

AgentSQL

智盾

快速上手

v4.13.0

agentsql.cn

v4.13.0 使用文档

目录

你将得到什么

明确不做什么

前置条件与兼容矩阵

路径 A：5 分钟零配置看效果（一条命令）

路径 B：接入你自己的库（真实生产闭环）

1. 安装并启动
2. 登录控制台
3. 添加数据源
4. 创建 Agent
5. 建立最小 allow 权限策略
6. 接入 MCP 客户端

任务级临时只读 Key：申请、异人审批与结果预算

5 分钟验收清单

v4.4.0：升级前只读检查

下一步

AgentSQL 快速上手

语言 / Language: 简体中文 | English

> 现行指南：适用于 v4.13.0 及兼容的 4.13.x 维护版本。历史引入版本与验收记录保留原始事实；日常入口见文档索引。

> 本版为独立问数 Agent 与独立 MCP Server 二合一产品；按三种使用方式选择部署。问数图表操作见该指南；完整支持边界见v4.13.0验收记录。以下标明历史版本的章节保留原始条件。

> 本文说明 AgentSQL v4.13.0 的安装与使用。执行下列固定 v4.13.0 tag 的安装、Demo 和镜像命令前，先在 GitHub Releases 核对正式资产及签名，并确认对应 GHCR 镜像可以匿名拉取；源码仓库为 `github.com/cuipengdba/agentsql`。

AgentSQL 是放在 AI Agent 与业务数据库之间的数据库安全网关和生产级 MCP Server，让每条 SQL 在执行前经过身份、权限、规则和决策检查，并在可审计的边界内执行。

```
人 → AI Agent / LLM → MCP → AgentSQL → PostgreSQL / MySQL 业务库
```

AgentSQL 同时提供数据库安全网关与独立的问数界面。问数服务可将自然语言转换为待执行 SQL；SQL 必须再经过网关的身份、权限、规则和审计检查。它不替代数据库账号与权限体系，数据库侧的最小权限运行账号始终是最后一道防线。

你将得到什么

- 为每个 Agent 分配独立 API Key 和 `readonly`、`dml`、`ddl` 能力档位。
- 按 Agent × 数据源 × 对象建立默认拒绝的授权策略。
- 对单条 SQL 做解析、规则判断、受控执行、结果限行和基础脱敏。
- 在控制台查看 `allow`、`deny`、`warn`、`approve`、`error` 的证据链与应用层审计。
- 通过 Streamable HTTP 或 stdio 向 MCP 客户端提供 9 个固定基础工具；默认开启的 PostgreSQL B5 会话/计划事务另提供 8 个工具。

明确不做什么

- 不替代数据库账号、网络隔离、TLS 终止、备份、监控与变更管理。

- 不保护绕过 AgentSQL 直连数据库的请求。
- PostgreSQL 的 B2 列级授权、B5 跨请求逻辑会话与计划事务均出厂默认开启；MySQL 不进入 B2 PostgreSQL 路径，且不支持 B5 跨请求事务。每个 operation 仍只允许一条顶层 SQL，并受预检计划和会话安全边界约束。
- 审批通过不会自动执行 SQL；审计不是法规级 WORM。
- 当前不支持完整 DLP、企业身份、高可用或 Kubernetes 编排。

前置条件与兼容矩阵

范围	当前支持	说明
一键安装宿主	`linux/amd64`、`linux/arm64`；glibc 2.28+、systemd	v0.4.0 起提供 linux/arm64 原生 glibc 包；`x86_64` / `amd64` 对应 amd64，`aarch64` / `arm64` 对应 arm64
其他宿主	Docker / Docker Compose	v4.13.0 GHCR tag 须经 `linux/amd64` + `linux/arm64` multi-arch manifest 与匿名拉取核验；musl / Alpine、CentOS 7 与上述两种架构之外的平台不提供原生包
业务数据库	MySQL 8；PostgreSQL 14–18	为 AgentSQL 创建独立、最小权限运行账号
控制面存储	SQLite；PostgreSQL 15+	SQLite 为默认；PostgreSQL 可分离 metadata 与 audit
浏览器控制台	`http://127.0.0.1:7780`	默认绑定回环地址
本地 Live Demo	Docker Engine / Docker Desktop、Docker Compose v2	使用 v4.13.0 精确 tag 的 Compose 和经核验的 GHCR 镜像；无需克隆源码或构建

