












PluginOK 中间件升级包制作说明书

PluginOK 中间件不同于传统的客户端软件，本身给前端提供了开发接口，可实现强制升级中间件本身，也可以安装或升级其它小程序包，这个过程可以做到不需要终端电脑人员确认，不弹升级提示窗口，做到无感知升级。下面重点给您介绍如何实现这个自动升级。

第一步、制作中间件自身的升级包

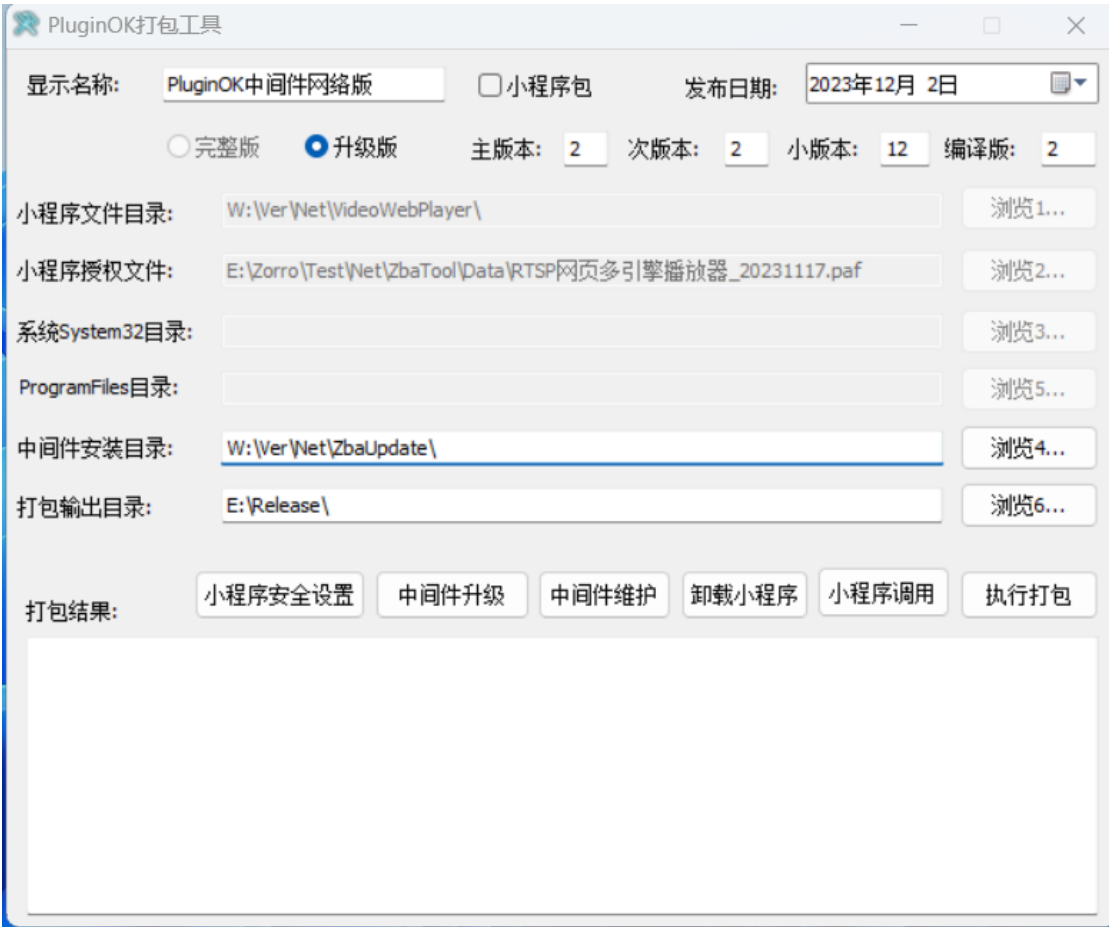
对比中间件新旧文件拿到文件的更新列表，一般来说，中间件的 EXE 和 DLL 等每次发版本都是同步更新，同时可能还有个别的配置文件更新，把这些更新文件单独存放到一个目录如 E:\Zorro\Ver\Net\ZbaUpdate 备用，注意升级文件列表中还要同时包含 X64 目录的所有文件，如需要同时升级某些小程序的文件，可直接放到 Plugins 子目录下小程序目录即可。如使用的是网络版时，一定要注意 ZbaConfig.json 中配置的授权服务器信息是对的，下面以网络高级版为例说明，如下图所示：

