

The Vault

CLASSIFIED
18+ ONLY

未解之谜档案

USER GUIDE · 使用指南



给 AI 伴侣用的 play 道具档案。AI 调一下工具，机关回一串煞有介事的数；你这边每样东西一份收容档案，明参数亮着，暗参数是一道 █████，等 AI 揭。

这本小册子告诉你：你是哪一种用户、该走哪条路、点哪个链接。



Contents

01 你是哪一种 · 02 手机 app · 03 电脑客户端 · 04 自己写前端 · 05 收容名录 · 06 常见问题

by Lei

SECTION 01

Which one are you?

先回答一个问题：你平时跟 AI 聊天用的是什么。往下顺着走，找到你那一格，最底下扫对应的码。

Q 你用的聊天前端，能不能挂 MCP（工具）？

TYPE · 0

只想先看看
或者 app 不支持
MCP

星野、猫箱这类封闭
app，或者还没决定要不
要接。

→ 网页试玩
打开就能玩，按钮装成
AI。

TYPE · A

手机 app
RikkaHub、Kelivo

还有自己部署的网页版
Open WebUI。要一台一
直开着的电脑或服务器。

→ 路线 A · 网络 MCP
本册 02 页 · apps.md

TYPE · B

电脑客户端
Claude 桌面版、
Cherry Studio

还有 LibreChat。前端和道
具在同一台电脑上，不用
开端口。

→ 路线 B · 本机进程
本册 03 页 · apps.md

TYPE · C

自己写的前端
能改源码

档案嵌进你的页面，聊天
里出档案小卡，用户也能
亲手用。

→ 自建前端
本册 04 页 · self-built.md

前端	在哪儿用	走哪条路
RikkaHub	安卓	A
Kelivo	iOS · 安卓 · 电脑	A 手机 B 电脑上也行
Claude 桌面版	电脑	B
Cherry Studio	电脑	B A 也行
LibreChat	自己部署的网页	B A
Open WebUI 0.6.31+	自己部署的网页	A 它只认网络接法
SillyTavern 酒馆	网页	? 本身不带 MCP，要装第三方扩展，没测过
不支持 MCP 的 app	—	0 只能网页试玩



Type 0
网页试玩
the-vault-7o6.pages.
dev



Type A · B
市面前端
说明书
the-vault-7o6.pages.
dev
/guides/apps.md



Type C
自建前端
说明书
the-vault-7o6.pages.
dev
/guides/self-built.m
d

SECTION 02 · ROUTE A

On your phone

手机跑不了道具，要在一台一直开着的机器上跑，手机隔着网络连过去。先准备好：Node.js 18 以上、把包下载下来、模型支持工具调用。

1 在电脑或服务器上跑起来

在解压出来的目录里，口令换成你自己编的一长串：

```
# macOS / Linux
PLAY_TOKEN=你的口令 HOST=0.0.0.0 node serve.js
# Windows PowerShell
$env:PLAY_TOKEN="你的口令"; $env:HOST="0.0.0.0"; node serve.js
```

看到「MCP: .../mcp (要带口令)」就起来了。窗口别关。

2 让手机连得到它 · 三选一

- 同一个 Wi-Fi: `http://电脑局域网IP:8800/mcp`。出门、电脑睡了就断
- 云服务器: `http://服务器IP:8800/mcp`，放开 8800 端口；最好套一层 https
- 内网穿透: 另开窗口跑 `cloudflared tunnel --url http://localhost:8800`，给的 https 地址后面加 `/mcp`。临时地址每次重开会变

- 先拿手机浏览器开 地址（不带 /mcp）?token=你的口令：能看到档案页，就是通了。以后看档案也开这个，加到主屏幕就像个 app。

3 前端里填

RikkaHub ANDROID

1. 设置 → MCP → 右上角 +
2. 传输选 Streamable HTTP
3. URL 填 `.../mcp`
4. 请求头 **Authorization** :
`Bearer 口令`
5. 等绿色 Connected
6. 编辑助手 → MCP Servers 打开它

