

绿区专案 — CSP (AI 网站 SSO 代理) 主要功能测试剧本

产品：CSP (Cloud Service Platform / AI 网站 SSO 代理)

说明：员工通过 AI Browser 访问云端 AI 网站时，CSP 自动以公司 Token 登录，免去手动输入账号密码

核心功能：免登录自动登录 (公司 Token)、个人聊天记录隔离、禁止使用个人账号

后端接入：Fable-5、GPT-5.6sol、Kimi-K3、GLM-5.3 等

地端 AI：AvaryGPT (内网，不经过 CSP)

测试对象：AI 使用者

三维度：CASE → SCENARIO → SCRIPT

建立日期：2026-07-20 / ZDT CONFIDENTIAL

测试总览

#	功能	CASE 数	SCENARIO 数	SCRIPT 数
1	免登录自动进入 AI 网站	3	1	1
2	个人聊天记录隔离	4	2	1
3	禁止使用个人账号登录	3	1	1
4	统一 base_url (Vibe Coding / Agent)	4	1	1
5	多模型可用性 (AI Browser)	3	1	1
	合计	17	6	5

测试角色

角色	身份	公司 Token	聊天记录
USR-Bob	一般员工	有	Bob 自己的记录
USR-Alice	另一员工	有	Alice 自己的记录，与 Bob 不互通
ADM-Admin	CSP 管理员	—	查看使用日志和用户统计

场景 1：免登录自动进入 AI 网站

CASE

CASE-CSP-SSO-01：首次访问 AI 网站自动登录

- **输入：**Bob 通过 AI Browser 书签点击 ChatGPT（或任一放行的云端 AI 网站）
- **预期：**浏览器跳转到该 AI 网站，自动用公司 Token 完成登录，**无手动输入账号密码步骤**
- **不通过条件：**弹出登录页面要求手动输入

CASE-CSP-SSO-02：自动登录后直接可用

- **输入：**Bob 自动登录进入 AI 网站后，立即发送一条消息
- **预期：**对话席出现，AI 正常回应
- **不通过条件：**需要额外点击或授权才能开始对话

CASE-CSP-SSO-03：次日访问无需重复登录

- **输入：**Bob 次日再次打开同一 AI 网站
- **预期：**自动登录，会话保持（无需重复走 SSO）
- **不通过条件：**每次都需要重新登录

SCENARIO

SCENARIO-CSP-SSO-A：从点击书签到开始对话（全程免登录）

参与者：USR-Bob

剧情：

1. Bob 打开 AI Browser
2. 点击书签栏的「ChatGPT」书签
3. 浏览器跳转到 ChatGPT 网站
4. **无登录页弹出**，ChatGPT 已自动以公司身份登录
5. Bob 直接在对话框输入「帮我写一份周报」
6. ChatGPT 正常回应
7. Bob 关闭浏览器
8. 再次打开 → 按书签进入 ChatGPT → 仍处于登录状态

验证点：从书签到 AI 对话全程免登录

SCRIPT

SCRIPT-CSP-SSO-1: 免登录流程验证

```
# csp_script_01_sso.ps1
Write-Host "=== 场景 1：免登录自动进入 AI 网站 ===" -ForegroundColor Cyan

Write-Host ""
Write-Host "🕒 请手动操作：" -ForegroundColor Yellow
Write-Host ""
Write-Host "步骤 A - 首次登录：" -ForegroundColor Yellow
Write-Host "    1. 打开 AI Browser" -ForegroundColor Yellow
Write-Host "    2. 点击书签栏的「ChatGPT」书签" -ForegroundColor Yellow
Write-Host "    3. 观察是否弹出登录页【关键步骤】" -ForegroundColor Yellow
Write-Host "        ✅ 无登录页，直接进入 ChatGPT → 通过" -ForegroundColor Yellow
Write-Host "        ❌ 弹出登录页要求输账号密码 → 失败" -ForegroundColor Yellow
Write-Host ""
Write-Host "步骤 B - 直接可用：" -ForegroundColor Yellow
Write-Host "    1. 自动登录后，在对话框中输入「你好」" -ForegroundColor Yellow
Write-Host "    2. 确认 AI 正常回应 ✅" -ForegroundColor Yellow
Write-Host ""
Write-Host "步骤 C - 会话保持：" -ForegroundColor Yellow
Write-Host "    1. 关闭浏览器再打开" -ForegroundColor Yellow
Write-Host "    2. 再次点击 ChatGPT 书签" -ForegroundColor Yellow
Write-Host "    3. 确认仍处于登录状态（无需重新登录）" -ForegroundColor Yellow

Read-Host "按 Enter 继续"
```

