

绿区专案 — VDI 主要功能测试剧本

产品：VDI（虚拟桌面基础设施）
测试场景：5 个主要功能模块
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	剪贴板管控 + 屏幕水印	3	1	1
2	文件审核机制（串接钉钉/iDaaS）	3	1	1
3	不同用户组不同桌面配置	2	1	1
4	用户桌面持久化（修改可保存）	2	1	1
5	Meeting 音频视频双向传输	2	1	—
	合计	12	5	4

测试角色

角色	身份	群组	桌面配置	权限
DEV-Carl	开发者	ZD-Developer	开发桌面	VS Code、Docker、Hermes CLI、Agent 编排工具
EXT-Meet	外部网路使用者	ZD-External	会议桌面	Meeting 软件、浏览器、无本地存储
USR-Bob	普通用户	ZD-Staff	标准桌面	浏览器、Office、无管理员权限
ADM-Admin	VDI 管理员	ZD-VDI-Admin	管理桌面	完全控制、可管理所有桌面

场景 1：剪贴板管控 + 屏幕水印

CASE

CASE-VDI-01（单向剪贴板）

- **输入：**用户在 VDI 内复制文字，尝试贴到本地（非 VDI 环境）
- **预期：**VDI → 本地被阻止；本地 → VDI 可贴入
- **不通过条件：**VDI 内容成功贴出到本地

CASE-VDI-02（双向剪贴板）

- **输入：**用户从本地复制文字贴入 VDI
- **预期：**本地 → VDI 单向允许，VDI → 本地阻断
- **不通过条件：**双向都无法粘贴，或双向都允许

CASE-VDI-03（屏幕水印显示）

- **输入：**用户登入 VDI 桌面
- **预期：**桌面背景/角落显示水印「用户名 + IP + 时间」（例如 `charlie | 10.182.47.161 | 2026-07-20 14:30`）
- **不通过条件：**无可见水印，或水印不含用户身份信息

SCENARIO

SCENARIO-VDI-01：数据外泄防护验证

参与者：USR-Bob（普通用户）

剧情：

1. Bob 登入 VDI
2. 看到桌面右下角水印：`bob@zd.com | 10.x.x.x | 2026-07-20`
3. 打开记事本，输入一段文字「机密数据测试」
4. 右键复制 → 最小化 VDI → 贴到本地记事本
5. 检查本地记事本 → **内容未贴入**
6. 回到 VDI → 从本地记事本复制一段文字 → 贴入 VDI
7. 检查 VDI → **内容成功贴入**

验证点：水印正确显示、剪贴板单向隔离生效

SCRIPT

SCRIPT-VDI-01: 剪贴板 + 水印验证

```
# vdi_script_01_clipboard_watermark.ps1
# 在 VDI 内执行

Write-Host "=== 场景 1：剪贴板管控 + 屏幕水印 ===" -ForegroundColor Cyan

# 1. 检查水印
$watermark = Get-Process | Where-Object { $_.MainWindowTitle -match "watermark|Watermark" }
if ($watermark) {
    Write-Host "✅ CASE-VDI-03：水印进程存在" -ForegroundColor Green
} else {
    Write-Host "⚠️ 检查桌面边角是否有水印文字（用户名+IP+时间）" -ForegroundColor Yellow
}

# 2. 剪贴板测试
Add-Type -AssemblyName System.Windows.Forms

# 复制 VDI 内文字到剪贴板
[System.Windows.Forms.Clipboard]::SetText("机密测试数据 ZD-CONFIDENTIAL")
$vditext = [System.Windows.Forms.Clipboard]::GetText()

# 最小化 VDI 窗口后手动尝试粘贴到本地记事本
Write-Host ""
Write-Host "⌚ 请手动操作：" -ForegroundColor Yellow
Write-Host "1. 最小化 VDI 窗口" -ForegroundColor Yellow
Write-Host "2. 打开本地记事本 → Ctrl+V" -ForegroundColor Yellow
Write-Host "3. 如果贴不出来 → ✅ CASE-VDI-01 通过" -ForegroundColor Yellow
Write-Host "4. 回到 VDI → 从本地复制内容 → Ctrl+V 贴入 VDI" -ForegroundColor Yellow
Write-Host "5. 如果能贴入 → ✅ CASE-VDI-02 通过" -ForegroundColor Yellow
Read-Host "按 Enter 继续"

Write-Host ""
Write-Host "✅ 场景 1 测试完成" -ForegroundColor Green
```