路径 A：5 分钟零配置看效果（一条命令）

这是 v4.13.0 专用 Demo 模式：使用合成数据和固定演示身份，无需克隆源码、无需编辑任何配置、无需构建镜像。执行前须核对 v4.13.0 Git tag 中的 Compose 文件与其引用的三个 GHCR 镜像均已公开且可匿名拉取；首次启动会拉取镜像并播种固定演示数据。

只需一个 `docker-compose.yml`（在任意空目录执行）：

```
# 1) 新建一个空目录并进入
mkdir agentsql-demo && cd agentsql-demo

# 2) 下载自包含 compose（仅这一个文件，约 2KB）
curl -fsSL -o docker-compose.yml
https://raw.githubusercontent.com/cuipengdba/agentsql/v4.13.0/deploy/quickstart/docker-compose.yml

# 3) 一条命令拉起全部容器（postgres + mysql + seed + gateway）
docker compose up -d
```

Windows（PowerShell）用等价命令：

```
New-Item -ItemType Directory -Force agentsql-demo | Out-Null; Set-Location agentsql-demo
Invoke-WebRequest -OutFile docker-compose.yml
https://raw.githubusercontent.com/cuipengdba/agentsql/v4.13.0/deploy/quickstart/docker-compose.yml
docker compose up -d
```

等待约 1–2 分钟，四个容器全部就绪后打开 http://127.0.0.1:17880：登录页已自动填好演示账号，直接点「登录」即可。也可以在终端确认进度：

```
docker compose ps            # gateway 状态变为 healthy 即就绪
docker compose logs -f seed  # 看到 DEMO_SEED_OK 表示播种完成
```

进入「演示台」后，按五张剧本卡逐个体验（每张卡都已内置准确的固定 SQL，点击即可看到真实结果）：

1. **列授权 + 脱敏**：密封 JOIN 查询放行，返回结果中 `phone`、`email` 两列被脱敏。
2. **列授权缺失**：查询未授权列被拒绝，不触达业务库。
3. **对象越权**：访问 `internal_notes` 被策略显式拒绝，不返回表内容。
4. **DDL 越权**：`DROP TABLE` 被拦截。
5. **写操作转人工**：`UPDATE` 不直接执行，转为 `approve` 等待人工审批。

体验完毕彻底清理（删除容器与演示数据卷）：

```
docker compose down -v
```

> Demo 说明：所有凭据均为固定演示值、写在 compose 中，仅用于本地体验，切勿连接真实数据或数据库。Demo 站点默认只绑定回环地址 `127.0.0.1`，不对局域网开放。

路径 B：接入你自己的库（真实生产闭环）

下面步骤必须按顺序完成，缺少授权策略时即使已经创建数据源和 Agent，也不能访问任何表。

1. 安装并启动

v0.4.0 起提供 linux/arm64 原生 glibc 包：一键安装支持 `linux/amd64`（`uname -m` 为 `x86_64` / `amd64`）和 `linux/arm64`（`aarch64` / `arm64`），均要求 glibc 2.28+、systemd，且命令需要 root 权限：

以下示例固定 v4.13.0。执行前先确认正式 Release 提供该版本的 `install.sh`、`SHA256SUMS` 和签名；按 发布资产说明 使用独立核验的公钥验签清单，再用清单中的哈希核验安装器。

```
# 适用前提：linux/amd64 或 linux/arm64 + glibc 2.28+ + systemd
curl -fsSLo install.sh https://github.com/cuipengdba/agentsql/releases/download/v4.13.0/install.sh
# 先用已验签的 SHA256SUMS 核验 install.sh，再运行：
sudo sh ./install.sh install --version v4.13.0
```

