

Windows系统下本地MQTT服务器搭建

1. 下载并安装emqx服务器

1. 访问Eqmx官网

The screenshot shows the official EMQX website. At the top, there's a navigation bar with links for '产品', '案例', '文档', 'MQTT', '博客', and '社区'. On the right side of the header, there are social media icons for GitHub (12.4k), a '联系我们' (Contact Us) button, and a green '立即开始' (Start Now) button. Below the header, the main title '大规模分布式 MQTT 消息服务器' is displayed in large bold letters. A subtitle below it reads '高效可靠连接海量物联网设备，实时处理分发消息与事件流数据，助力构建关键业务的物联网与云应用。'. Underneath the subtitle are two prominent buttons: a red-bordered '立即下载 →' (Download Now) button and a green '试用 EMQX Cloud →' (Try EMQX Cloud) button. A red arrow points to the '立即下载' button. Below these buttons, there's a section titled '全球近千家企业用户信赖' (Trusted by nearly 1,000 enterprises worldwide) featuring logos from various companies like Intel, Foxconn, and Verifone. To the right, there's a watermark for 'CSDN @豪哥追求卓越'.

2. 选中合适的MQTT服务器版本

由于我们使用的是本地部署MQTT服务器，而且只使用基础功能的MQTT服务器功能，所以选中“大规模分布式MQTT消息服务器”即可，如下如图所示，点击下载按钮，下载。

The screenshot shows the '选择适合您的版本，开始免费试用' (Choose your version, start free trial) section of the website. It compares three options:

- EMQX**: 大规模分布式 MQTT 消息服务器. Features include: 基于 APN 2.0 开放源码协议, 完整 MQTT 3.x 和 5.0 规范, Masterless 高可用集群架构, 高并发、低时延、高性能, 可扩展的网关和插件体系. Download button: 下载 →
- EMQX Enterprise**: 企业级 MQTT 物联网接入平台. Features include: 标准或专有多协议支持, 基于 SQL 的 IoT 数据集成, 数据持久化与数据桥接, 管理与监控中心, 7x24 小时技术支持服务. Free Trial button: 免费试用 →
- EMQX Cloud**: 全托管的 MQTT 消息云服务. Features include: 即用即付, 14 天免费试用, 按需扩容缩容, 无需担忧服务器性能, 40+ 种完备的数据集成链路, 支持 7 大主流云服务提供商多达 30+ 区域, 根据您的需求定制 EMQX 的规格和配置. Free Trial button: 免费试用 →

A red arrow points to the 'EMQX Enterprise' section. At the bottom right, there's a watermark for 'CSDN @豪哥追求卓越'.

3. 选中“Windows”，下载“emqx-5.3.0-windows-amd64.zip”，如下图所示。

下载 EMQX

Docker Ubuntu Debian CentOS/RHEL Amazon Linux macOS 1 Windows Kubernetes Terraform