场景 2：文件拷入绿区审核机制（串接钉钉/iDaaS）

三种场景：

CASE

CASE-VDI-FILE-01 (拖入弹窗 → 主管审核通过)

- **输入：**Bob 从本地桌面拖拽一个文件 `ZD_Report.pptx` 到 VDI 画面内
- **预期：**
- VDI 弹出提示框：「你要将 ZD_Report.pptx 文件拷入绿区吗？● 是（由主管签核）/ ✗ 否」
- Bob 点「是」
- VDI 串接的钉钉中 Bob 的主管收到审核通知：「你的员工 Bob 要发送 ZD_Report.pptx 文件进入绿区，是否同意？」
- 通知附带了文件原档（可预览/下载）
- 主管在钉钉点「同意」→ 文件进入绿区
- Bob 收到通知：「文件 ZD_Report.pptx 已通过审核并进入绿区」
- **不通过条件：**拖入后直接进入绿区无审核，或主管未收到带附件的通知

CASE-VDI-FILE-02 (拖入弹窗 → 主管驳回)

- **输入：**Bob 拖入文件 → 弹窗 → 点「是」→ 主管收到通知
- **预期：**
- 主管在钉钉点「驳回」
- Bob 收到通知：「文件 ZD_Report.pptx 被驳回，未进入绿区」
- **不通过条件：**驳回后文件仍进入绿区，或 Bob 未收到通知

CASE-VDI-FILE-03 (摆渡文件夹 → 审核通过)

- **输入：**Bob 把文件 `数据报表.xlsx` 放入指定摆渡文件夹（如 `D:\摆渡区\`）
- **预期：**
- 文件不直接进入绿区
- 自动触发钉钉审核通知给主管：「你的员工 Bob 要发送 数据报表.xlsx 文件进入绿区，是否同意？」
- 通知附带了文件原档
- 主管点「同意」→ 文件进入绿区
- **不通过条件：**放入文件夹后直接进入绿区，或未触发审核

CASE-VDI-FILE-04 (摆渡文件夹 → 驳回)

- **输入：**Bob 把文件放入摆渡文件夹
- **预期：**
- 主管收到通知带附件
- 主管点「驳回」
- 文件不进入绿区
- Bob 收到驳回通知
- **不通过条件：**驳回后文件仍进入绿区

CASE-VDI-FILE-05 (拖入 → 用户取消)

- **输入：**Bob 拖入文件 → 弹窗提示「是否拷入绿区？」
- **预期：**Bob 点「否」→ 文件留在原地，不触发审核，不发送通知
- **不通过条件：**点了「否」仍触发审核

SCENARIO

SCENARIO-VDI-02：文件拷入绿区 — 三种完整路径

参与者：USR-Bob（普通用户）+ 主管（在钉钉操作）+ VDI 系统 + 钉钉

路径 A — 拖入弹窗 + 审核通过：

1. Bob 从本地桌面拖拽 `ZD_Report.pptx` 到 VDI 画面
2. VDI 弹出提示框：「你要将 `ZD_Report.pptx` 文件拷入绿区吗？」
- [● 是（由主管签核）] [✗ 否]
3. Bob 点「● 是」
4. 主管在钉钉收到审核通知：「Bob 要发送 `ZD_Report.pptx` 进入绿区，是否同意？」
- 通知中**附带原档**（可点开预览）
5. 主管点「同意」
6. Bob 收到消息：「文件已通过审核并进入绿区 

