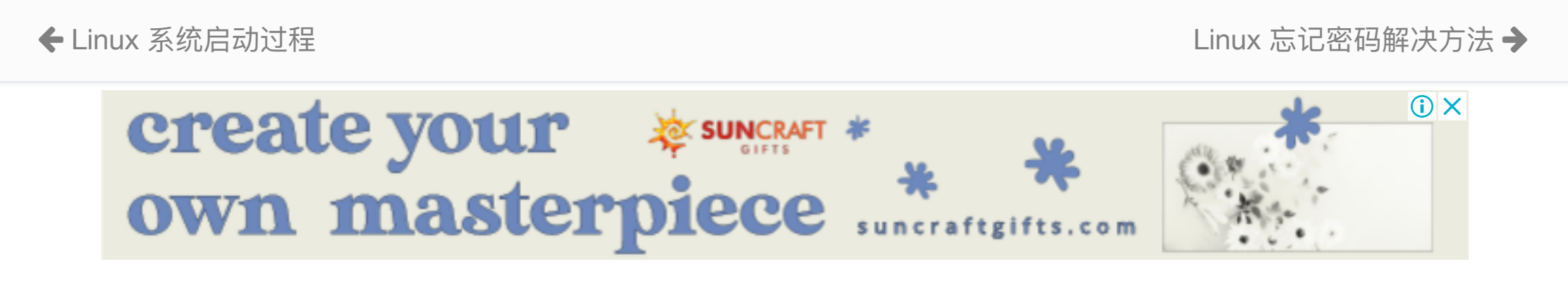


| |
|----------------|
| Linux 教程 |
| Linux 简介 |
| Linux 安装 |
| Linux 云服务器 |
| Linux 系统启动过程 |
| Linux 系统目录结构 |
| Linux 忘记密码解决方法 |
| Linux 远程登录 |
| Linux 文件基本属性 |
| Linux 文件与目录管理 |
| Linux 用户和用户组管理 |
| Linux 磁盘管理 |
| Linux vi/vim |
| linux yum 命令 |
| Linux apt 命令 |
| Shell 教程 |
| Shell 教程 |
| Shell 变量 |
| Shell 传递参数 |
| Shell 数组 |
| Shell 运算符 |
| Shell echo命令 |
| Shell printf命令 |
| Shell test 命令 |
| Shell 流程控制 |
| Shell 函数 |
| Shell 输入/输出重定向 |
| Shell 文件包含 |
| Linux 参考手册 |
| Linux 命令大全 |
| Nginx 安装配置 |
| MySQL 安装配置 |