版本
v5.3.0 更新日志 历史版本

1 下载 emqx-5.3.0-windows-amd64.zip，解压
2 命令行下进入解压路径，启动 EMQX
./emqx/bin/emqx start

使用指南 配置估算 NEW

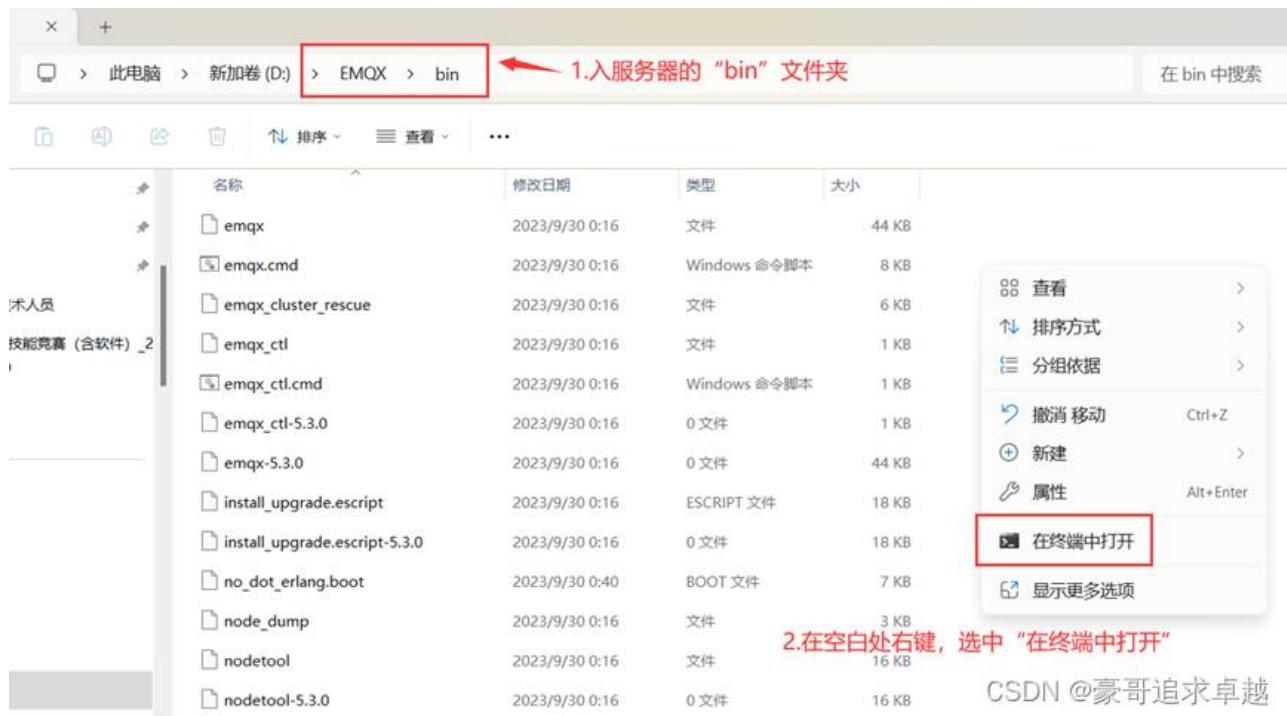
CSDN @豪哥追求卓越

4. 在电脑任意一个盘，新建文件夹，例如“EMQX”文件，然后将下载的压缩文件解压，如下图所示。



2. 运行emqx服务器（即MQTT服务器），并登录服务器后管理页面。

1. 启动EMQX（即MQTT服务器）



然后，在命令行输入“./emqx start”，即可启动MQTT服务器，如下图所示。

```

Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

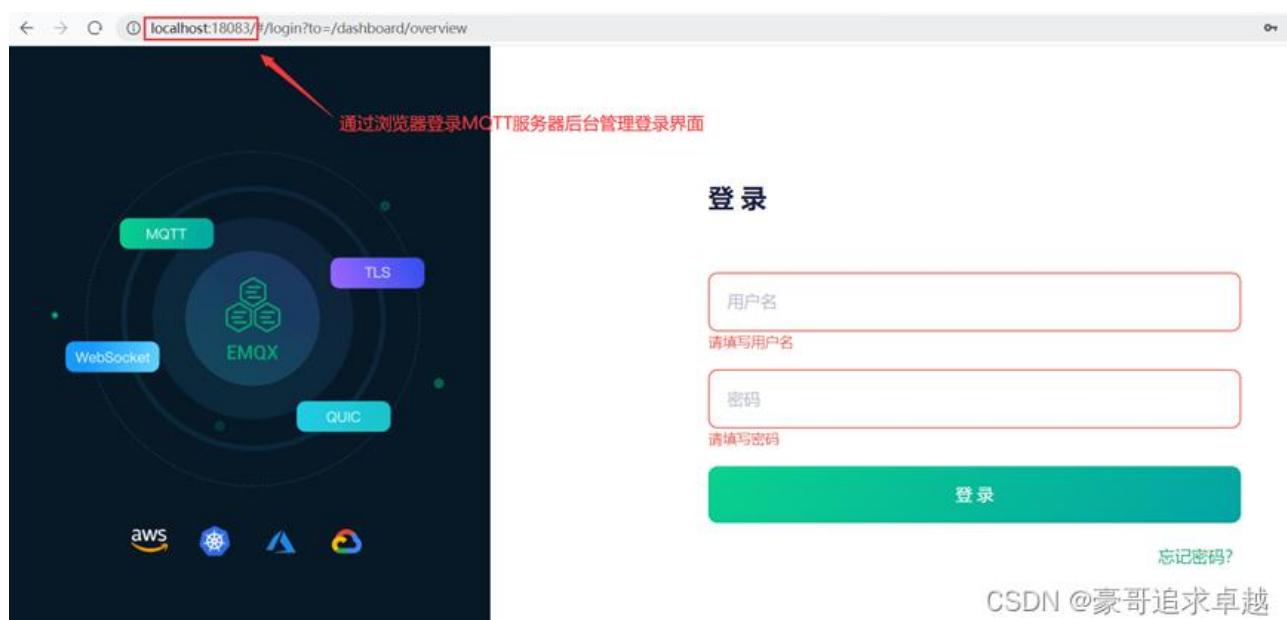
安装最新的 PowerShell，了解新功能和改进！https://aka.ms/PSWindows
PS D:\EMQX\bin> ./emqx start
MQTT_NODE_DB_ROLE [node.role]: core
MQTT_NODE_DB_BACKEND [node.db_backend]: mnesia
PS D:\EMQX\bin> |