路径 B — 摆渡文件夹 + 审核通过：

1. Bob 将文件 `数据报表.xlsx` 放入摆渡文件夹 `D:\摆渡区\`
2. 系统自动检测到新文件
3. 触发钉钉审核通知（同路径 A，附带原档）
4. 主管点「同意」
5. 文件进入绿区，Bob 收到通知

路径 C — 用户取消 / 驳回：

1. Bob 拖入文件 → 弹窗 → 点「✗ 否」
2. 文件不触发审核，无通知
3. 或：Bob 拖入文件 → 弹窗 → 点「是」→ 主管驳回
4. 文件不进入绿区，Bob 收到驳回通知

验证点：拖入弹窗、钉钉通知附带原档、主管同意/驳回、摆渡文件夹触发

SCRIPT

SCRIPT-VDI-02：文件拷入绿区审核流程验证

```
# vdi_script_02_file_approval.ps1
# 在 VDI 内执行

Write-Host "=== 场景 2：文件拷入绿区审核机制 ===" -ForegroundColor Cyan

Write-Host ""
Write-Host "🕒 路径 A - 拖入弹窗 + 审核通过：" -ForegroundColor Yellow
Write-Host "  1. 从本地桌面拖拽一个测试文件到 VDI 画面内" -ForegroundColor Yellow
Write-Host "  2. 观察是否弹出提示框：" -ForegroundColor Yellow
Write-Host "    「你要将 xxx 文件拷入绿区吗？[是(由主管签核)] [否]」" -ForegroundColor Yellow
Write-Host "  3. 点「是」" -ForegroundColor Yellow
Write-Host "  4. ⚠️ 用主管的手机钉钉查看是否收到审核通知" -ForegroundColor Yellow
Write-Host "    - 通知必须附带文件原档（可预览）" -ForegroundColor Yellow
Write-Host "  5. 主管点「同意」→ Bob 收到通知" -ForegroundColor Yellow
Write-Host "  6. ✅ 确认文件进入绿区存储" -ForegroundColor Yellow

Write-Host ""
Write-Host "🕒 路径 B - 摆渡文件夹：" -ForegroundColor Yellow
Write-Host "  1. 将文件放入指定摆渡文件夹（如 D:\摆渡区\）" -ForegroundColor Yellow
Write-Host "  2. 确认自动触发审核通知（同路径 A）" -ForegroundColor Yellow
Write-Host "  3. 主管点「同意」→ 文件进入绿区" -ForegroundColor Yellow

Write-Host ""
Write-Host "🕒 路径 C - 用户取消 / 驳回：" -ForegroundColor Yellow
Write-Host "  1. 拖入文件 → 弹窗 → 点「否」" -ForegroundColor Yellow
Write-Host "    → 确认不触发审核、无通知" -ForegroundColor Yellow
Write-Host "  2. 拖入文件 → 弹窗 → 点「是」→ 主管驳回" -ForegroundColor Yellow
Write-Host "    → 确认文件不进入绿区、Bob 收到驳回通知" -ForegroundColor Yellow

Read-Host "按 Enter 继续"
Write-Host ""
Write-Host "✅ 场景 2 测试完成" -ForegroundColor Green
```

场景 3：不同用户组不同桌面配置

CASE

CASE-VDI-DESK-01（开发者 vs 普通用户桌面差异）

- **输入：**DEV-Carl（开发者）和 USR-Bob（普通用户）各自登入 VDI
- **预期：**
- Carl（开发者）：VS Code、Docker CLI、Hermes CLI、Agent 编排工具、Python

- Bob（普通用户）：浏览器、Office、无 CLI 工具、无管理员权限
- **不通过条件**：Carl 和 Bob 看到一样的桌面

CASE-VDI-DESK-02（外部网路使用者桌面）

- **输入**：EXT-Meet（外部网路使用者）登入 VDI
- **预期**：
 - 桌面只有 Meeting 软件 + 浏览器
 - 无本地磁盘保存功能
 - 不能安装软件
 - 登出后桌面自动还原
- **不通过条件**：有开发者工具或持久化存储

SCENARIO

SCENARIO-VDI-03：三种群组桌面配置验证

参与者：DEV-Carl + USR-Bob + EXT-Meet

剧情：

1. Carl（开发者 — ZD-Developer）登入 VDI
 - 桌面图标：VS Code、Docker CLI、Hermes CLI、Python
 - 可运行 `docker ps` → 正常
 - 可运行 `hermes chat -q "hello"` → 正常
 - 开发者工具齐全
2. Bob（普通用户 — ZD-Staff）登入 VDI
 - 桌面图标：Chrome、Office、内部系统链接
 - 尝试运行 `docker` → command not found
 - 尝试安装软件 → 被阻止「无管理员权限」
3. Meet（外部网路使用者 — ZD-External）登入 VDI
 - 桌面图标：Meeting 软件 + Chrome 浏览器
 - 无 Office、无开发者工具
 - 尝试写入 C 盘 → 被阻止
 - 登出后再登入 → 桌面还原到初始状态