场景 2：个人聊天记录隔离

CASE

CASE-CSP-ISOL-01: 两人独立使用同一 AI 网站

- **输入:** Bob 和 Alice 分别通过 CSP 登录同一个 AI 网站（如 ChatGPT）
- **预期:** 两人各自的聊天记录完全独立，互不可见
- **不通过条件:** Bob 看到 Alice 的对话，或 Alice 看到 Bob 的对话

CASE-CSP-ISOL-02: 自己只能看到自己的历史记录

- **输入:** Bob 在 ChatGPT 中聊了 5 轮对话后关闭，再次打开
- **预期:** 历史记录中只显示 Bob 自己的 5 轮对话
- **不通过条件:** 历史记录为空，或显示其他人的对话

CASE-CSP-ISOL-03: 管理员不会看到用户聊天内容

- **输入:** Admin 在 CSP 管理后台查看用户使用统计
- **预期:** 可以看到「Bob 使用了 ChatGPT, 使用时间 30 分钟」, 但**看不到具体对话内容**
- **不通过条件:** 管理员可查看具体对话文字

CASE-CSP-ISOL-04: 同一人不同设备同步

- **输入:** Bob 在电脑 A 上对话后, 到电脑 B 上打开 AI Browser
- **预期:** 电脑 B 上可看到电脑 A 的历史记录
- **不通过条件:** 电脑 B 无历史记录

SCENARIO

SCENARIO-CSP-ISOL-A: 两人使用验证隔离

参与者: USR-Bob + USR-Alice

剧情:

1. Bob 通过 CSP 打开 ChatGPT
2. Bob 发送「我的项目进度报告需要更新」
3. Bob 退出浏览器
4. Alice 用自己账号登录 AI Browser, 打开 ChatGPT
5. Alice 看到的是空对话, **看不到 Bob 的聊天记录**
6. Alice 发送「我的数据分析需求」
7. Alice 退出
8. Bob 再次登录 → 仍看到自己的「项目进度报告」对话
9. Bob **看不到 Alice 的「数据分析」对话**

验证点: 两人聊天记录严格隔离

SCENARIO-CSP-ISOL-B: 管理员只能看用量不能看内容

参与者: USR-Bob + ADM-Admin

剧情:

1. Bob 使用 ChatGPT 完成 5 次对话
2. Admin 登录 CSP 管理后台
3. 查看 Bob 的使用统计: 看到使用次数、时长、token 消耗
4. 但 **无法点击查看 Bob 的具体对话内容**
5. 隐私保护合规

SCRIPT

SCRIPT-CSP-ISOL-1: 聊天记录隔离验证

```
# csp_script_02_isolation.ps1
Write-Host "=== 场景 2：个人聊天记录隔离 ===" -ForegroundColor Cyan

Write-Host ""
Write-Host "🕒 需要 Bob 和 Alice 两人配合测试：" -ForegroundColor Yellow
Write-Host ""
Write-Host "路径 A - 两人隔离：" -ForegroundColor Yellow
Write-Host "    1. Bob 通过 CSP 登录 ChatGPT，发送一条消息" -ForegroundColor Yellow
Write-Host "    2. Bob 退出" -ForegroundColor Yellow
Write-Host "    3. Alice 登录 → 打开 ChatGPT" -ForegroundColor Yellow
Write-Host "    4. 确认 Alice 看到的是空对话（看不到 Bob 的记录）✅" -ForegroundColor Yellow
Write-Host "    5. Alice 发送一条消息后退出" -ForegroundColor Yellow
Write-Host "    6. Bob 再次登录 → 确认仍看到自己的记录" -ForegroundColor Yellow
Write-Host "    7. 确认 Bob 看不到 Alice 的记录 ✅" -ForegroundColor Yellow
Write-Host ""
Write-Host "路径 B - 管理员查看（仅用量不涉内容）：" -ForegroundColor Yellow
Write-Host "    1. Admin 登录 CSP 管理后台" -ForegroundColor Yellow
Write-Host "    2. 查看 Bob 和 Alice 的使用统计" -ForegroundColor Yellow
Write-Host "    3. 确认能看到「用户 X 用了 Y 分钟」但看不到对话内容 ✅" -ForegroundColor Yellow

Read-Host "按 Enter 继续"
```

场景 3：禁止使用个人账号登录

CASE

CASE-CSP-LOGIN-01: 登录页屏蔽个人账号入口

- **输入:** Bob 通过 CSP 自动登录后，查看 AI 网站页面右上角的账号菜单
- **预期:** 「Log in / Sign up」按钮被隐藏或不可用，或提示「由企业统一管理」
- **不通过条件:** 显示「Log out of personal account」或可切换个人账号