```

CSDN @豪哥追求卓越

2. 登录MQTT服务器后台管理界面

①通过浏览器访问<http://localhost:18083/>（localhost 可替换为您的实际 IP 地址），如下图所示



②输入默认账号和默认密码，登录MQTT服务器后台管理界面

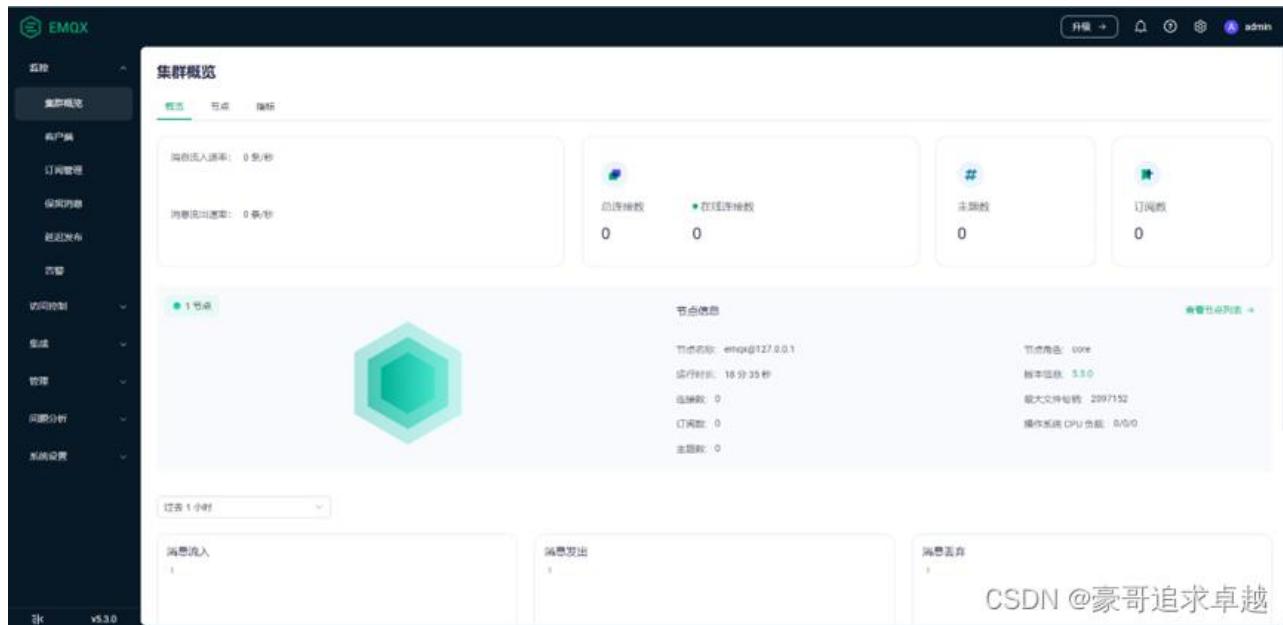
输入默认账号为“admin”，默认密码为“public”，然后点击登录按钮，进入MQTT服务器后台管理界面。