名称	修改日期	类型	大小
 TestWrl.txt	2023/12/2 9:05	文本文档	21 KB
 ChangeLog.txt	2023/12/2 8:57	文本文档	56 KB
 ZbaSetup.exe	2023/12/1 18:28	应用程序	566 KB
 ZbaAgent.exe	2023/12/1 18:28	应用程序	344 KB
 ZbaHttp.dll	2023/12/1 18:28	应用程序扩展	2,185 KB
 ZbaService.exe	2023/12/1 18:28	应用程序	298 KB
 ZbaBrowser.dll	2023/12/1 18:28	应用程序扩展	410 KB
 ZbaEngine.dll	2023/12/1 18:28	应用程序扩展	1,634 KB
 ZbaApplet.dll	2023/12/1 18:28	应用程序扩展	218 KB
 ZbaBase.dll	2023/12/1 18:28	应用程序扩展	903 KB
 X64	2022/4/7 8:08	文件夹	

以上文件列表中，如果是网络版，需要去掉 ZbaAuth.wdb 文件，如您使用的是标准版，可以去掉 ZbaApplet.dll 和 ZbaBrowser.dll 这两个动态库文件(包括 X64 下的同名文件)。

找到 SDK 开发包中的打包工具程序 ZbaSDK\Package\ZbaPackage.exe(单机版对应是 WrlSDK\Package\WrlPackage.exe)并启动，设置显示名称，更新日期，版本号，其中版本号请从 ChangeLog.txt 中获取，每次版本是固定的，注意单机版和网络版版本号是不一样的，不能弄错，另外如下截图中的中间件安装目录，是指

需要安装到中间件目录的升级文件所在的目录，支持子目录，注意不是中间件当前实际运行的目录，另外记得把小程序包的勾选去掉。具体配置如下：



然后点击执行打包，会生成中间件的升级指令 JSON 包，自己再加上几条关键的升级描述和在服务器上的实际文件下载地址，并把打包输出目录生成的 Zba_Update.pid 文件上传到可以通过 HTTP 协议下载的服务器之上，最新形成类似以下内容的 JSON 配置文件：

```
{
  "req": "Wrl_Update",
  "rid": 1,
  "para": {
    "Name": "BrowserApplet 中间件网络版升级包",
    "Date": "2023-12-02",
    "Desc": "1、优化中间件网络版生成唯一标识 UniID 算法，解决个别电脑生成相同标识问题；2、解决中间件安装时如 VLC 还未安装时弹出警告问题，增强中间件升级时对数据访问的稳健性；3、PageHiOffice 网页组件改进带公式内容转换图片算法，支持获取文档修订及批注详细内容；4、PageHiPlayer 使用 VLC 播放引擎时支持旋转或翻转视频画面播放，优化对 VLC 低版本及 XP 等 ...",
    "DownAddr": "http://local.zorrosoft.com/Files/Update/Zba_Update.pid",
    "MD5": "CF328A5D9D72A03828651B5DACA1E868",
    "Version": "2.2.12.2",
    "Size": 14090240,
    "HideIns": 0,
    "Cookie": "",
    "Auth": "",
    "Open": "",
    "TK": "452232B908586E90F0045EB0752E208C14DA9E0CA8F9DCFE471F0E708185AD243C2BAC8C3E0AF8DAC9D29CDF239B62864D48D421FDB65549CE529C4F0889E2117FB00136CA7DDF1DD2D5A1548F90B51B5C5EAE5160873C8AC6C9BBB2880F4C289A67C60373092261AFED47F0B194EB7F397C89C280BAC0038E13B0E9627DB35C8DC917333FB3A248A966AB05152DD646618202857B0B0D017DBFFABC5A21875622AECB336DAC48183B2E1FBA530968A8669138FDFF7ECFE0F9320DAF940AC9B91BC872A4BB98AE1E74A7E0D18D15938942B3AEAE8"
  }
}
```