Linux 系统目录结构

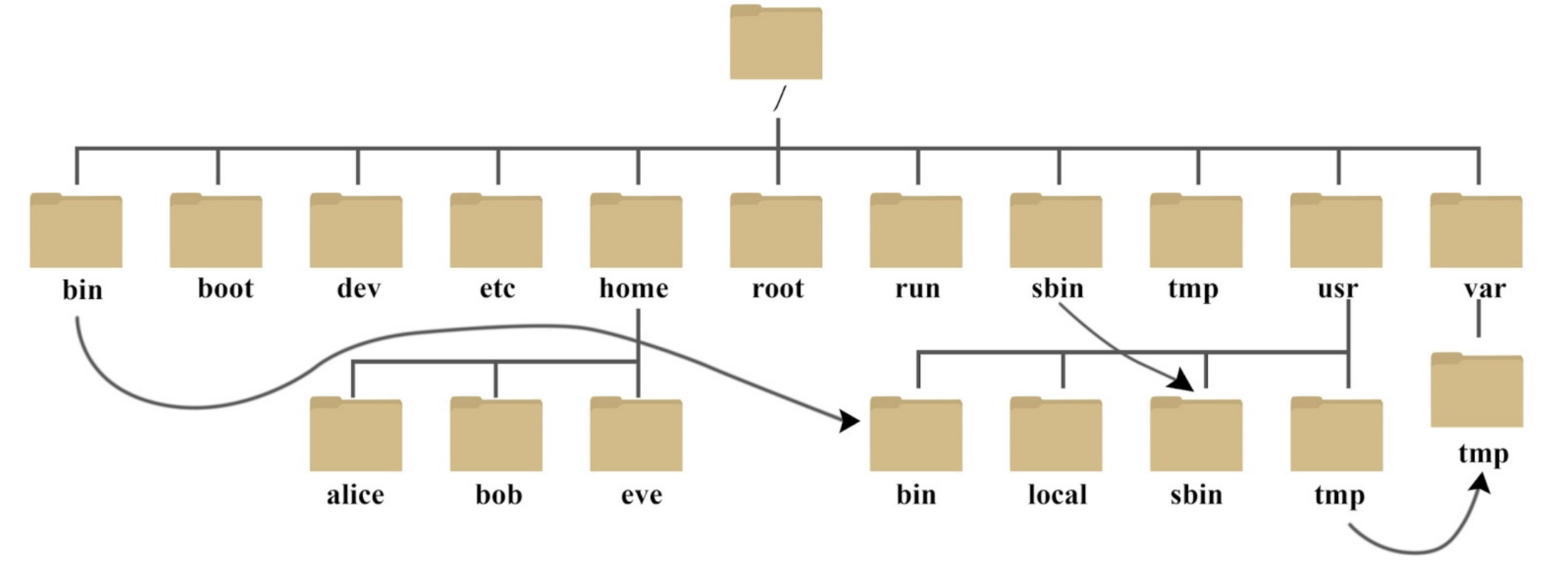
登录系统后，在当前命令窗口下输入命令：

```
ls /
```

你会看到如下图所示：

```
[root@localhost ~]# ls /
bin    dev    home  lost+found  mnt    proc  sbin    srv    tmp    var
boot  etc    lib   media      opt    root  selinux sys   usr
```