如果第一次使用，会提示“为了保护账户安全，请修改默认密码”，如下图所示，你可以选择“跳过”，或者修改密码。



登录成功后，MQTT服务器后台管理界面，如下图所示。



The screenshot shows the EMQX Cluster Overview dashboard. On the left, a sidebar lists various monitoring and configuration tabs: 集群概况 (Cluster Overview), 客户端 (Clients), 订阅管理 (Subscription Management), 保留消息 (Retained Messages), 超时发布 (Timeout Publishing), 防篡改 (Anti-tampering), 访问控制 (Access Control), 集成 (Integration), 策略 (Policy), 问题分析 (Problem Analysis), 和系统设置 (System Settings). The main area displays cluster statistics: 消息流入速率 (Message Inflow Rate) at 0 每秒, 消息流出速率 (Message Outflow Rate) at 0 每秒, 总连接数 (Total Connections) at 0, 在线连接数 (Online Connections) at 0, 主题数 (Topic Count) at 0, and 订阅数 (Subscription Count) at 0. A central node summary shows 1 个节点 (1 node) with details: 节点名称 (Node Name) emqx@127.0.0.1, 运行时间 (Run Time) 16 分 35 秒, 话题数 (Topic Count) 0, 消息数 (Message Count) 0, 主题数 (Topic Count) 0, 节点角色 (Node Role) core, 版本 (Version) 3.3.0, 最大文件句柄 (Max File Handles) 2097152, 操作系统 (Operating System) CPU 使用率 (CPU Usage) 0/0. The bottom right corner features a watermark: CSDN @豪哥追求卓越.

3. 增加MQTT1和MQTT2用户。

1. 创建认证

点击按钮创建认证，如下图所示。



The screenshot shows the '客户端认证' (Client Authentication) creation page. The left sidebar has a '客户端认证' tab selected (marked with a red box 1). The main area has a '+ 创建' (Create) button highlighted with a red box 3. The table header includes '数据源及认证方式' (Data Source and Authentication Method), '数据源状态' (Data Source Status), '是否启用' (Enabled), and '操作' (Operations). Below the table, there is a note: '暂无数据' (No data available). The bottom right corner has a watermark: CSDN @豪哥追求卓越.

认证方式选中“Password-Based”，即使用客户端用户名、Client ID与密码进行认证，如下图所示。



The screenshot shows the first step of the client authentication configuration: '认证方式' (Authentication Method). It features three tabs: 1. Password-Based (highlighted with a red box), 2. JWT, and 3. SCRAM. Below the tabs, a note says: '使用客户端用户名、Client ID 与密码进行认证' (Authenticate using client username, Client ID, and password). At the bottom, there are '取消' (Cancel) and '下一步' (Next Step) buttons, with '下一步' highlighted with a red box 2. The bottom right corner has a watermark: CSDN @豪哥追求卓越.

“数据源”选中“内置数据库”，如下图所示。

客户端认证

< 返回 | 创建认证

认证方式 (Password-Based) | 数据源 | 配置参数

选择存储认证数据的数据库或提供认证数据功能的 HTTP 服务器

内置数据库 1 MySQL MongoDB PostgreSQL
Redis HTTP 服务

上一步 下一步 2 CSDN @豪哥追求卓越

配置参数，账号类型选中“username”，密码加密方式选中“sha256”，加盐方式选中“suffix”，如下图所示。

客户端认证

< 返回 | 创建认证

认证方式 (Password-Based) | 数据源 (内置数据库) | 配置参数

账号类型 * 1
username 1
密码加密方式 2
sha256
加盐方式 3
suffix 3
上一步 创建 4 CSDN @豪哥追求卓越

2. 创建用户

如下图所示，创建认证后，点击“用户管理”，进入用户管理界面。

客户端认证

数据源及认证方式	数据源状态	是否启用	操作
内置数据库 Password-Based	已连接	<input checked="" type="checkbox"/>	用户管理 设置 更多

