **API freeze (v0.1)**, ensuring stability for integrators [1]. All client applications should target the `/v0.1/` path for server discovery.

Endpoint	Method	Description	Purpose
`/v0.1/servers`	`GET`	Returns a list of all registered MCP servers. Supports query parameters for filtering and searching.	**Server Discovery**
`/v0.1/servers/{id}`	`GET`	Retrieves the full metadata for a specific MCP server, identified by its unique ID (e.g., `io.github.user/server-name`).	**Metadata Retrieval**

`/v0.1/publish`	`POST`	Endpoint for registering a new MCP server. This is primarily used by the `mcp-publisher` CLI and requires authentication.	**Server Registration**

Example Client Connection (Discovery):

To retrieve a list of all registered servers, a client would make a simple `GET` request:

Shell

```
curl "https://registry.modelcontextprotocol.io/v0.1/servers"
```

Note on CORS: For browser-based tools, ensure your API gateway or client is configured to handle **CORS (Cross-Origin Resource Sharing)** correctly to allow fetching registry metadata from different domains [3].

II. Publishing to the Official MCP Registry (Server Development)

To list your own MCP server in the official registry, you must use the dedicated command-line interface (CLI) tool, `mcp-publisher`, and adhere to the official `server.json` schema.

1. Prerequisites

- **An MCP Server**: Your server must be ready to be packaged and published.
- **GitHub Account**: Required for authentication and namespace verification.
- **`mcp-publisher` CLI**: The official tool for interacting with the registry.

2. Installing the `mcp-publisher` CLI

The recommended way to install the CLI is via the official installation script or Homebrew [4].

Linux/macOS (using `curl`):

Shell

```
curl -L
"https://github.com/modelcontextprotocol/registry/releases/latest/download/mc
```

```
p-publisher_$(uname -s | tr '[:upper:]' '[:lower:]' )_$(uname -m | sed
's/x86_64/amd64/;s/aarch64/arm64/').tar.gz" | tar xz mcp-publisher && sudo
mv mcp-publisher /usr/local/bin/
```

macOS (using Homebrew):

Shell

```
brew install mcp-publisher
```

Verify the installation:

Shell

```
mcp-publisher --help
```

3. Creating the `server.json` Metadata File

The `server.json` file contains all the necessary metadata for your MCP server. The `mcp-publisher` tool can generate a template for you.

1. **Generate Template:** Run the following command in your server project directory:

   Shell

   ```
   mcp-publisher init
   ```

2. **Schema Reference:** The generated file will reference the official schema, which is essential for validation:

   > `https://static.modelcontextprotocol.io/schemas/2025-12-11/server.schema.json` [4]

3. **Edit Metadata:** Manually edit the generated `server.json` file, ensuring the `name` property follows the required namespace format (e.g., `io.github.your-username/your-server-name`) and matches the `mcpName` in your `package.json` (if applicable) [4].

4. Authenticating and Publishing

The registry uses **GitHub OAuth** for authentication to verify your identity and namespace ownership [4].

1. **Login:** Authenticate with your GitHub account:

   Shell

```
mcp-publisher login github
```

Follow the on-screen instructions to visit the provided URL and enter the authorization code.

2. **Publish:** Once authenticated, publish your server metadata:

Shell

```
mcp-publisher publish
```

A successful publication will return a confirmation message, and your server will be available for discovery via the `/v0.1/servers` endpoint.

References

[1] modelcontextprotocol/registry. (n.d.). MCP Registry. GitHub. Retrieved December 19, 2025, from

[2] Official MCP Registry. (n.d.). Official MCP Registry. Retrieved December 19, 2025, from

[3] GitHub Docs. (n.d.). Configure an MCP registry for your organization or enterprise. Retrieved December 19, 2025, from

[4] modelcontextprotocol/registry. (n.d.). Quickstart: Publish an MCP Server to the MCP Registry. GitHub. Retrieved December 19, 2025, from