树状目录结构：

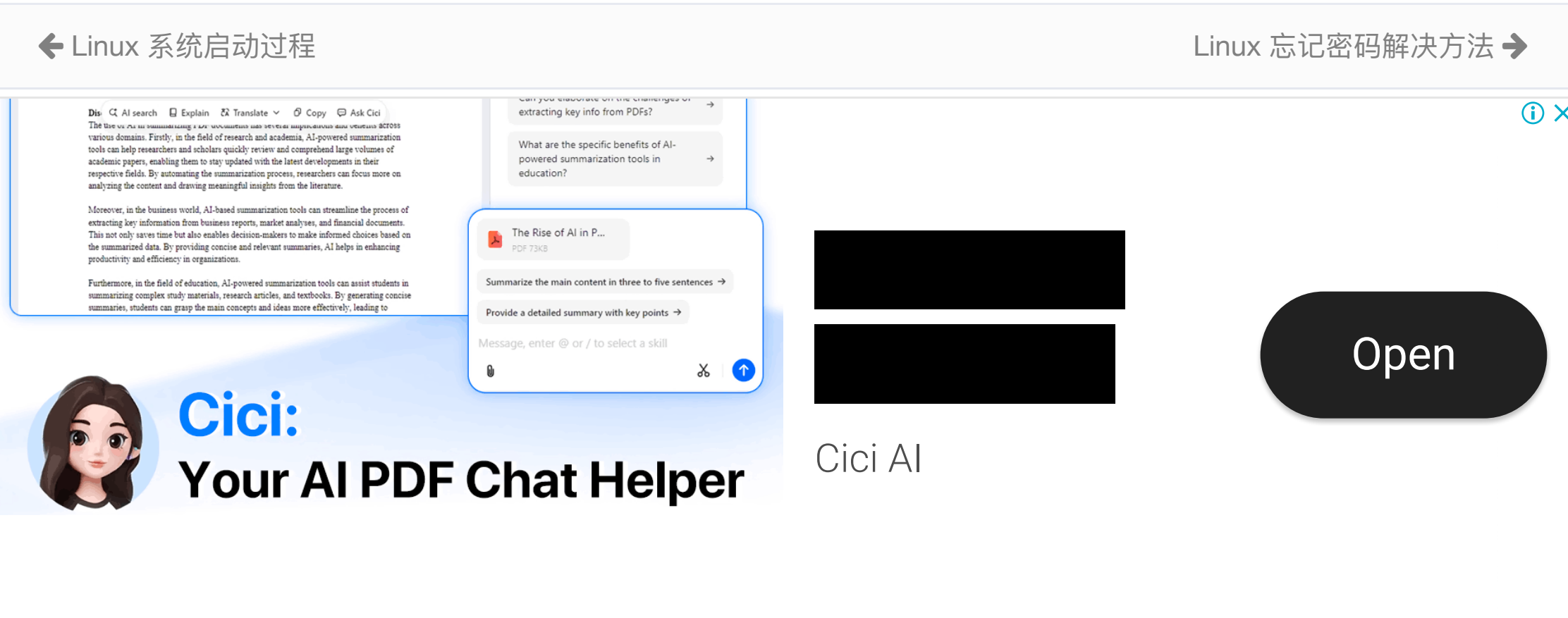


以下是对这些目录的解释：

- **/bin:**
bin 是 Binaries (二进制文件) 的缩写, 这个目录存放着最经常使用的命令。
- **/boot:**
这里存放的是启动 Linux 时使用的一些核心文件, 包括一些连接文件以及镜像文件。
- **/dev :**
dev 是 Device(设备) 的缩写, 该目录下存放的是 Linux 的外部设备, 在 Linux 中访问设备的方式和访问文件的方式是相同的。
- **/etc:**
etc 是 Etcetera(等等) 的缩写,这个目录用来存放所有的系统管理所需要的配置文件和子目录。
- **/home:**
用户的主目录, 在 Linux 中, 每个用户都有一个自己的目录, 一般该目录名是以用户的账号命名的, 如上图中的 alice、bob 和 eve。
- **/lib:**
lib 是 Library(库) 的缩写这个目录里存放着系统最基本的动态连接共享库, 其作用类似于 Windows 里的 DLL 文件。几乎所有的应用程序都需要用到这些共享库。
- **/lost+found:**
这个目录一般情况下是空的, 当系统非法关机后, 这里就存放了一些文件。
- **/media:**
linux 系统会自动识别一些设备, 例如U盘、光驱等等, 当识别后, Linux 会把识别的设备挂载到这个目录下。
- **/mnt:**
系统提供该目录是为了让用户临时挂载别的文件系统的, 我们可以将光驱挂载在 /mnt/ 上, 然后进入该目录就可以查看光驱里的内容了。
- **/opt:**
opt 是 optional(可选) 的缩写, 这是给主机额外安装软件所摆放的目录。比如你安装一个ORACLE数据库则就可以放到这个目录下。默认是空的。
- **/proc:**
proc 是 Processes(进程) 的缩写, /proc 是一种伪文件系统 (也即虚拟文件系统) , 存储的是当前内核运行状态的一系列特殊文件, 这个目录是一个虚拟的目录, 它是系统内存的映射, 我们可以通过直接访问这个目录来获取系统信息。
这个目录的内容不在硬盘上而是在内存里, 我们也可以直接修改里面的某些文件, 比如可以通过下面的命令来屏蔽主机的ping命令, 使别人无法ping你的机器:

```
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all
```
- **/root:**
该目录为系统管理员, 也称作超级权限者的用户主目录。
- **/sbin:**
s 就是 Super User 的意思, 是 Superuser Binaries (超级用户的二进制文件) 的缩写, 这里存放的是系统管理员使用的系统管理程序。
- **/selinux:**
这个目录是 Redhat/CentOS 所特有的目录, Selinux 是一个安全机制, 类似于 windows 的防火墙, 但是这套机制比较复杂, 这个目录就是存放selinux相关的文件的。
- **/srv:**
该目录存放一些服务启动之后需要提取的数据。
- **/sys:**
这是 Linux2.6 内核的一个很大的变化。该目录下安装了 2.6 内核中新出现的一个文件系统 sysfs 。
sysfs 文件系统集成了下面3种文件系统的信息: 针对进程信息的 proc 文件系统、针对设备的 devfs 文件系统以及针对伪终端的 devpts 文件系统。
该文件系统是内核设备树的一个直观反映。
当一个内核对象被创建的时候, 对应的文件和目录也在内核对象子系统中被创建。
- **/tmp:**
tmp 是 temporary(临时) 的缩写这个目录是用来存放一些临时文件的。
- **/usr:**
usr 是 unix system resources(unix 系统资源) 的缩写, 这是一个非常重要的目录, 用户的很多应用程序和文件都放在这个目录下, 类似于 windows 下的 program files 目录。
- **/usr/bin:**
系统用户使用的应用程序。
- **/usr/sbin:**
超级用户使用的比较高级的管理程序和系统守护程序。
- **/usr/src:**
内核源代码默认的放置目录。
- **/var:**
var 是 variable(变量) 的缩写, 这个目录中存放着在不断扩充着的东西, 我们习惯将那些经常被修改的目录放在这个目录下。包括各种日志文件。
- **/run:**
是一个临时文件系统, 存储系统启动以来的信息。当系统重启时, 这个目录下的文件应该被删掉或清除。如果你的系统上有 /va
r/run 目录, 应该让它指向 run。

