

**Bplus CT测试手册**

产品名称：BplusCT测试

产品型号：

文档编号：W0000001

文档密级：秘密

(QR21-00-04) 版权所有 不得复制

**文档修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **版本** | **内容** | **修订人** | **日期** | **审核人** | **批准人** | **生效日期** |
| V0.1 | 创建初稿 |  | 2020/04/22 |  |  |  |

注：对该文件内容增加、删除或修改均需填写此变更记录，详细记载变更信息，以保证其可追溯性。初建版本号从V1.0开始，之后每次以0.1递增，遇规模较大的升级，则直接跳至2.0或3.0，以此类推。

目 录

[1 文档约定 3](#_Toc38443649)

[2 术语、定义和缩略语 4](#_Toc38443650)

[2.1 术语、定义 4](#_Toc38443651)

[2.2 缩略语 4](#_Toc38443652)

[3 相关文档 5](#_Toc38443653)

[4 概述 6](#_Toc38443654)

[5 环境搭建 6](#_Toc38443655)

[5.1.1 newman安装（Linux环境） 6](#_Toc38443656)

[6 测试用例脚本执行 6](#_Toc38443657)

# 文档约定

本文档遵循以下约定：

1. 表头文字使用了20%灰度背景。
2. 插图一律“嵌入”于描述正文中，而非“浮于文字上方”。
3. 用同号、同体但加粗的文字来强调需要读者重视的内容。
4. Linux命令，用consolas字体+紫色显示。

# 术语、定义和缩略语

## 术语、定义

本文使用的专用术语、定义见表2.1。

表2.1

| 术语/定义 | 含 义 |
| --- | --- |
| postman | 开源的软件自动化测试工具 |
| node.js | *Node*.*js* 是一个基于 Chrome V8 引擎的 JavaScript 运行环境 |

## 缩略语

本文使用的专用缩略语见表2.2。

表2.2

| 缩略语 | 英文原文 | 中文含义 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

# 相关文档

本文涉及的相关文档见表3.1。

表3.1

| 文件编号 | 文件名称 | 版本号 | 说明 |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 概述

本文详细阐述了如何使用newman在Linux机上对Bplus服务进行CT测试，详细介绍了环境的搭建、用例的执行。

# 环境搭建

### newman安装（Linux环境）

#### node.js安装

通过命令:

wget <https://nodejs.org/dist/v10.16.2/node-v10.16.2-linux-x64.tar.xz> （由于内网机器没有外网，该安装包暂时由我提供）下载nodejs安装包，具体版本根据自己需要下载。

通过命令tar -xvf node-v10.16.2-linux-x64.tar.xz 解压安装包，执行mv node-v10.16.2-linux-x64 node将解压后的文件重命名为node。

打开执行vim/etc/profile打开配置文件，配置环境变量，在该文件末尾添加：

export NODE\_HOME=/usr/local/node（该处为node的地址）  
export PATH=$NODE\_HOME/bin:$PATH

执行source /etc/profile使环境变量生效，此时nodejs安装成功，可通过node –v查看node的版本，通过npm –v查看npm版本

#### newman安装

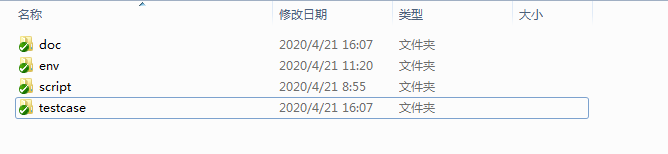
进入到newman目录下并执行npm install -g newman安装newman，待安装完成后，通过newman –v查看newman版本以确定newman安装成功。

# 测试用例脚本执行

### 获取测试用例

目前Bplus所有服务的测试用例都放置在git地址：

https://git.sinosun.com.cn/product/bplus/mall/backend/main下的test/testcase目录，该仓库用例存放的结构如下：



其中doc中为测试相关文档，env中为测试用例的环境变量，script中放置的是执行用例的脚本，testcase文件夹中放置的是所有服务的测试用例。

### 运行脚本执行用例

通过执行目录script下的脚本，可全量执行testcase目录下的全部用例文件，或者指定具体服务的测试用例文件。进入script目录下，执行命令：

python bplusTest.py -c param1 - r param2 –s param3 –t param4 -e param5

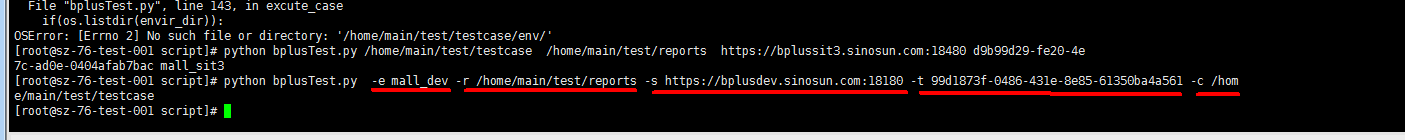
其中，

-c 参数：为用例路径，可以是用例目录，也可以指定到单个用例文件路径（结尾不带“/”）;  
-r 参数：为测试报告路径（结尾不带“/”）;  
-s 参数：为访问的地址（结尾不带“/”）;  
-t 参数：为获取token的密钥，例如开发环境：99d1873f-0486-431e-8e85-61350ba4 a561，sit2环境：a1a84de7-2859-4198-a90f-f19422b007a4，sit3环境：d9b 99d29-fe20-4e7c-ad0e-0404afab7bac；  
-e 参数：为商城的环境, 例如mall\_dev，mall\_sit2，mall\_sit3。如果不填，默认为开发环 境mall\_dev。

举例说明： 从git拉下来仓库目录为 ./main，报告路径指定为./main/test/reports，环境地址为：https://bplusdev.sinosun.com:18180，client\_secret和mall\_type为开发环境

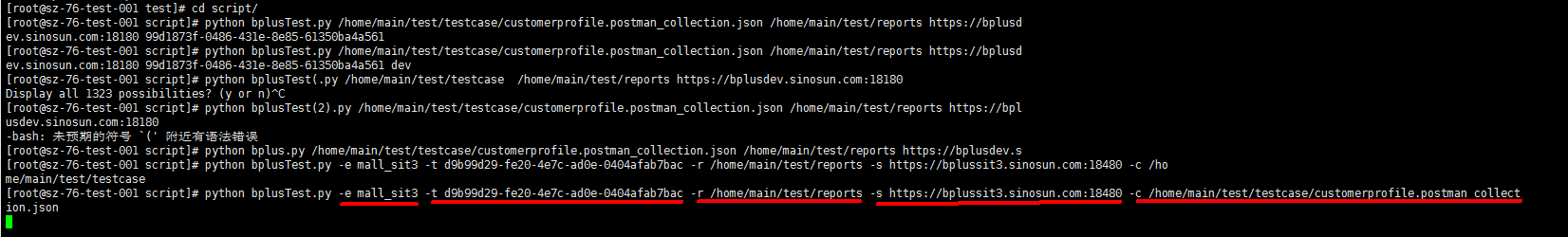
全量执行测试用例命令：

python bplusTest.py -c ./main/test/testcase -r ./main/test/reports -s <https://bplusdev.sinosun.com:18180> -t 99d1873f-0486-431e-8e85-61350ba4a561 -e mall\_dev



单独执行某个服务的测试用例文件命令：

python bplusTest.py –c ./home/main/test/testcase/xxx.json –r ./main/test/reports -s <https://bplusdev.sinosun.com:18180> -t 99d1873f-0486-431e-8e85-61350ba4a561 -e mall\_dev



命令执行完成后，在main/test/reports目录下生成xml格式的测试报告，以及日志文件。

