

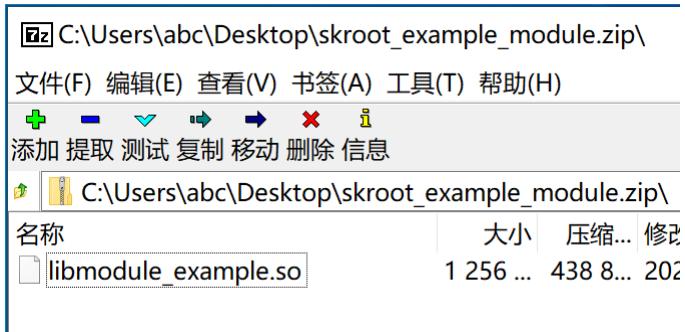
# SKRoot(Pro) 模块打包指南

## 一、 打包一个简单的 SKRoot 模块。

1. 包含头文件 `#include "kernel_module_kit_umbrella.h"`
2. 编写模块入口 `skroot_module_main` 与 “模块名片”

```
// SKRoot 模块名片
SKROOT_MODULE_NAME("演示模块名称")
SKROOT_MODULE_VERSION("1.0.0")
SKROOT_MODULE_DESC("演示模块描述")
SKROOT_MODULE_AUTHOR("演示作者名字")
SKROOT_MODULE_UUID32("3608c9af28db4dcfc05c32bbc584753e")
```

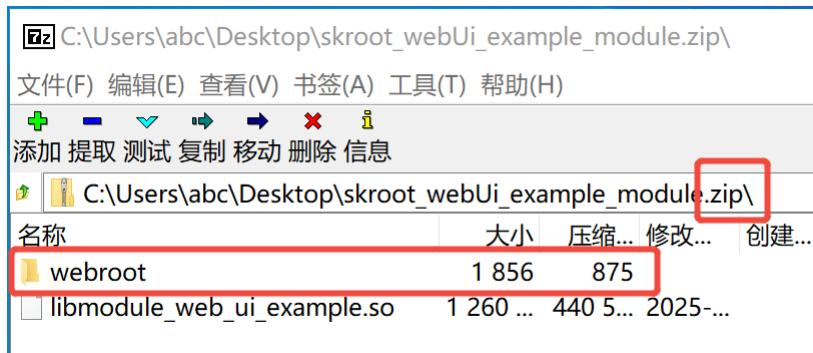
3. 使用 Android NDK 编译出.so 文件，再压缩成 **ZIP 压缩包**，就是模块产物了。



参考项目：**module\_example**

## 二、打包一个具有 WebUI 页面的 SKRoot 模块。

**方法 1：**新建 wwwroot 文件夹并在里面放入网页资源文件，首页命名为 index.html，然后将 wwwroot 文件夹一起打包进 ZIP 压缩包。



**方法 2：**在源码处的“模块名片”添加 **SKROOT\_MODULE\_WEB\_UI**，并填入 HTTP 回调响应类。

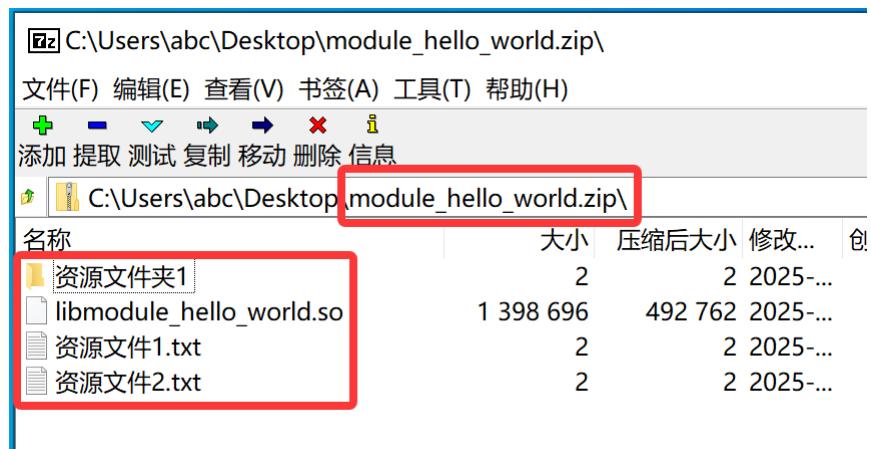
```
// SKRoot 模块名片
SKROOT_MODULE_NAME("内置 WebUI 示例模块")
SKROOT_MODULE_VERSION("1.0.0")
SKROOT_MODULE_DESC("演示在模块中使用WebUI页面")
SKROOT_MODULE_AUTHOR("SKRoot官方教程")
SKROOT_MODULE_UUID32("6080b19fb2db26c534af3051103f541f")
SKROOT_MODULE_WEB_UI(MyWebHttpHandler)
```

以上两种实现方法可并存。

参考项目：*module\_web\_ui\_example*

### 三、 打包一个附带资源文件的 SKRoot 模块。

- 直接一起打包进 ZIP 压缩包就可以了，安装模块时会自动释放到模块运行目录下。



- 在编程中，可使用 module\_private\_dir 访问此目录。

```
int skroot_module_main(const char* root_key, const char* module_private_dir) {
    return 0;
}
```