除一键安装外，也可按 部署指南 使用源码或 Docker Compose 构建。确认 v4.13.0 镜像已公开、双架构 manifest 和匿名拉取均通过后，运行 `docker pull ghcr.io/cuipengdba/agentsql:v4.13.0`；manifest 应匹配 `linux/amd64` / `linux/arm64`。发布状态和资产以 GitHub Releases 页面为准。

2. 登录控制台

打开 http://127.0.0.1:7780，默认管理员用户名是 `admin`。非交互安装不会把随机密码打印到终端；root 可读取 `/etc/agentsql/agentsql.env`，或重新安装时显式加 `--show-password`。

3. 添加数据源

先在业务库中为 AgentSQL 单独创建最小权限运行账号，绝不要使用 `superuser`、数据库 owner 或 MySQL `root`。在「数据源」填写 `host`、`port`、`database`、`username`、`password`，按实际查询需要授予最少的表权限，然后点击「测试连接」。

PostgreSQL 与 Vastbase G100 Build 9 PG 模式的指定实例可在数据源表单选择 `verify-full` 并填写网关可读的 CA 绝对路径及可选证书主机名。Vastbase 服务端须为该只读账号拒绝明文连接；其他数据库 TLS 能力按各自验收范围配置，详见 v1.8.0 TLS 验收。

4. 创建 Agent

在「Agent」创建身份，选择能力档位：

- `readonly`：只读查询。
- `dml`：查询及 `INSERT`、`UPDATE`、`DELETE`。
- `ddl`：包含结构变更能力。

保存创建后只显示一次的 API Key。请立即放入客户端的秘密存储，不要写入仓库或聊天记录。

5. 建立最小 allow 权限策略

> **关键步骤：AgentSQL 默认无授权即拒绝。光创建数据源和 Agent，仍访问不了任何表。**

在「权限」中新建绑定该 Agent 与数据源的 `allow` 策略，从确实需要的单张表开始，例如 `public.customers` 或 `customers`。不要用 `*` 或 `schema.*` 做普通示例：两者虽然合法，却代表整表全列授权，会让列级白名单失效。

安装 v4.13.0 后，控制台「接入向导」可复用同一个 Agent 和数据源选择做连接与权限预检、目标 SQL 离线解释，并生成 Dify 1.17.1 或 MCP Inspector 0.16.8 的无密钥接入说明。预检不执行 Agent SQL；策略数量大于零或离线结论为“未知”都不证明目标 SQL 已获授权。请在真实客户端分别验证允许与拒绝请求，再用响应中的审计 ID 跳转到审计详情核对。配置步骤与客户端可达地址的限制见使用手册。

6. 接入 MCP 客户端

已由安装器或 systemd 运行服务时，首选 Streamable HTTP。按 MCP 接入指南 把下列最小骨架合并到客户端配置；客户端具体文件路径和 HTTP transport 键名以该客户端官方 MCP 文档为准：

```
{
  "mcpServers": {
    "agentsql": {
      "url": "http://127.0.0.1:7780/mcp",
      "headers": {
        "Authorization": "Bearer <Agent API Key>"
      }
    }
  }
}
```

重启或重新加载客户端后，先调用 `list_datasources` 和 `list_schema`。再通过 `query` 发送一条已授权、带 `WHERE` 的单条 `SELECT`：

```
-- 适用前提：替换为已授权的真实表和列；通过 MCP query 工具提交
SELECT id, name FROM public.customers WHERE id = 1 LIMIT 10;
```

在「审计」页确认出现 `allow`。随后用 `readonly` Agent 通过 `execute_write` 发送一条无 `WHERE` 的更新，并填写 `reason`：

```
-- 适用前提：仅用于验证拦截；通过 MCP execute_write 工具提交并填写 reason
UPDATE public.customers SET name = 'blocked-test';
```

