

有心故我在

HOME

CONTACT

GALLERY



理解Linux系统/etc/init.d目录和/etc/rc.local脚本

2014-09-30 15:27 by youxin, 426 阅读, 0 评论, 收藏, 编辑

本文英语版本来自：<http://www.ghacks.net/2009/04/04/get-to-know-linux-the-etcinitd-directory/>

以下内容是作者自己的翻译版本，如需转载到CSDN外其他网站，请注明本文链接。

一、关于/etc/init.d

如果你使用过linux系统，那么你一定听说过init.d目录。这个目录到底是干嘛的呢？它归根结底只做了一件事情，但这件事情非同小可，是为整个系统做的，因此它非常重要。**init.d目录包含许多系统各种服务的启动和停止脚本**。它控制着所有从acpid到x11-common的各种事务。当然，init.d远远没有这么简单。（译者注：acpid是linux操作系统新型电源管理标准；X11也叫做X Window系统，X Window系统(X11或X)是一种位图显示的视窗系统。它是在 Unix 和 类Unix **操作系统**，以及 OpenVMS 上建立**图形用户界面** 的标准工具包和协议，并可用于几乎已有的**现代操作系统**）。

当你查看/etc目录时，你会发现许多**rc#.d**形式存在的目录（这里#代表一个指定的初始化级别，范围是0~6）。在这些目录之下，包含了许多对进程进行控制的脚本。这些脚本要么以"K"开头，要么以"S"开头。以K开头的脚本运行在以S开头的脚本之前。这些脚本放置的地方，将决定这些脚本什么时候开始运行。在这些目录之间，系统服务一起合作，就像运行状况良好的机器一样。然而，有时候你希望能在不使用kill或killall命令的情况下，能干净的启动或杀死一个进程。这就是/etc/init.d能够派上用场的地方了！

如果你在使用Fedora系统，你可以找到这个目录：**/etc/rc.d/init.d**。实际上无论**init.d**放在什么地方，它都发挥着相同的作用。

为了能够使用**init.d**目录下的脚本，你需要有**root**权限或**sudo**权限。每个脚本都将被作为一个命令运行，该命令的结构大致如下所示：

/etc/init.d/command 选项

comand是实际运行的命令，选项可以有如下几种：

- ↪ start
- ↪ stop
- ↪ reload
- ↪ restart
- ↪ force-reload

大多数的情况下，你会使用start,stop,restart选项。例如，如果你想关闭网络，你可以使用如下形式的命令：

/etc/init.d/networking stop

又比如，你改变了网络设置，并且需要重启网络。你可以使用如下命令：

/etc/init.d/networking restart

init.d目录下常用初始化脚本有：

- ↪ networking
- ↪ samba
- ↪ apache2



SEARCH

最新评论

💬 Re:java中序列化之子类继承父类序列化

↳ 基于JDK1.8 只要有一个实现了Serializable就可以了 -- 平林新袖

💬 Re:python计算程序运行时间

↳ 感谢楼主，如果能再详细点就更好了。 -- xiaopo

💬 Re:layerX offsetX pageX

↳ pageX与滚动条移动有关，而clientX与滚动条移动无关 -- ITBoy-Men

💬 Re:C++抽象类的纯虚函数

↳ "f1()"是一个普通的重载."同名同参不应该是把基类中的f1()隐藏了吗？ -- C_hp

💬 Re:jquery获取自身元素的html

↳ 学习了。你写了好多博客啊。 -- 芝麻的西瓜

随笔分类

📁 ACM/ICPC(6)

📁 android 安卓(6)

📁 Apache/Nginx服务器(6)

📁 asp.net(13)

📁 Assemble汇编(25)

📁 C(56)

📁 C#(1)

📁 C++(202)

📁 c++_boost(19)

📁 c++---STL(9)

📁 Design Pattern & SA(66)

📁 Dos(4)

📁 English(2)

📁 fp_haskell(10)

📁 fp_Lisp(scheme)(47)

📁 go(2)

📁 google api(9)

📁 J2EE(37)

📁 Java(141)

📁 Linux(176)

📁 Linux网络编程(57)

📁 Lua(19)

📁 matlab(2)

📁 message queue(9)