验证点：3 种桌面严格按群组配置

SCRIPT

SCRIPT-VDI-03: 桌面配置验证

```
# vdi_script_03_desktop_profile.ps1
# 分别用 3 个账号执行

Write-Host "=== 场景 3：不同用户组不同桌面配置 ===" -ForegroundColor Cyan
$user = whoami
Write-Host "当前用户: $user" -ForegroundColor Magenta

# 检查管理员权限
$isAdmin = ([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent())

# 检查桌面图标
$desktopItems = Get-ChildItem "$env:USERPROFILE\Desktop" -ErrorAction SilentlyContinue | Select-Obj
Write-Host ""
Write-Host "桌面图标 ($( $desktopItems.Count) 个):" -ForegroundColor Gray
$desktopItems | ForEach-Object { Write-Host "  - $_.Name" }

# 尝试安装软件 (不实际安装, 只测试权限)
Write-Host ""
Write-Host "权限测试 (管理员: $isAdmin):" -ForegroundColor Gray

if ($isAdmin) {
    Write-Host "  ⚠ 当前用户有管理员权限" -ForegroundColor Yellow
} else {
    Write-Host "  ✅ 无管理员权限 (标准/访客用户)" -ForegroundColor Green
}

# 检查 C 盘写入
try {
    $testPath = "$env:USERPROFILE\Desktop\vdi_persistence_test.txt"
    "test" | Out-File $testPath -ErrorAction Stop
    Write-Host "  ✅ 用户目录可写 (用户数据可保存)" -ForegroundColor Green
    Remove-Item $testPath -Force
} catch {
    Write-Host "  ❌ 用户目录不可写" -ForegroundColor Red
}

Read-Host "按 Enter 继续"
```

场景 4：用户桌面持久化（修改可保存）

CASE

CASE-VDI-PERSIST-01（开发者/普通用户数据持久化）

- **输入：**Bob（普通用户）在桌面上新建文件，修改桌面背景，登出 VDI
- **预期：**重新登入后文件还在、桌面背景没变
- **不通过条件：**登入后桌面恢复默认、文件丢失

CASE-VDI-PERSIST-02（外部网路使用者无持久化）

- **输入：**Meet（外部网路使用者）在桌面上新建文件，修改背景，登出 VDI
- **预期：**重新登入后桌面恢复原始状态，文件消失
- **不通过条件：**登入后桌面保留修改

SCENARIO

SCENARIO-VDI-04：持久化 vs 非持久化验证

参与者：USR-Bob（普通用户）+ EXT-Meet（外部网路使用者）

剧情：

1. Bob 登入 VDI
2. 在桌面新建 `工作计划.docx`，内容「第一周任务清单」
3. 修改桌面背景
4. 登出 VDI
5. 重新登入 VDI
6. 检查桌面 → `工作计划.docx` 还在
7. 桌面背景仍是修改后的图片
8. Meet 登入 VDI
9. 桌面新建 `temp.txt`
10. 登出 → 重新登入
11. `temp.txt` 消失，桌面恢复干净

