**将定制RPM包加入内部Yum Server**

2013-01-07 15:12:25

标签：[yum](http://blog.51cto.com/tag-yum.html) [定制](http://blog.51cto.com/tag-%E5%AE%9A%E5%88%B6.html) [rpm](http://blog.51cto.com/tag-rpm.html) [HAPAProxy](http://blog.51cto.com/tag-HAPAProxy.html)

原创作品，允许转载，转载时请务必以超链接形式标明文章 [原始出处](http://dreamway.blog.51cto.com/1281816/1110874) 、作者信息和本声明。否则将追究法律责任。<http://dreamway.blog.51cto.com/1281816/1110874>

#####################################################

##如有转载，请务必保留本文链接及版权信息

##欢迎广大运维同仁一起交流linux/unix网站运维技术!

##QQ:335623998

##E-mail:335623998@qq.com

##博客： <http://dreamway.blog.51cto.com/>

##weibo：<http://weibo.com/zhaixiangpan>

#####################################################

**如何定制RPM包**

<http://dreamway.blog.51cto.com/1281816/1110822>

前言：此文以生产案例应用通过HAProxy1.4.22源码包为例定制生成rpm包，并加入内部YUM Server仓库，便于以后的软件包安装

1、制作RPM包linux系统环境

1. # cat /etc/redhat-release
2. CentOS release 5.8 (Final)
3. # uname -r
4. 2.6.18-308.el5
5. # uname -m
6. x86\_64

2、构建前的准备

创建rpmbuild所需的目录结构，通常安装好的Cent0S5.8系统已经创建好目录结构，如下：

1. # ls /usr/src
2. debug kernels redhat
3. # ls /usr/src/redhat/
4. BUILD RPMS SOURCES SPECS SRPMS

root用户制作RPM包使用/usr/src/redhat/下的目录即可

如果使用普通用户，则手工创建好目录即可,命令如下

1. # mkdir -p  /home/zxp/{BUILD,RPMS,SOURCES,SPECS,SRPMS}

**目录说明**：

BUILD:   用于编译软件包时，源码的临时存放空间。  
RPMS:   制作好的二进制包的输出位置。  
SOURCES:  源码的位置。  
SPECS:   spec文件存放的位置。  
SRPMS:   制作好的源码RPM包的输出位置，该包安装时仍需要先编译。

3、SPEC文件详解

构建一个标准的 RPM 包，就需要在目录SPECS下创建.spec文件，里面包含即将被安装的软件的所有详细信息。然后对这个文本在系统中执行rpmbuild命令，系统会按照.spec内容设置的步骤自动生成最终的 RPM 包。

3.1 SPEC文件 语法详解（以HAProxy为例）

1. # cat haproxy.spec
2. Summary: HA-Proxy is a TCP/HTTP reverse proxy for high availability environments       #软件摘要信息
3. Name: haproxy       #软件名称
4. Version: 1.4.22     #软件版本
5. Release: 20130106\_hexun\_as5  #软件分支版本
6. License: GPL                 #软件版权
7. Group: System Environment/Daemons  #软件所属分类
8. URL: http://haproxy.1wt.eu/       #软件主页
9. Source0: http://haproxy.1wt.eu/download/1.4/src/%{name}-%{version}.tar.gz #源码位置
10. BuildRoot: %{\_tmppath}/%{name}-%{version}-root  #安装目录
11. BuildRequires: pcre-devel               #编译依赖软件包
12. Requires: /sbin/chkconfig, /sbin/service #安装依赖软件包
13. %description  #软件详细的描述信息
14. HA-Proxy is a TCP/HTTP reverse proxy which is particularly suited for high
15. availability environments. Indeed, it can:
16. - route HTTP requests depending on statically assigned cookies
17. - spread the load among several servers while assuring server persistence
18. through the use of HTTP cookies
19. - switch to backup servers in the event a main one fails
20. - accept connections to special ports dedicated to service monitoring
21. - stop accepting connections without breaking existing ones
22. - add/modify/delete HTTP headers both ways
23. - block requests matching a particular pattern
24. It needs very little resource. Its event-driven architecture allows it to easily
25. handle thousands of simultaneous connections on hundreds of instances without
26. risking the system's stability.
27. # %prep定义了构建前要做的准备，通常是%setup定义如何解包
28. %prep
29. %setup -q
30. # We don't want any perl dependecies in this RPM:
31. %define \_\_perl\_requires /bin/true #定义不使用perl依赖关系
32. %build #编译源码命令，通常是./configure && make，根据具体包安装方法而定
33. %{\_\_make} ARCH=%{\_target\_cpu} TARGET=linux26
34. %install  #安装阶段
35. [ "%{buildroot}" != "/" ] && %{\_\_rm} -rf %{buildroot}
36. %{\_\_install} -d %{buildroot}%{\_sbindir}
37. %{\_\_install} -d %{buildroot}%{\_sysconfdir}/rc.d/init.d
38. %{\_\_install} -d %{buildroot}%{\_sysconfdir}/%{name}
39. %{\_\_install} -d %{buildroot}%{\_mandir}/man1/
40. %{\_\_install} -s %{name} %{buildroot}%{\_sbindir}/
41. %{\_\_install} -c -m 644 examples/%{name}.cfg %{buildroot}%{\_sysconfdir}/%{name}/
42. %{\_\_install} -c -m 755 examples/%{name}.init %{buildroot}%{\_sysconfdir}/rc.d/init.d/%{name}
43. %{\_\_install} -c -m 755 doc/%{name}.1 %{buildroot}%{\_mandir}/man1/
44. %clean  #清理BUILD目录阶段
45. [ "%{buildroot}" != "/" ] && %{\_\_rm} -rf %{buildroot}
46. %post   #安装后制定的脚本
47. /sbin/chkconfig --add %{name}
48. %preun #卸载前执行的脚本
49. if [ $1 = 0 ]; then
50. /sbin/service %{name} stop >/dev/null 2>&1 || :
51. /sbin/chkconfig --del %{name}
52. fi
53. %postun  #卸载后执行的脚本
54. if [ "$1" -ge "1" ]; then
55. /sbin/service %{name} condrestart >/dev/null 2>&1 || :
56. fi
57. %files  #文件列表，主要是设置安装rpm包后的文件、目录属性
58. %defattr(-,root,root)  #定义默认属性
59. %doc CHANGELOG TODO examples/\*.cfg doc/haproxy-en.txt doc/haproxy-fr.txt doc/architecture.txt doc/configuration.txt #指定软件文档
60. %doc %{\_mandir}/man1/%{name}.1\*          #指定man帮助文档
61. %attr(0755,root,root) %{\_sbindir}/%{name}  #单独指定目录属性
62. %dir %{\_sysconfdir}/%{name}               #定义软件安装目录
63. %attr(0644,root,root) %config(noreplace) %{\_sysconfdir}/%{name}/%{name}.cfg   #单独指定文件属性
64. %attr(0755,root,root) %config %{\_sysconfdir}/rc.d/init.d/%{name} #单独指定文件属性
65. %changelog   #软件升级日志
66. \* Tue Aug 14 2012 Willy Tarreau <w@1wt.eu>
67. - updated to 1.4.22
68. …………略………………
69. \* Thu Oct 16 2003 Simon Matter <simon.matter@invoca.ch>
70. - initial build