FE05DCE109DD4AE1DB5D2180963C3998252FE69F015CE0466A3021C303ABC6477E
C77589F91FC939E975D7E"}}}

以上升级请求就是下一步实现本次升级的 JSON 配置，高级版授权中支持 HideIns 改为 1 可实现静默安装。在制作中间件的升级包时，可以在 Plugins 下放入自己的小程序升级文件，这样可以实现同时升级的效果。















第二步、前端代码编写实现升级

- 1、前端 JS 脚本连接指定端口的中间件后，发送以下 JSON 包内容：
{ "req": "Wrl_Version", "rid": 2, "para": { "Version": "2.2.11.1" } }, 中间件会返回是否升级的信息，Version 需要修改为最新的升级版本号，如返回 { "ret": 0, "rid": 2, "req": "Wrl_Version", "data": { "Update": 1 } }, 代表需要升级，就是目前终端电脑的中间件版本不是最新的版本，如 Update 为 0 则无需升级；
- 2、前述步骤 Update 为 1 需要升级时，再请求发送上一步最后生成的 JSON 内容(注意升级 JSON 包中需要下载的 pid 文件，需要放到自己的 B/S 系统服务器上，支持 HTTP 下载，并填写实际下载路径)，即可启动中间件的升级直到结束。中间件升级过程中，会自动断开前端到中间件的 WS 连接，前端可酌情弹出一个提示请用户耐心等待升级完成，然后定时再连接中间件指定端口，一旦可以连接上，就代表中间件升级完成。

你可能会疑问，在中间件升级过程中，能不能同时做小程序的升级呢？当然是可以的，只是一般不推荐这么做，因为只有在中间件本身有升级的时候，才可以同时加上小程序的升级，具体做法就是在中间件的文件更新目录下，把小程序需要升级的文件按实际运行时对应的目录结构同时放入即可。如果中间件本身没有需要升级的，可以对小程序进行单独的升级包制作。下面是中间件未升级的情况下，如何单独升级小程序的做法：

第三步，制作指定小程序的升级包

- 1) 同样需要把小程序的更新文件单独存放到某个干净的目录下，下面就以制作 VLC 网页播放小程序为例说明，将 VLC 网页播放小程序的文件放到 F:\Zorro\Ver\Net\VLCWebPlayer 目录下，如下图所示：

名称	修改日期	类型	大小
 VideoProxyPlayer.exe	2023/12/1 18:29	应用程序	179 KB
 VlcWebPlayer4.exe	2023/12/1 18:29	应用程序	680 KB
 VlcPlayerApplet.exe	2023/12/1 18:29	应用程序	680 KB
 PluginConfig.json	2023/12/1 8:47	JSON 文件	1 KB
 VlcOcxTest.exe	2023/11/30 11:51	应用程序	113 KB
 VlcOcx.dll	2023/11/30 11:50	应用程序扩展	506 KB
 Vlc4Ocx.dll	2023/11/30 11:50	应用程序扩展	506 KB
 Vlc2Ocx.dll	2023/11/30 11:50	应用程序扩展	498 KB
 Config.json	2023/11/13 15:20	JSON 文件	1 KB
 PHWebControl.dll	2023/7/1 17:03	应用程序扩展	404 KB
 Remove.json	2022/4/13 10:43	JSON 文件	1 KB
 VLC.png	2021/6/19 0:23	PNG 文件	16 KB
 Logo.png	2018/1/26 0:57	PNG 文件	14 KB
 X64	2023/11/30 11:56	文件夹	

请先编辑好 **PluginConfig.json** 中的版本信息和更新日期。小程序的版本号可以自己指定，即使 EXE 文件本身没有更新，也可以通过升级版本来更新其它文件。一般来说，这个目录必须有 **PluginConfig.json** 和对应的 EXE 执行文件，否则打包过程中会报错。如果是 C# 的 DLL 小程序，对应的 DLL 文件需要单独存到一个目录，然后在打包时的中间件安装目录对应就设置这个目录。其它小程序一般无需指定中间件安装目录，除非您确实有文件需要安装到中间件主程序目录下。如下图所示：