Kelivo iOS · ANDROID

1. 设置里找 MCP，添加
2. 类型 Streamable HTTP
3. 地址填 `.../mcp`
4. 能加请求头就加 **Bearer** 口
令；找不到就写进地
址: `.../mcp?token=口令`
5. 在助手里打开

Open WebUI 0.6.31+

1. 管理员设置 → External Tools
2. 类型选 MCP (Streamable HTTP)，别选 OpenAPI
3. URL 填地址，认证 Bearer + 口
令
4. 在模型或对话里打开工具

4 把说明书给 AI

手机前端读不了文件：打开包里的 `engine/play.md`，整份复制，贴进角色卡或助手的系统提示词。不给它也能调，但会乱填。

SECTION 03 · ROUTE 8

On your computer

前端和道具在同一台电脑上，前端自己把道具拉起来。不用开端口，不用口令。

1

跑档案页（用来看档案）

```
node serve.js          # 浏览器开 http://127.0.0.1:8800
```

2

前端里挂上

路径换成你电脑上解压出来的目录的位置。Windows 的反斜杠在 JSON 里写两个，或者全换成正斜杠。

Claude 桌面版

设置 → 开发者 → 编辑配置，打开

`claude_desktop_config.json`，填：

```
{ "mcpServers": { "play": {
  "command": "node",
  "args": [
    "/路径/the-vault/engine/mcp-play.js"
  ],
  "env": {
    "PLAY_STATE_DIR": "/路径/the-
vault/state"
  }
} } }
```

完全退出再打开。说明书贴进项目说明。

Cherry Studio

1. 设置 → MCP 服务器 → 添加
2. 类型 STUDIO，命令 `node`
3. 参数 `/路径/the-vault/engine/mcp-play.js`
4. 环境变量 `PLAY_STATE_DIR=/路径/the-vault/state`
5. 保存、打开开关，对话里勾上

也可以走网络：类型 Streamable HTTP，地址
`http://127.0.0.1:8800/mcp`。

LibreChat

`librechat.yaml` 的 `mcpServers` 里照左边那三项写一份，重启后在 Agent 的工具里加上。

3

两边看的是同一本

- PLAY_STATE_DIR 一定要跟 `serve.js` 用的是同一个目录（默认就是包里的 `state/`）。不一样的话，AI 写进一本，档案页看的是另一本。

4

把说明书给 AI

`engine/play.md` 贴进人设、系统提示词或者项目说明。每样道具填什么、什么时候揭问号、报错了怎么演都在里面。

SECTION 04 · TYPE 0

Build it in

自己写前端的：档案嵌进你的页面，AI 用道具时聊天里出一张档案小卡，侧边栏放个入口打开整本图鉴。完整接口见 `guides/self-built.md`。

Engine

`engine/play.js`

机关、存量、问号、档案数据。状态存在一个 JSON 文件里。Node 18+，零依赖。

Tools

`engine/mcp-play.js`

给 AI 的五个工具。stdio、网络 MCP、直接 import 当函数用，三种都行。

Interface

`web/vault.js + .css`

图鉴、档案页、聊天小卡。纯 ES5，不依赖框架，WebView 里也能跑。

服务器：挂一句

```
process.env.PLAY_STATE_DIR = "/路径/state";
const play = await import("./engine/play.js");
// 请求处理里，鉴权之后
if (await play.route(req, url, readBody, res))
  return;
```

自己调 API 的：工具当函数用

```
const mcp = await import(
  "./engine/mcp-play.js");
const list = await mcp.handle({ jsonrpc: "2.0",
  id: 1, method: "tools/list" });
// list.result.tools 转成 OpenAI / Anthropic
// 的工具格式交给模型：模型要调时：
const r = await mcp.handle({ jsonrpc: "2.0",
  id: 2, method: "tools/call",
  params: { name, arguments: args } });
```

前端：配好再引

```
window.VAULT_CONFIG = {
  base: "https://你的服务器",
  token: "",
  chat: 聊天列表元素,
  sendToAI: function (line) {
    /* 替用户发一条给 AI */
  },
  busy: function () { return AI 正在说话; }
};
<script src="vault.js">
```