在 Linux 系统中, 有几个目录是比较重要的, 平时需要注意不要误删除或者随意更改内部文件。
/etc: 上边也提到了, 这个是系统中的配置文件, 如果你更改了该目录下的某个文件可能会导致系统不能启动。
/bin, /sbin, /usr/bin, /usr/sbin: 这是系统预设的执行文件的放置目录, 比如 **ls** 就是在 **/bin/ls** 目录下的。
值得提出的是 **/bin, /usr/bin** 是给系统用户使用的指令 (除 root 外的通用用户) , 而 **/sbin, /usr/sbin** 则是给 root 使用的指令。
/var: 这是一个非常重要的目录, 系统上跑了很多程序, 那么每个程序都会有相应的日志产生, 而这些日志就被记录到这个目录下, 具体在 **/var/log** 目录下, 另外 mail 的预设放置也是在这里。



2 篇笔记

在 Linux 或 Unix 操作系统中, 所有的文件和目录都被组织成一个根节点开始的倒置的树状结构。
文件系统的最顶层是由根目录开始的, 系统使用 / 来表示根目录。在根目录之下的既可以是目录, 也可以是文件, 而每一个目录中又可以包含子目录文件。如此反复就可以构成一个庞大的文件系统。
在Linux文件系统中有两个特殊的目录, 一个用户所在的工作目录, 也叫当前目录, 可以使用一个点 . 来表示; 另一个是当前目录的上一级目录, 也叫父目录, 可以使用两个点 .. 来表示。

- . : 代表当前的目录, 也可以使用 / 来表示;
- .. : 代表上一层目录, 也可以 ./ 来代表。

如果一个目录或文件名以一个点 . 开始, 表示这个目录或文件是一个隐藏目录或文件(如: .bashrc)。即以默认方式查找时, 不显示该目录或文件。

黄太狼 8年前 (2017-03-26)

系统启动必须：

- **/boot:** 存放的启动Linux 时使用的内核文件, 包括连接文件以及镜像文件。
- **/etc:** 存放所有的系统需要的配置文件和子目录列表, 更改目录下的文件可能会导致系统不能启动。

- **/lib:** 存放基本代码库 (比如c++库) , 其作用类似于Windows里的DLL文件。几乎所有的应用程序都需要用到这些共享库。
- **/sys:** 这是linux2.6内核的一个很大的变化。该目录下安装了2.6内核中新出现的一个文件系统 sysfs 。sysfs文件系统集成了下面3种文件系统的信息: 针对进程信息的proc文件系统、针对设备的devfs文件系统以及针对伪终端的devpts文件系统。该文件系统是内核设备树的一个直观反映。当一个内核对象被创建的时候, 对应的文件和目录也在内核对象子系统中

指令集合：

- **/bin:** 存放着最常用的程序和指令
- **/sbin:** 只有系统管理员能使用的程序和指令。

外部文件管理：

- **/dev :** Device(设备)的缩写, 存放的是Linux的外部设备。注意：在Linux中访问设备和访问文件的方式是相同的。
 - **/media:** 类windows的其他设备, 例如U盘、光驱等等, 识别后linux会把设备放到这个目录下。
 - **/mnt:** 临时挂载别的文件系统的, 我们可以将光驱挂载在 /mnt/ 上, 然后进入该目录就可以查看光驱里的内容了。
- 临时文件：
- **/run:** 是一个临时文件系统, 存储系统启动以来的信息。当系统重启时, 这个目录下的文件应该被删掉或清除。如果你的系统上有 /var/run 目录, 应该让它指向 run。
 - **/lost+found:** 一般情况下为空的, 系统非法关机后, 这里就存放一些文件。
 - **/tmp:** 这个目录是用来存放一些临时文件的。