> Ver > Net > Wrl

名称

PluginNetDll.dll

> Ver > Net > PluginNetDll

名称

PluginConfig.json

2) 同样找到 SDK 开发包中的打包工具程序 ZbaSDK\Package\ZbaPackage.exe(单机版对应是 WrlSDK\Package\WrlPackage.exe)并启动, 在设置了小程序文件目录和小程序授权文件后, 显示名称, 更新日期, 版本号会自动从配置文件 PluginConfig.json 中获取, 版本号一般不需要再编辑, 具体配置如下:



然后点击执行打包，会生成小程序的升级指令 JSON 包，加上实际的文件下载地址，并把打包输出目录生成的 VlcWebPlayer_Update.pid 文件上传到可以通过 HTTP 协议下载的服务器之上，最新形成类似以下内容的 JSON 配置文件：

```
{
  "req": "Plugin_Update",
  "rid": 1,
  "para": {
    "Name": "VLC 网页播放器",
    "PID": "VLCWebPlayer",
    "Date": "2023-11-19",
    "Desc": "PluginOK 之上跨浏览器的 VLC 网页播放小程序，支持在 Chrome、Edge、Firefox、IE、Oprea、360、QQ 等最新版本浏览器网页中直接内嵌使用，可播放多路海康及大华等设备的实时视频",
    "DownAddr": "http://local.zorrosoft.com/Files/Net/VlcPlayerApplet_Update.pid",
    "MD5": "FF59D839DB712DC34B2B26C7C78D745A",
    "Version": "2.2.12.1",
    "Size": 8257536,
    "HideIns": 0,
    "Type": 8,
    "Cookie": "",
    "Auth": "",
    "TK": "24960A332DB88C3E45906C62C5D72D2B59AE05B2E0F9C69E577998B5A2F7D32931FFE79D3836E904C10BD5F3A33248D8B3E0E30419C8B25823F2364548DDB2E2155E0D65CF47BD09FCF5D73DD420FF83AC7A37A69E387176A634946B0191CE79B72ADF9ABC6160AEA3AB438A6EFA703805CFF7599D822434F286051F08B77E4CA6F8811B364E049942C810240612FC67780684755B9A8D137794B42E3C8C81E39CD241BEEAF200F0E67BECF2E17AFA07BB2311EB9BFD2D32A34FF9C070B59DC2ADCCF8E29B327DEC9E07E8EA609B94BEF3212FE311CAEA82B09268AF85B511E9837AADC05A7E3C015442AA1AA1A04189AB771D66B91D469A5B6CB30D4DA45C4"}
}
```

以上内容就是第二步需要的本次升级 JSON 配置。HideIns 改为 1 可实现静默安装。当前前提是您的中间件授权中，支持静默安装特性，一般是高级版才提供。

第四步 前端代码编写实现升级

- 1) 前端 JS 脚本连接指定端口的中间件后，发送以下 JSON 包内容：
`{"req":"Wrl_Version","rid":3,"para":{"PID":"VLCWebPlayer","Version":"2.2.12.1"}}`
，Version 需要修改为最新的升级版本号，中间件会返回是否升级的信息，如返回`{"ret":0,"rid":3,"req":"Wrl_Version","data":{"Update":1}}`，代表需要升级，就是目前终端电脑的此小程序版本不是最新的版本，如 Update 为 0 则无需升级，做升级的前提是指定的小程序已经存在，如果不确定的话，可以通过如下指令获得是否存在的信息：`{"req":"Plugin_Exist","rid":19,"para":{"PID":"VLCWebPlayer"}}`；
- 2) 前述步骤 Update 为 1 需要升级时，再请求发送第一步最后生成的 JSON 内容，即可启动小程序的升级，直到结束，在升级过程中，会不断收到小程序的下载和升级进度，直到收到类似这样的信息`{"event":"Plugin_Install","data":{"PID":"VLCWebPlayer","Status":16,"Info":"安装成功"}}`时，代表升级成功。

Update 为 0 时，也是可以请求执行 Plugin_Install 或 Wrl_Update 的，这种情况一般是为了恢复安装，解决可能部分文件可能是删除的问题。

关于 PluginOK 中间件打包工具的详细使用说明，请参考 SDK 包中的文档“PluginOK 开发者手册.pdf”，如还有问题，可加以下企业微信咨询：

