

# 用户手册

- [C/C++接口使用方法](#)
- [Python3 接口使用方法](#)
- [JS 接口使用方法](#)
  - [Web 异步接口使用方法](#)
  - [Web 同步接口使用方法](#)
  - [Vue 接口使用方法](#)
  - [NodeJS 接口使用方法](#)
  - [TypeScript 接口使用方法](#)

## C 接口使用方法

1. 获取到接口包后，解压缩，并找到对应的接口安装包，安装接口包；

```
sudo dpkg -i lpapi_x.x.x_xxx.deb
```

2. 使用 lpapi 接口

- 如果项目使用的是 cmake 进行编译的，可以在 CMakeList.txt 中通过 find\_package()来配置头文件及链接库

```
# 查找LPAPI模块
find_package(LPAPI REQUIRED)
if(LPAPI_FOUND)
    target_include_directories(${TEST_TARGET} PRIVATE ${LPAPI_INCLUDE_DIR})
    target_link_libraries(${TEST_TARGET} ${LPAPI_LIBRARY})
endif()
```

- 如果没有使用 cmake，直接添加头在安装位置中找到对应的头文件及动态库，直接使用即可，头文件及动态库位置如下：
  - lpapi.h 位于/usr/local/include/lpapi/lpapi.h
  - liblpapi.so 位于/usr/local/lib/lpapi.so
- 具体使用方法可参考 demo/ctest 目录；

## Python3 接口使用方法

- 中压缩包中找到 lpapi-x.x.x-xxx.whl 文件，通过 pip3 安装接口包

```
pip3 install lpapi-xxx.whl
```

- 在 pytho 代码中通过 import 引入 lpapi 接口，开始使用

```
from LPAPI import lpapi
```

具体使用方法参考 demo/python3/test\_lpapi.py

## JS 接口使用方法

### 1. 安装 dtpweb 打印服务接口

所有的 js 打印接口都是基于 dtpweb 服务来实现的，所以使用 js 打印接口前需要先安装 dtpweb 服务，dtpweb 服务是集成在 lpapi 接口保重的。

从接口包中找到指定架构的接口安装程序，安装接口包 lpapi-xxx.deb；

```
sudo dpkg -i lpapi-xxx.deb
```

## web 接口使用方法

js 接口是一种异步接口，是通过 Promise 来实现异步处理的，低版本浏览器，或者不习惯使用 Promise 的用户可以使用同步接口来进行打印处理。

- 安装 lpapi 接口；

```
sudo dpkg -i lpapi-xxx.deb
```

- 引入 dtpweb.js 文件

```
<script type="text/javascript" src="./dtpweb/lib/dtpweb.js"></script>
```

- 在 window.onload 中获取 api 接口；

```

var api = null;
window.onload = function () {
    dtpweb.DTPWeb.getApi().then(function (_api) {
        if (_api) {
            api = _api;
            updatePrinterList();
        } else {
            alert("未检测到打印机插件");
        }
    });
};

/**
 * 更新打印机列表。
 */
async function updatePrinterList() {
    if (api) {
        var printers = await api.getPrinters(true, false);
        // 清空旧的列表
        printerElements.innerHTML = "";
        // 重新添加打印机列表
        if (printers && printers.length > 0) {
            console.log(printers);
        } else {
            console.log("未检测到打印机");
        }
    }
}

```

**备注：**具体可参考：<demo/html/index.html>

## web 同步接口使用方法

在一些低版本浏览器中，或者不习惯使用 Promise 异步接口的用户可以使用同步接口进行打印处理

- 安装 dtpweb 插件

```
sudo dpkg -i lpapi-xxx.deb
```

- 引入同步 dtpweb.js 文件

```
<script type="text/javascript" src="./dtpweb/lib_sync/dtpweb.js"></script>
```

- 获取打印接口，开始使用

```

var api = null;
window.onload = function () {
    dtpweb.DTPWeb.getApi(function (_api) {
        if (_api) {
            api = _api;
            updatePrinterList();
        } else {
            alert("未检测到打印机插件! ");
        }
    });
};

/**
 * 更新打印机列表。
 */
function updatePrinterList() {
    if (api) {
        var printers = api.getPrinters(true, false);
        if (printers instanceof Array && printers.length > 0) {
            console.log(printers);
        } else {
            console.log("未检测到打印机");
        }
    }
}

```

**备注：**具体可参考：<demo/html/index.html>

## VUE 接口使用方法

1. 安装 dtpweb 服务；

```
sudo dpkg -i lpapi-xxx.deb
```

2. 引入 dtpweb.js 接口

- 方法一：通过 npm/yarn 方式引入；

```
npm install dtpweb
```

或者：

```
yarn add dtpweb
```

- 方法二：

直接将 dtpweb/lib/dtpweb.js 复制到项目中合适的位置；

### 3. 通过 import 导入 dtpweb

```
<script>
import { DTPWeb } from "dtpweb";

    mounted() {
        this.api.checkPlugin().then((value) => {
            if (value) {
                this.updatePrinterList();
            } else {
                this.api = undefined;
                this.printers[0].name = "未检测到打印插件";
            }
        });
    },
</script>
```

**备注：**具体可参考：demo/vuetest/App.vue

## NodeJS 接口使用方法

使用方法参考[VUE 接口使用方法](#)

## Typescript 接口使用方法

1. 安装 dtpweb 服务；

```
sudo dpkg -i lpapi-xxx.deb
```

2. 将 dtpweb/src 目录中的所有源代码复制到项目中即可
3. 具体使用方法可参考[VUE 接口使用方法](#)