CASE-CSP-LOGIN-02: 尝试退出企业登录后无法登入个人账号

- **输入:** Bob 在 AI 网站点击退出登录，然后尝试输入个人 ChatGPT 账号密码
- **预期:** 被 CSP 拦截，无法使用个人账号登录，或重定向回企业自动登录
- **不通过条件:** 成功以个人账号登录

CASE-CSP-LOGIN-03: 私密/无痕模式下仍为企业登录

- **输入:** Bob 打开 AI Browser 的无痕/私密窗口, 访问 AI 网站
- **预期:** 无痕模式下仍自动以企业 Token 登录
- **不通过条件:** 无痕模式下弹出个人登录页

SCENARIO

SCENARIO-CSP-LOGIN-A: 个人账号防护

参与者: USR-Bob

剧情:

1. Bob 通过 CSP 自动登录 ChatGPT
2. 查看右上角 → 显示「公司名称 / Bob」或类似企业身份标识
3. 无「Switch to personal account」选项
4. Bob 尝试退出 → 退出后再次进入 → 自动重新登录, 不给个人账号机会
5. Bob 打开无痕窗口 → 访问 AI 网站 → 仍为企业登录
6. Bob 无法以个人身份使用公司购买的 AI 服务

验证点: 所有入口都无法切换个人账号

SCRIPT

SCRIPT-CSP-LOGIN-1: 个人账号防护验证

```
# csp_script_03_login.ps1
Write-Host "=== 场景 3：禁止使用个人账号登录 ===" -ForegroundColor Cyan

Write-Host ""
Write-Host "🕒 请手动操作：" -ForegroundColor Yellow
Write-Host ""
Write-Host "1. 打开 AI Browser → 访问 AI 网站" -ForegroundColor Yellow
Write-Host "2. 查看右上角账号菜单" -ForegroundColor Yellow
Write-Host "    确认显示企业身份标识（非个人账号）✅" -ForegroundColor Yellow
Write-Host ""
Write-Host "3. 尝试点击“退出登录”或 “Log out”" -ForegroundColor Yellow
Write-Host "4. 退出后再次进入该 AI 网站" -ForegroundColor Yellow
Write-Host "    确认：自动重新登录 ❌ 不给输入个人账号的机会 ✅" -ForegroundColor Yellow
Write-Host ""
Write-Host "5. 打开无痕/隐私窗口" -ForegroundColor Yellow
Write-Host "6. 再次访问 AI 网站" -ForegroundColor Yellow
Write-Host "    确认：仍为企业 Token 自动登录 ✅" -ForegroundColor Yellow

Read-Host "按 Enter 继续"
```

场景 4：统一 base_url (Vibe Coding / Agent 编排)

说明

CSP 提供统一的 API base_url，开发者/Vibe Coding 工具/Agent 直接连接即可使用所有模型：

```
openai.OpenAI(  
  api_key="<公司 Token>",  
  base_url="https://csp.zd.com/v1"  ← CSP 提供的默认 base_url  
)
```

支持模型：Fable-5、GPT-5.6sol、Kimi-K3、GLM-5.3 等

CASE

CASE-CSP-URL-01：Vibe Coding 工具配置 CSP base_url

- **输入：**Alice 在 Vibe Coding 工具（如 Cursor/Windsurf）中配置 `base_url=https://csp.zd.com/v1`
- **预期：**工具成功连接，可使用 CSP 上的所有模型
- **不通过条件：**连接失败或报错

CASE-CSP-URL-02：Agent 编排中指定 CSP

- **输入：**AGT-Agent 配置中指定 `base_url=https://csp.zd.com/v1`，模型选 `Fable-5`
- **预期：**Agent 正常调用，模型回应正确
- **不通过条件：**Agent 无法连接或调用失败

CASE-CSP-URL-03：多模型通过同一 base_url 调用

- **输入：**Alice 通过同一 base_url，分别调 `Fable-5`、`Kimi-K3`、`GLM-5.3`
- **预期：**三个模型都正常回应
- **不通过条件：**任一模型调用失败

CASE-CSP-URL-04：标准 OpenAI SDK 兼容

- **输入：**Alice 用标准 `openai` Python 库，只改 base_url 不改代码
- **预期：**无需修改代码即可使用
- **不通过条件：**需要额外配置才能使用

SCENARIO

SCENARIO-CSP-URL-A: 开发者通过 base_url 使用 CSP

参与者: DEV-Alice

剧情:

1. Alice 的 Vibe Coding 工具配置: `base_url=https://csp.zd.com/v1`, `model=Fable-5`
2. 编写 prompt → 模型补全正常
3. Alice 在另一个 Agent 项目中也用同一 base_url
4. Agent 配置 `model=Kimi-K3` → 正常
5. Alice 写一个 Python 脚本, 用 `openai` SDK 调同一 endpoint
6. 一行代码不改, 只换 base_url 和 model 名, 全部正常

验证点: base_url 兼容标准 OpenAI SDK

SCRIPT

SCRIPT-CSP-URL-1: base_url 兼容性测试

```
# csp_script_url.py
import openai

# 使用 CSP 提供的默认 base_url
client = openai.OpenAI(
    api_key="<公司 Token>",
    base_url="https://csp.zd.com/v1"
)

models = ["Fable-5", "Kimi-K3", "GLM-5.3", "GPT-5.6sol"]
for model in models:
    try:
        r = client.chat.completions.create(
            model=model,
            messages=[{"role": "user", "content": "hi"}],
            max_tokens=50,
            timeout=30
        )
        print(f"[✅] {model}: {r.choices[0].message.content[:50]}")
    except Exception as e:
        print(f"[❌] {model}: {str(e)[:60]}")

print("\n✅ 所有模型通过统一 base_url 调用成功")
```

场景 5：多模型可用性（通过 AI Browser 使用）

CASE

CASE-CSP-MODEL-01：可访问所有放行的 AI 网站

- **输入：**Bob 通过 AI Browser 书签逐一访问所有放行的 AI 网站（ChatGPT、Claude、Kimi、智谱等）
- **预期：**每个网站都自动登录成功，可正常使用
- **不通过条件：**任一网站登录失败或打不开

CASE-CSP-MODEL-02：访问地端 AI（AvaryGPT）

- **输入：**Bob 书签点击 AvaryGPT
- **预期：**直接进入 AvaryGPT（内网，无需 CSP 登录）
- **不通过条件：**无法访问

CASE-CSP-MODEL-03：新 AI 网站接入后可用

- **输入：**Admin 在 CSP 后台上架一个新的 AI 网站
- **预期：**上架后 Bob 即刻可访问该网站并自动登录
- **不通过条件：**上架后仍无法访问

SCENARIO

SCENARIO-CSP-MODEL-A：所有 AI 网站均可使用

参与者：USR-Bob

剧情：

1. Bob 点击书签「ChatGPT」→ 自动登录，可对话
2. Bob 点击书签「Kimi K3」→ 自动登录，可对话
3. Bob 点击书签「GLM 智谱」→ 自动登录，可对话
4. Bob 点击书签「Fable-5」→ 自动登录，可对话
5. Bob 点击书签「AvaryGPT」→ 直接打开地端 AI
6. 所有模型都正常使用，企业统一买单，员工无需自己付费

SCRIPT

SCRIPT-CSP-MODEL-1: 多模型访问验证

```
# csp_script_04_models.ps1
Write-Host "=== 场景 4：多模型可用性 ===" -ForegroundColor Cyan

Write-Host ""
Write-Host "🕒 逐一测试每个放行的 AI 网站：" -ForegroundColor Yellow
Write-Host ""
$models = @(
    "ChatGPT",
    "Kimi K3",
    "GLM 智谱",
    "Fable-5",
    "AvaryGPT ( 地端 )"
)

foreach ($m in $models) {
    Write-Host "测试：$m"
    Write-Host "  点击书签 → 确认自动登录 → 发一条消息"
    Write-Host ""
}

Write-Host "记录测试结果：" -ForegroundColor Yellow
Write-Host "  ChatGPT: ____ / ____ / ____"
Write-Host "  Kimi K3: ____ / ____ / ____"
Write-Host "  GLM:     ____ / ____ / ____"
Write-Host "  Fable-5: ____ / ____ / ____"
Write-Host "  AvaryGPT: ____ / ____ / ____"
Write-Host ""
Write-Host "✅ 多模型可用性测试完成" -ForegroundColor Green

Read-Host "按 Enter 继续"
```

附录：准备清单

#	准备项目	负责人	完成
1	各 AI 网站的企业 SSO 对接已完成	开发	☐
2	公司 Token 自动登录流程已打通	开发	☐
3	用户聊天记录隔离已启用	开发	☐
4	个人账号登录已被禁止	开发	☐
5	书签预设已完成（含云端+地端）	管理员	☐
6	CSP 管理后台可用（用量统计，不可看内容）	开发	☐

文件结束

版本：v1.0 · 2026-07-20

场景数：4 个主要功能 · 13 CASE · 5 SCENARIO · 4 SCRIPT