用户管理 CSDN @豪哥追求卓越

进入用户管理界面之后，点击“+”按钮，分别添加“MQTT1”和“MQTT2”用户，如下图所示。

客户端认证

< 返回 | 内置数据库

已连接 password_based

概览 设置 用户管理

用户名 是否为超级用户 操作

暂无数据 CSDN @豪哥追求卓越

用户名分为“MQTT1”和“MQTT2”，密码都是“123456”，都是“超级管理员”，如下所示。

1 Mqtt1 2.这里密码为123456 3 是否为超级用户 4 保存

1 Mqtt2 2.密码为123456 3 是否为超级用户 4 保存

创建完MQTT1和MQTT2用户后，用户界面如图下所示。

客户端认证

< 返回 | 内置数据库

已连接 password_based

概览 设置 用户管理

用户名 是否为超级用户 操作

MQTT1	是	编辑 删除
MQTT2	是	编辑 删除

CSDN @豪哥追求卓越

4. 测试本地部署的MQTT服务器

1. 下载EMQX的（MQTT客户端），如下图所示。



Why MQTTX 准备 MQTT 服务器 连接测试 EMQX 立即开始 3.1k

NEW Newsletter 202304 | v1.9.2 发布，CLI 支持输出模式 →

MQTT 5.0 客户端工具

MQTTX 是一个强大的跨平台 MQTT 5.0 桌面和 CLI 客户端，使学习、探索和开发 MQTT 变得快速而简单。

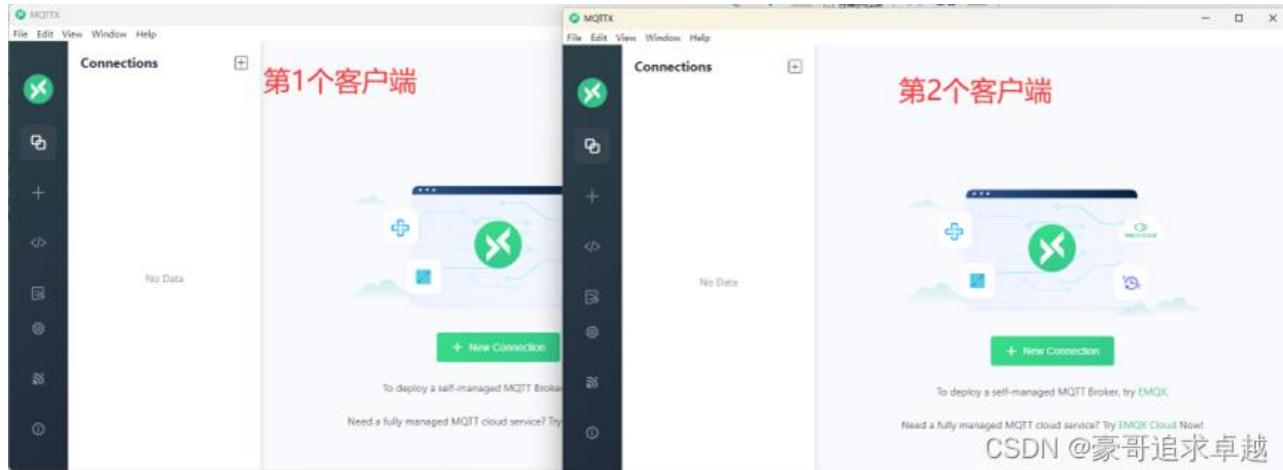
立即下载 →

GitHub →



CSDN @豪哥追求卓越

2. 分别打开两个MQTT5.0客户端工具。



3. 将两个MQTT5.0客户端工具的语言设置为“简体中文”，步骤下图所示。



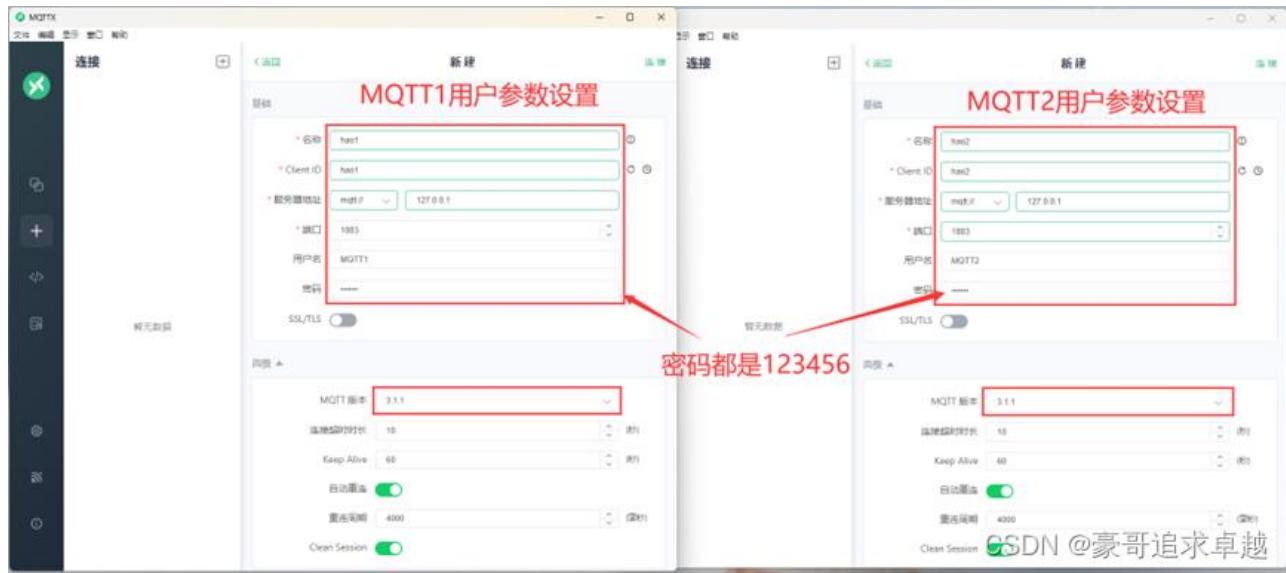
4. 新建MQTT1和MQTT2的用户连接

分别点击客户端1和客户端2的新建连接按钮，如下图所示。