📁 python(118)

📁 ruby(10)

📁 searchEngine(1)

📁 vim(3)

📁 visual studio(24)

📁 web design(25)

📁 web service(1)

📁 web---css(75)

📁 web---css---layout(17)

📁 web---css---sass&less(3)

📁 web---Dom(7)

C++

🔖 effective c++

🔖 Fred: C++ Notes:

🔖 Tutorials on Selected C/C++/Linux/UNIX Topics

🔖 wanging scheme

JAVA

🔖 chinaitlab 设计模式

🔖 dofactory.com/Patterns/Patterns.a

🔖 http://c2.com/cgi/wiki?CategoryPattern

🔖 ibm多线程

🔖 java glossary

🔖 Java Notes

🔖 javadocexamples

🔖 JavaGossip

🔖 java并发编程

🔖 oracle java specification

🔖 oracle java tutorial

🔖 sourcemaking

🔖 sourcemaking.com/design_pattern

🔖 多线程编程博客

🔖 解道设计模式

🔖 某人笔记2

🔖 某人设计模式笔记

🔖 设计模式各种语言实现

🔖 设计模式刘伟

🔖 设计原则概述

PHP

🔖 php某人博客

PYTHON

🔖 python 魔术方法

🔖 python文档2

🔖 pytho极好的文档

- [↗ ftpd](#)
- [↗ sshd](#)
- [↗ dovecot](#)
- [↗ mysql](#)

当然，你可能有其他更多常用的脚本，这个取决于你安装了什么linux操作系统。

二、关于/etc/rc.local

rc.local也是我经常使用的一个脚本。该脚本是在系统初始化级别脚本运行之后再执行的，因此可以安全地在里面添加你想在系统启动之后执行的脚本。常见的情况是你可以再里面添加nfs挂载/mount脚本。此外，你也可以在里面添加一些调试用的脚本命令。例如，我就碰到过这种情况：samba服务总是无法正常运行，而检查发现，samba是在系统启动过程中就该启动执行的，也就是说，samba守护程序配置保证了这种功能本应该正确执行。碰到这种类似情况，一般我也懒得花大量时间去查为什么，我只需要简单的在/etc/rc.local脚本里加上这么一行：

```
/etc/init.d/samba start
```

这样就成功的解决了samba服务异常的问题。

三、总结

Linux是灵活的。正因为它的灵活性，我们总是可以找到许多不同的办法来解决同一个问题。启动系统服务的例子就是一个很好的佐证。有了/etc/init.d目录下的脚本，再加上/etc/rc.local这个利器，你可以放心的确保你的服务可以完美的启动和运行。

转自：<http://blog.csdn.net/acs713/article/details/7322082>

另一篇：

本文包括3部分内容

- 1、Linux的引导过程
- 2、运行级别
- 3、/etc/rc.d/ 与/etc/rc.d/init.d的关系

都仅限于自身的理解，如有差错和不足的地方请指正和补充！一起学习，一起进步。

“/etc/rc.d/init.d/目录下的脚本就类似与windows中的注册表，在系统启动的时候某些指定脚本将被执行”。开始之前，先引用李善明经理昨天晚上总结时的一个理解，让大家先对init.d目录有个大概的印象。在进入init.d之前，我们一起来做两个准备工作，linux的引导过程和运行级别的概念。

一、Linux的引导过程

系统启动之后，在进入init.d之前，我们先来看看系统都做了什么工作，先看看一个图(此图来自网络，后期补上)

从这个图中，我们从比较高的角度去看开始引导的整个过程，比较清晰明了。系统加电之后，首先进行的硬件自检，然后是bootloader对系统的初始化，加载内核。

内核被加载到内存中之后，就开始执行了。一旦内核启动运行，对硬件的检测就会决定需要对哪些设备驱动程序进行初始化。从这里开始，内核就能够挂装根文件系统（这个过程类似于Windows识别并存取C盘的过程）。内核挂装了根文件系统，并已初始化所有的设备驱动程序和数据结构等之后，就通过启动一个叫init的用户级程序，完成引导进程。

二、运行级别(run level)

[web---Html5 + CSS3\(5\)](#)

[web---JavaScript\(180\)](#)