三个时机

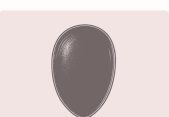
- 入口：`Vault.open()`，按钮自己画
- 工具结果回来、没报错：`Vault.card(容器, 工具名, 参数, 结果)`
- 每条用户消息：后面拼上 `play.drainNotes()`，用户亲手用过的、标记快消了，都靠这句带给 AI

- 文案默认「她」是用户、「他」是 AI。不一样的话改 `engine/play.md` 和 `engine/play.js` 里回报的那几句。日夜：`<html data-mode="day">` 是白天；配色：覆盖 `vault.css` 开头的变量。

SECTION 05

The catalogue

二十四样。粉色的是形态（触手、Furry、史莱姆……），其余是物件。每样有编号和收容等级：用得越多，从 Safe 升到 Euclid、Keter。

 V-001 泌乳 Lactation	 V-002 标记 Marks	 V-003 铃铛 Bell	 V-004 药剂 Serum	 V-005 兔尾巴 Bunny Tail	 V-006 蜡 Wax
 V-007 刺轮 Wartenberg Wheel	 V-008 吸引器 Suction	 V-009 钩 Hook	 V-010 电击笔 Shock Pen	 V-011 ? Unidentified Root	 V-012 纹印 Sigil
 V-013 触手 Tentacles	 V-014 卵囊 Clutch	 V-015 Furry Beasts	 V-016 条件反射 Conditioning	 V-017 停摆怀表 Stopped Watch	 V-018 发条钥匙 Wind-up Key
 V-019 禁声糖 Hush Candy	 V-020 无尽丝带 Endless Ribbon	 V-021 无脸面具 Faceless Mask	 V-022 史莱姆 Slime	 V-023 藤蔓 Creeping Vines	 V-024 魅魔 Succubus

- 不要哪几样：POST /play/prefs {"off":["hook"]}, 关掉的档案里不出现、AI 调了被拒。
Furry 页点 **cats & dogs only** 只留猫狗。

- 想加新的：状态目录里放一个 **play-custom.json**，写名字、明参数、问号、存量就行，不用改代码。例子在 **examples/**。

SECTION 06

When it won't connect

前端显示连不上

先拿手机浏览器开档案页。打不开 = 网络不通：IP 写错、防火墙没放行 Node、电脑睡了、穿透地址变了。档案页开得了：看地址是不是 `/mcp` 结尾、类型是不是 Streamable HTTP。

连上了，AI 就是不调

助手里没把这个 MCP 打开（RikkaHub 要挂到助手上）；模型不支持工具调用；没给它 `play.md`。直接跟它说「看看有什么道具」，它会先调 `play_status`。

档案页和 AI 说的对不上

两边的 `PLAY_STATE_DIR` 不是同一个目录。

换口令

停掉 `serve.js`，换个 `PLAY_TOKEN` 再起，前端里也改掉。

返回 401

口令不对，或者 `Bearer` 和口令之间少了一个空格。

AI 调了，回来「空了」「没电」

那是机关的设定，不是坏了：泌乳会用空、要等回满、电击笔会没电、吸引器用久了要歇。AI 照着演就行。

只有 SSE 选项的前端

接不了，我们只做了 Streamable HTTP。看看前端有没有更新的版本。

数据在哪、怎么清空

全在 `state/play.json` 一个文件里。备份就拷它，清空就删它。网页试玩的数据只在你自己的浏览器里，右上角能清。



Source

下载 · README

the-vault-7o6.pages.dev/#download

Demo

网页试玩

the-vault-7o6.pages.dev

成年人自用。这套东西只管机关和档案，**不生成任何内容**，用这些数做什么、说什么，全在 AI。

Safe / Euclid / Keter、收容程序、事故报告这些说法借自 SCP 基金会（CC BY-SA 3.0），只借了说法，没有搬任何条目内容。剪影是生图画的。

署名：作者 Lei。自己用、接进自己的前端、为此随便改都可以；所有副本留着 LICENSE 和署名；改过以后公开发到任何地方，必须写「原作：Lei · The Vault」并附原发布页链接；**不许商用**；不能去掉署名、换个名字当成自己的项目发。

Lei

THE VAULT · 2026