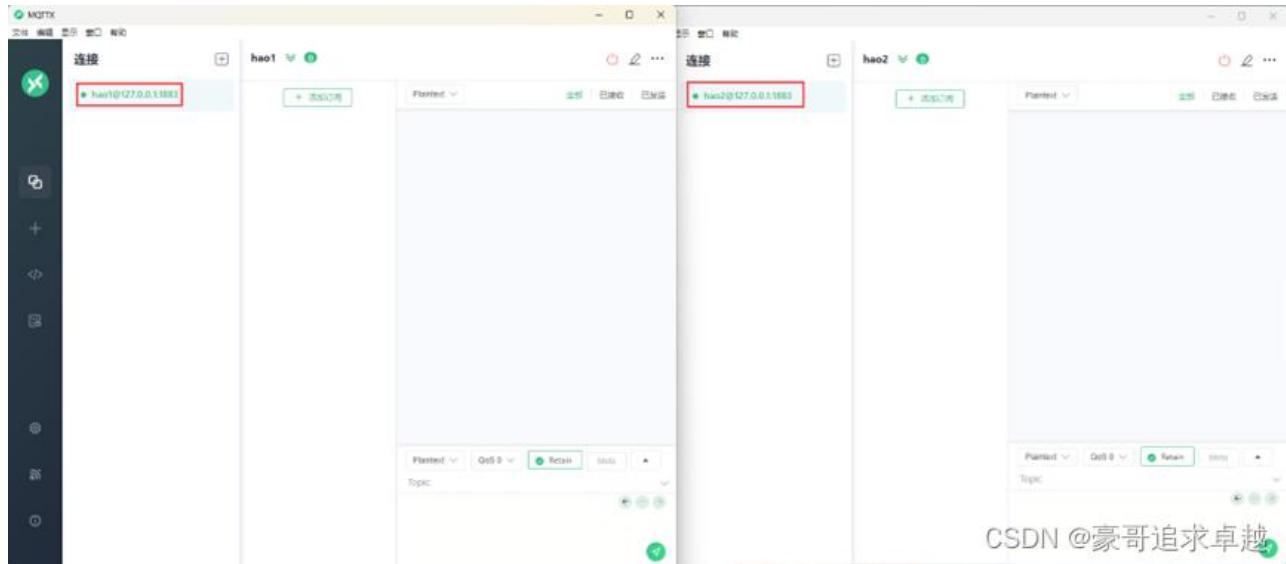


1. 配置MQTT1和MQTT2的用户连接参数

名称和Client ID可以任意填写，这里客户端1填写hao1，客户端2填写hao2，服务器地址都选中“mqtt://”并填写“127.0.0.1”，端口都是“1883”，用户名分别填写“MQTT1”和“MQTT2”，密码都是“123456”，MQTT版本都选中“3.1.1”，如下图所示



6. 点击两个客户端的连接按钮，连接MQTT服务器，如下图所示



7. 设置两个客户端的发布主题。

客户端1的发布主题为“/MQTT1/pub”，发布的数据为：

```

1  {
2      "wendu": 23.56
3  }
4

```

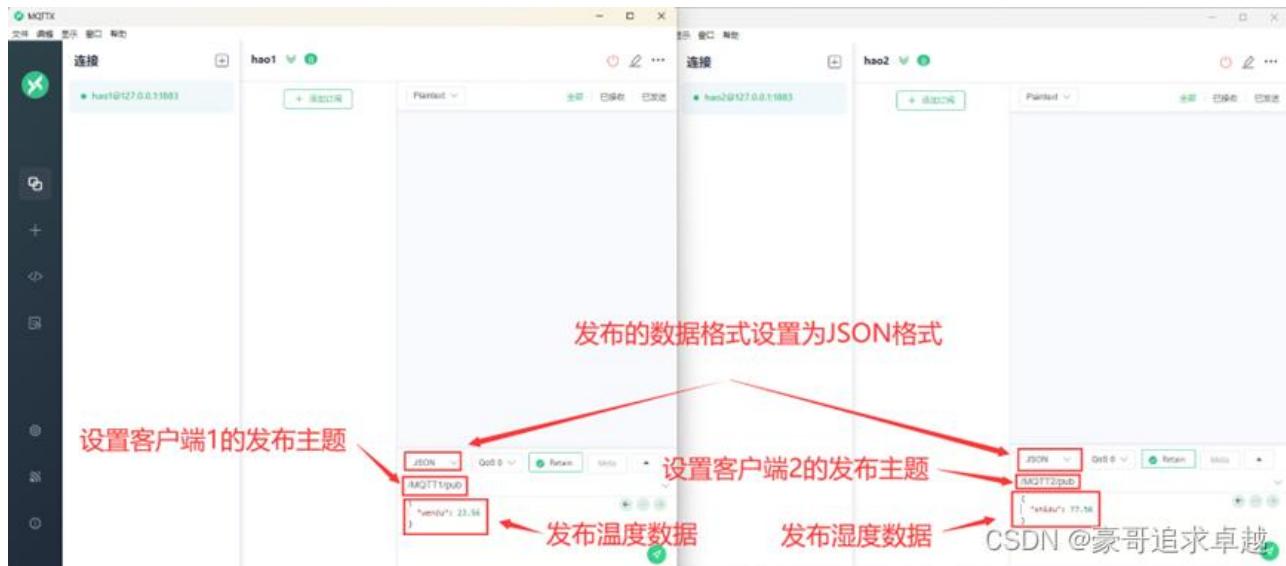
客户端2的发布主题为“/MQTT2/pub”，发布的数据为：

```

1  {
2      "shidu": 77.56
3  }
4