[web---JavaScript---bootstrap\(11\)](#)

[web---JavaScript---jQuery\(63\)](#)

[web---node.js\(22\)](#)

[web---php&Mysql\(219\)](#)

[web---php框架\(99\)](#)

[win32 sdk 和 MFC \(41\)](#)

[wordpress\(2\)](#)

[WPF\(1\)](#)

[xmpp&openfire\(5\)](#)

[编程心得经验\(14\)](#)

[编程之美和编程珠玑\(2\)](#)

[编译原理\(8\)](#)

[开发（并行，多线程）\(14\)](#)

[操作系统\(15\)](#)

[创业\(9\)](#)

[计算机网络\(5\)](#)

[计算机硬件\(4\)](#)

[技术大牛\(14\)](#)

[面试题\(20\)](#)

[软件\(18\)](#)

[软件工程\(45\)](#)

[收藏\(12\)](#)

[手机\(2\)](#)

[数据处理（含data mining\)\(20\)](#)

[数据结构\(22\)](#)

[数据结构--并查集\(1\)](#)

[数据结构---后缀数组\(1\)](#)

[数据库\(35\)](#)

[算法\(7\)](#)

[算法---大数+数论\(55\)](#)

[算法导论\(1\)](#)

[算法---递归\(13\)](#)

[算法---迭代法（辗转法）\(2\)](#)

[算法---动态规划\(8\)](#)

[算法---分治法\(6\)](#)

[算法--概率\(12\)](#)

[算法---哈希\(6\)](#)

[算法---回溯法\(10\)](#)

[算法---排序\(17\)](#)

[算法--时间复杂度\(2\)](#)

[算法---树和图 \(16\)](#)

[算法---随机算法\(3\)](#)

[算法---贪心法\(5\)](#)

[算法---字符串\(21\)](#)

[消息队列\(4\)](#)

[行业前景\(7\)](#)

[阅读学习\(23\)](#)

推荐排行榜

[1. html body width height 100%使用\(6\)](#)

[2. php单引号和双引号的区别与用法\(5\)](#)

[3. 判断栈和堆的生长方向\(4\)](#)

[stackoverflowPython问题精选](#)

WEB

[javascriptkit.com](#)

[webmonkey.com/](#)

[www.dzone.com](#)

[cssdrive.com](#)

[dynamicdrive](#)

[google code tutorial](#)

[http://ajaxian.com/](#)

[http://fc-lamp.blog.163.com/](#)

[javascript秘密花园](#)

[pixel2life.com](#)

[Practical PHP Programming](#)

[www.script-tutorials.com/](#)

Init进程是系统启动之后的第一个用户进程，所以它的pid(进程编号)始终为1。init进程上来首先做的事是去读取/etc/目录下inittab文件中initdefault id值，这个值称为运行级别(run-level)。它决定了系统启动之后运行于什么级别。运行级别决定了系统启动的绝大部分行为和目的。这个级别从0到6，具有不同的功能。不同的运行级定义如下：

```
# 0 - 停机（千万别把initdefault设置为0，否则系统永远无法启动）
# 1 - 单用户模式
# 2 - 多用户，没有 NFS
# 3 - 完全多用户模式(标准的运行级)
# 4 - 系统保留的
# 5 - X11（x window）
# 6 - 重新启动（千万不要把initdefault 设置为6，否则将一直在重启）
```

这是两个表较常用运行级别(后期补上),左图redhat9 级别3启动的将是文本界面，右图fc7级别5启动的将是图形界面。

三、/etc/rc.d/与/etc/rc.d/init.d的关系

写到这里，应该差不多要进入init.d了，可是我觉得单写/etc/rc.d/init.d的话不一定能说得清楚明白，就拿它跟/etc/rc.d这个它上一级的目录一起来讨论，可能比较合适一些，因为他们之间有着千丝万缕的关系。

在这里先解释一下init.d里面放的都是什么东西。这个目录存放的是一些脚本，一般是linux以rpm包安装时设定的一些服务的启动脚本。系统在安装时装了好多rpm包，这里面就有很多对应的脚本。执行这些脚本可以用来启动，停止，重启这些服务。