该请求应被 R002（无条件批量写）和/或 R003（只读 Agent 写操作）拒绝，不触达业务库；在审计页确认 `deny`。工具层前置拒绝可能发生在流水线之前，因此缺少 `reason` 等错误不保证产生审计；验收时必须填写 `reason`。

v0.7.0 起提供的 MCP 入口：支持 `2026-07-28` 的客户端可改用 `http://127.0.0.1:7780/mcp/2026-07-28`。每次 POST 都要携带 Agent Bearer Key、`MCP-Protocol-Version: 2026-07-28`、`Mcp-Method`，调用工具时还要携带与工具名一致的 `Mcp-Name`；请求体 `params._meta` 要声明同版协议及客户端能力。该入口无需 `initialize`，也没有旧入口的传输会话、GET 事件流和 B5 会话/计划事务工具。现有客户端继续使用 `/mcp`；请求示例与版本边界见 MCP 接入指南。

任务级临时只读 Key：申请、异人审批与结果预算

具备 `query.execute` 的用户在控制台「临时只读授权」提交精确 SQL、Agent、数据源、最长 24 小时的到期时间，以及 1–10 次结果预算（默认 1 次）。另一位同租户且同时具备 `strategy.manage` 与 `query.execute` 的用户查看 SQL 后批准或拒绝；批准后只有原申请人能一次性领取任务 Key。列表只显示 SQL 哈希与状态，密钥只在领取成功的响应中出现。该 Key 只能调用 MCP `query`，SQL 原始字节必须与申请时完全一致；原 Agent 的对象权限、规则、脱敏和数据库只读账号约束继续生效。用完可撤销，并可在「审计」页按授权 ID 核对管理和执行记录。v0.7.0 的直接签发 API 在本版返回 HTTP 410，既有 v0.7.0 Key 继续按到期与撤销规则生效。操作与恢复边界见 任务级临时只读授权。

5 分钟验收清单

- `GET /healthz` 返回 HTTP 200，JSON 中 `status` 为 `ok` 且包含 `version`。
- `GET /readyz` 返回 HTTP 200。
- 控制台「审计」页至少有 1 条 `allow` 与 1 条 `deny`。
- `deny` 详情显示命中规则，且目标业务表没有发生更新。
- API Key、数据库密码和 `AGENTSQL_SECRET` 未写入命令历史、仓库或公开日志。

```
# 适用前提：AgentSQL 已在本机 127.0.0.1:7780 运行
curl -fsS http://127.0.0.1:7780/healthz
curl -fsS -o /dev/null -w '%{http_code}\n' http://127.0.0.1:7780/readyz
```

v4.4.0：升级前只读检查

使用 v4.4.0 `agentsqlctl` 和目标环境配置先检查控制面 schema 版本。托管文件操作和 SQLBot 子绑定父级记录迁移已在 v4.2.0 引入；v4.4.0 沿用合库 metadata 第 29 版、分离 metadata 第 27 版和 audit 第 7 版。从 v4.1.0 或更早版本升级须成套备份 metadata 与 audit、全停旧版网关及问数服务、等待 B2 租约过期，并在隔离环境演练成套恢复。三网关步骤见维护窗口操作指南；不支持新旧二进制同时活跃：

```
agentsqlctl doctor --config /etc/agentsql/config.yaml --format json
```

退出码 `0` 表示所有已配置迁移流均为当前版本，`2` 表示存在待执行迁移，`1` 表示配置、连接或迁移历史等错误。该检查只读，不执行 DDL，也不替代备份或数据库结构与数据一致性检查。升级前须停止写入，成对备份控制面数据库及 `AGENTSQL_SECRET`；PostgreSQL 分库部署要同时备份 metadata 和 audit 库。再由 migration owner 显式迁移，步骤见 部署指南与 当前版本运维手册。已发布的 v0.6.0 安装包没有本节的新命令选项。

下一步

- 使用手册：按控制台菜单学习每项能力及边界。
- 敏感列发现指南：显式表范围、采样安全、置信度与脱敏草稿边界。
- MCP 接入指南：两种传输、11 个基础工具、B5 会话/事务工具、调用时序和客户端配置。
- 任务级临时只读授权：签发、撤销、精确 SQL 与审计边界。
- 部署指南：源码、Compose、systemd、升级、备份和控制面迁移。
- 当前版本运维手册：迁移、审计故障与新旧 MCP 入口排查。
- 本地 Live Demo：完整演示凭据、重置脚本与安全清单。