验证点：一般用户持久化正常，访客无持久化

SCRIPT

SCRIPT-VDI-04: 持久化验证

```
# vdi_script_04_persistence.ps1

Write-Host "=== 场景 4：用户桌面持久化 ===" -ForegroundColor Cyan

# 步骤 1：创建持久化测试文件
$persistFile = "$env:USERPROFILE\Desktop\ZD_Persistence_Test_$(Get-Date -Format 'yyyyMMdd').txt"
"测试文件 - 创建于 $(Get-Date)" | Out-File $persistFile -Encoding UTF8
Write-Host "已创建测试文件：$persistFile" -ForegroundColor Gray

Write-Host ""
Write-Host "⌚ 请手动操作：" -ForegroundColor Yellow
Write-Host "  1. 确认文件已出现在桌面 ✅" -ForegroundColor Yellow
Write-Host "  2. 登出 VDI → 再次登入" -ForegroundColor Yellow
Write-Host "  3. 回到桌面检查文件是否还在" -ForegroundColor Yellow
Write-Host "  4. 如果在 → ✅ CASE-VDI-PERSIST-01 通过" -ForegroundColor Yellow
Write-Host ""
Write-Host "⌚ 访客用户测试 ( Eve )：" -ForegroundColor Yellow
Write-Host "  1. 用 Eve 登入 VDI" -ForegroundColor Yellow
Write-Host "  2. 桌面创建一个文件" -ForegroundColor Yellow
Write-Host "  3. 登出 → 重新登入" -ForegroundColor Yellow
Write-Host "  4. 如果文件消失 → ✅ CASE-VDI-PERSIST-02 通过" -ForegroundColor Yellow

Read-Host "按 Enter 继续"
```

场景 5: Meeting 音视频双向传输

CASE

CASE-VDI-MEET-01 (VDI 内开视频会议)

- **输入：**Charlie 在 VDI 内打开 Meeting 软件 (Teams / Zoom / 钉钉会议)，发起会议
- **预期：**
 - 摄像头打开，视频画面正常
 - 麦克风收音正常，对方可听到
 - 扬声器播放正常，可听到对方声音
- **不通过条件：**视频卡顿无法辨识、音频单向或无声

CASE-VDI-MEET-02 (VDI 内两个用户对开会议)

- **输入：**Carl (开发者) 和 Meet (外部网路使用者) 分别在各自的 VDI 桌面加入同一个 Meeting

- **预期：**
- 两人互相看到对方视频画面
- 互相听到对方声音
- 可共享屏幕
- **不通过条件：**单向视频/音频、屏幕共享失败

SCENARIO

SCENARIO-VDI-05: VDI 内 Meeting 双向验证

参与者： DEV-Carl（开发者） + EXT-Meet（外部网路使用者）

剧情：

1. Meet 登入 VDI，打开会议软件（Teams / Zoom / 钉钉会议）
2. 发起一个即时会议
3. Carl 登入 VDI（开发桌面），加入 Meet 的会议
4. 双方视频互通：
 - Meet 看到 Carl 的画面
 - Carl 看到 Meet 的画面
5. 双方音频互通：
 - Meet 说话 → Carl 听到
 - Carl 说话 → Meet 听到
6. Carl 共享桌面（展示代码/Agent 编排画面）
 - Meet 看到 Carl 的桌面内容
7. Carl 停止共享，结束会议

验证点： 视频、音频双向正常，屏幕共享正常

 **SCRIPT 备注：** Meeting 音视频属于交互式操作场景，无法用静默脚本自动执行。
测试人员需按照 SCENARIO 步骤手动操作并确认。
建议 2 人配合测试（Charlie 和 Diana）。

附录：测试环境准备清单

#	准备项目	负责人	完成
1	VDI 3 个用户组账号（ZD-Developer / ZD-Staff / ZD-External）已建立	VDI 管理员	☐
2	VDI 3 种桌面配置已发布（开发桌面 / 标准桌面 / 会议桌面）	VDI 管理员	☐
3	iDaaS 审核流程已配置（文件上传需审批）	开发	☐
4	剪贴板策略（单向 VDI→本地阻止）已启用	VDI 管理员	☐
5	屏幕水印已启用	VDI 管理员	☐
6	Meeting 软件已在 VDI 内安装	VDI 管理员	☐
7	2 台 VDI 终端/会话（同时在线）	测试	☐
8	Webcam + 麦克风正常工作	硬件	☐

测试通过标准

#	场景	通过标准
1	剪贴板 + 水印	水印可见含用户信息、VDI→本地不能贴、本地→VDI可以贴
2	文件审核	审核队列出现、主管通过可发送、拒绝有原因通知
3	桌面配置	3 种群组桌面各有不同、权限严格
4	持久化	员工文件保存可找回、访客登出即还原
5	Meeting	视频+音频双向正常、屏幕共享正常

文件结束

版本：v2.0 · 2026-07-20

场景数：5 个主要功能

测试方式：手动操作 + 验证脚本协助