前面说到，/etc/rc.d/init.d这个目录下的脚本就类似与windows中的注册表，在系统启动的时候执行。程序运行到这里(init进程读取了运行级别)，相信从命名的角度大家也能猜到该运行/etc/rc.d/init.d里面的脚本了，不然它为什么也叫init(d)呢是吧。没错，是该运行init.d里的脚本了，但是并不是直接运行，而是有选择的因为系统并不需要启动所有的服务。

那么，系统是如何选择哪些需要启动哪些不要呢？这时刚才说的运行级别就起作用了。在决定了系统启动的run level之后，/etc/rc.d/rc这个脚本先执行。在RH9和FC7的源码中它都是一上来就check_runlevel()（虽然实现的代码不一样，也大同小异），知道了运行级别之后，对于每一个运行级别，在rc.d下都有一个子目录分别是rc0.d，rc1.d rc6.d。每个目录下都是到init.d目录的一部分脚本一些链接。每个级别要执行哪些服务就在相对应的目录下，比如级别5要启动的服务就都放在rc5.d下，但是放在这个rc5.d下的都是一些链接文件，链接到init.d中相对应的文件，真正干活到init.d里的脚本。

```
redhat9 ls
```

```
fc7 ls -l
```

这样看的就很清楚了。

到这里，估计大家可能都比较清楚了，我开始也以为是这样的。可是后来我仔细看过和比较这些链接文件和init.d里真正被执行的脚本的文件名之后，一直有几个问题没弄明白。借着写这个文章的机会，我做了一些功课，总算大概是解开了那些疑惑。

1、这些链接文件前面为什么会带一个Kxx或者Sxx呢？

是这样的，带K的表示停止(Kill)一个服务，S表示开启(Start)的意思

2、K和S后面带的数字呢？干什么用的

这个我开始的时候还以为排列起来好看或者数数用呢。后来发现不是的。它的作用是用来排序，就是决定这些脚本执行的顺序，数值小的先执行，数值大的后执行。很多时候这些执行顺序是很重要的，比如要启动Apache服务，就必须先配置网络接口，不然一个没有IP的机器来启动http服务那岂不是很搞笑。。。

3、无意中我发现同一个服务带S的和带K的链接到init.d之后是同一个脚本。我就纳闷了，为什么会是执行同一个脚本呢？

这个时候真是S和K的妙用了，原来S和K并不止是用来看起来分的清楚而

4. windows 查看文件被哪个进程占用 (3)

5. tcp 服务端如何判断客户端断开连接 (3)

阅读排行榜

1. python安装模块与包(34399)

2. mysql查找字符串出现位置(22950)

3. c++ vector 删除指定元素(22539)

4. Vim中如何全选并复制? (21762)

5. C++ string 比较(20750)

已。S给和K还分别给init.d下面的脚本传递了start和stop的参数。哦，是这样的(恍然大悟的样子,呵呵)！这时我才想起来原来曾经无数用过的/etc/rc.d/init.d/network restart命令。原来传S时相当于执行了/etc/rc.d/init.d/xxx start这条命令，当然K就相当于/etc/rc.d/init.d/xxx stop了

转自：<http://blog.csdn.net/r91987/article/details/7625411>

好文要顶

关注我

收藏该文

youxin

关注 - 31

粉丝 - 198

+加关注

0

推荐

0

反对

« 上一篇：[mongodb管理工具rockmongo](#)

» 下一篇：[ubuntu创建用户](#)

分类：[Linux](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

最新IT新闻：

- [因为一张黑板Word图画 微软向加纳老师捐赠电脑](#)
 - [运营商被曝即将迎来大调整：流量不分省内外](#)
 - [贾跃亭终爆仓 谁要进场接盘被割韭菜](#)
 - [OKEX公众号被封，火币网公众号被删名](#)
 - [周鸿祎委员：希望国家明确网民的数据所有权与使用权](#)
- » [更多新闻...](#)

最新知识库文章：

- [写给自学者的入门指南](#)
 - [和程序员谈恋爱](#)
 - [学会学习](#)
 - [优秀技术人的管理陷阱](#)
 - [作为一个程序员，数学对你到底有多重要](#)
- » [更多知识库文章...](#)