账户：

- **/root:** 系统管理员的用户主目录。
- **/home:** 用户的主目录, 以用户的账号命名的。
- **/usr:** 用户的很多应用程序和文件都放在这个目录下, 类似于windows下的program files目录。
- **/usr/bin:** 系统用户使用的应用程序与指令。
- **/usr/sbin:** 超级用户使用的比较高级的管理程序和系统守护程序。
- **/usr/src:** 内核源代码默认的放置目录。

运行过程中要用：

- **/var:** 存放经常修改的数据, 比如程序运行的日志文件 (/var/log 目录下) 。

- **/proc:** 管理内存空间！虚拟的目录, 是系统内存的映射, 我们可以直接访问这个目录来, 获取系统信息。这个目录的内容不在硬盘上而是在内存里, 我们也可以直接修改里面的某些文件来做修改。

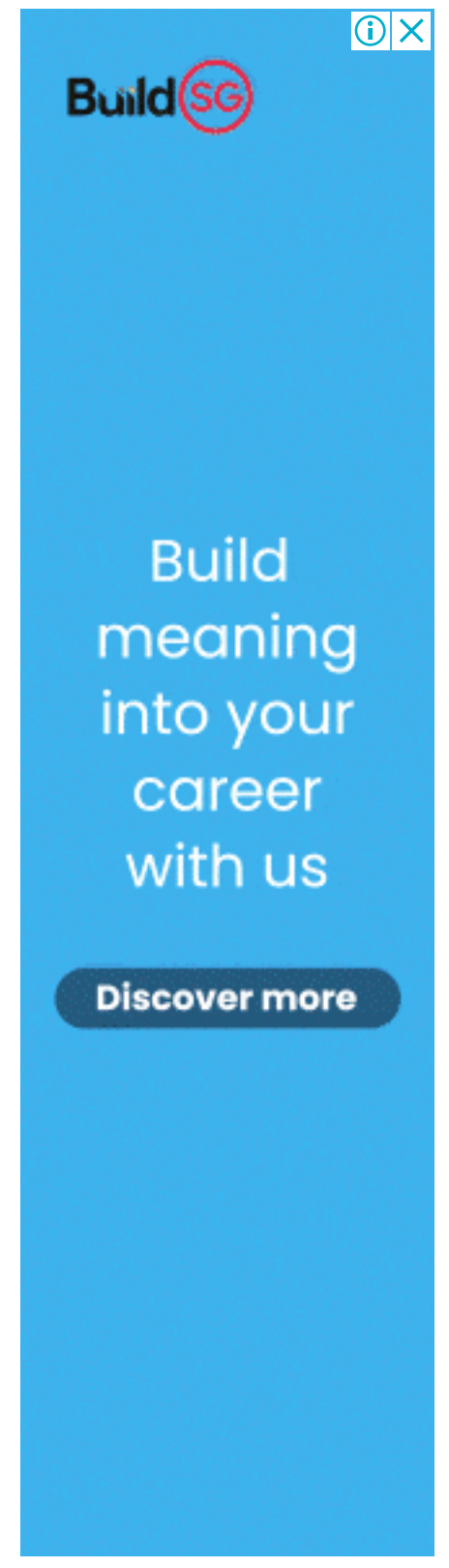
扩展用的：

- **/opt:** 默认是空的, 我们安装额外软件可以放在这个里面。
- **/srv:** 存放服务启动后需要提取的数据 (不用服务器就是空)

blockpunk 6年前 (2018-12-10)

| |
|-------------|
| 分类导航 |
| HTML / CSS |
| JavaScript |
| 服务端 |
| 数据库 |
| 数据分析 |
| 移动端 |
| XML 教程 |
| ASP.NET |
| Web Service |
| 开发工具 |
| 网站建设 |

Advertisement



- HTML 实例
- CSS 实例
- JavaScript 实例
- Ajax 实例
- jQuery 实例
- XML 实例
- Java 实例

- HTML 字符集设置
- HTML ASCII 字符集
- JS 混淆/加密
- PNG/JPEG 图片压缩
- HTML 拾色器
- JSON 格式化工具
- 随机数生成器

- HTML5 Web Index...
- docker compose ...
- docker compose ...
- docker compose ...
- docker compose ...
- docker compose ...
- docker compose ...

- 意见反馈
- 免责声明
- 关于我们
- 文章归档