```

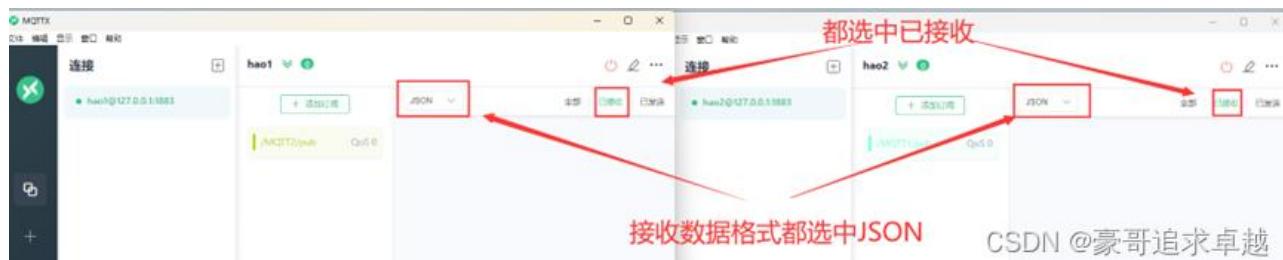
如下图所示。



8. 设置两个客户端的订阅主题。

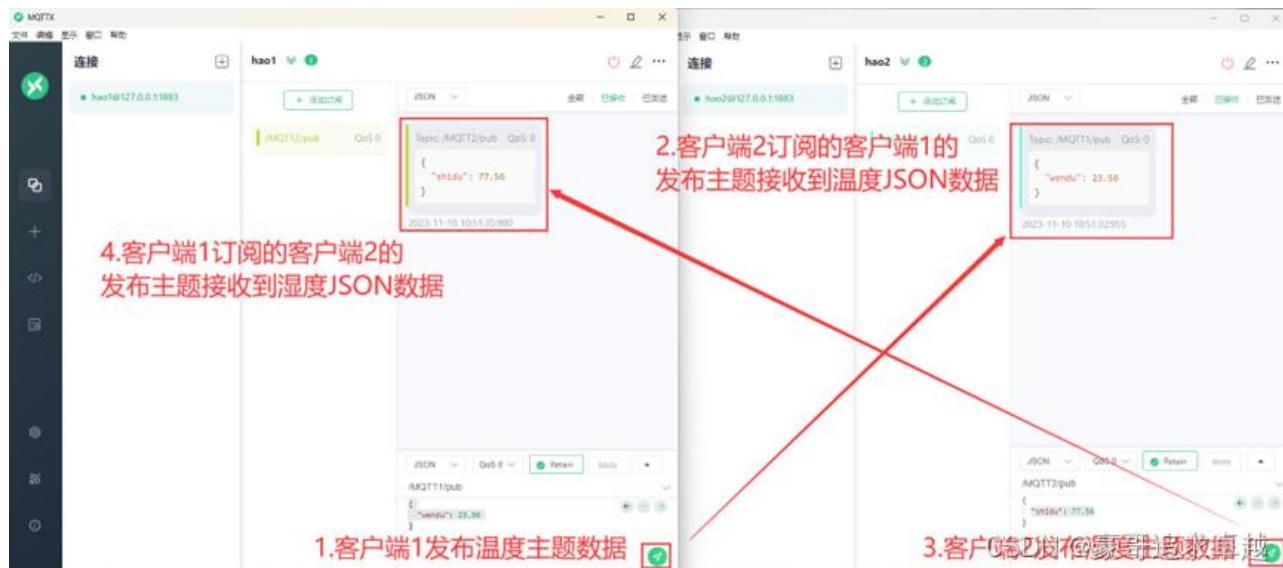


设置订阅接收数据格式为JSON并且选中只显示已接收数据，如下图所示。



9. 分别点击两个客户端发布按钮。

如下图所示，当我们点击客户端1的发布主题按钮时，在客户端2上的订阅主题接收到客户端1发布的主题的温度JSON数据，当我们点击客户端2的发布主题按钮时，在客户端1上的订阅主题接收到客户端2发布的主题的湿度JSON数据。



本教程讲述了如何在Windows电脑本地上[搭建MQTT服务器](#)，怎样进入MQTT服务器后台管理界面，如何在MQTT服务器上添加用户，如何通过MQTT客户端连接MQTT服务器，并通过两个MQTT客户端演示了MQTT通信的简单发布主题和订阅主题。