3、获取spec文件方法(**三种方法**)

3.1 方法1：通过src.rpm包获取

下载&安装相应的src.rpm包

1. # cd SRPMS/
2. # wget http://1wt.eu/tools/haproxy/src/devel/haproxy-1.2.3-1.src.rpm
3. # rpm -ivh haproxy-1.2.3-1.src.rpm
4. 1:haproxy                ########################################### [100%]

 这里的“安装”是指把xxx.src.rpm中的tar.gz、patches、xxx.spec等文件分别输出到/usr/src/redhat/的SOURCES、SPECS等子目录中

查看生成的文件

1. # ls /usr/src/redhat/{SOURCES,SPECS}
2. /usr/src/redhat/SOURCES:
3. haproxy-1.2.3.tar.gz
4. /usr/src/redhat/SPECS:
5. haproxy.spec

找到spec文件，进行备份、修改

1. # cd /usr/src/redhat/SPECS
2. # cd /usr/src/redhat/SPECS
3. # cp haproxy.spec haproxy.spec\_bak$(date +%F)
4. # ls
5. haproxy.spec  haproxy.spec\_bak2013-01-06

3.2 方法2：新建spec文件

再目录SPECS下，创建一个新的spec文件，根据spec文件语法规范编写，这个难度较高，适用于自己开发的软件。

3.3 方法3：从源码包中获取已有spec文件 （推荐的方法）

1. # tar zxf haproxy-1.4.22.tar.gz
2. # cd haproxy-1.4.22
3. # ll haproxy.spec
4. -rw-rw-r-- 1 root root 7442 Aug 14 15:09 haproxy.spec
5. #cd haproxy-1.4.22/examples/
6. # ls
7. haproxy.spec

将spec放入目录SPECS

1. # cp haproxy.spec  /usr/src/redhat/SPECS/

备份

1. # cp haproxy.spec  haproxy.spec.$(date +%F)

本文采用此方法获取spec

4、编辑spec

1. # pwd /usr/src/redhat/SPECS
2. # vim haproxy.spec

修改前后内容对比（后面为原文件）

1. [root@study02 SPECS]# diff haproxy.spec haproxy.spec.2013-01-07
2. 4c4
3. < Release: 20130106\_hexun\_as5
4. ---
5. > Release: 1
6. 36c36
7. < %{\_\_make} ARCH=%{\_target\_cpu} TARGET=linux26
8. ---
9. > %{\_\_make} USE\_PCRE=1 DEBUG="" ARCH=%{\_target\_cpu} TARGET=linux26

5、下载HAProxy源码包

下载源码包并放入SOURCES目录中

1. # pwd
2. /usr/src/redhat/SOURCES
3. # wget http://haproxy.1wt.eu/download/1.4/src/haproxy-1.4.22.tar.gz
4. # ls haproxy-1.4.22.tar.gz

6、创建RPM 包

通过rpmbuild命令来解析SPEC文件生成对应的RPM包

6.1 确认rpmbuild 已经安装

1. # yum install rpm-build -y

6.2 安装依赖包

在build rpm包的系统上安装pcre-devel

1. # yum -y  install pcre-devel

6.3开始build rpm：

1. # rpmbuild  -v -ba SPECS/haproxy.spec

6.4 看到如下内容即创建过程，红色内容为build包位置

1. Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.44698
2. + umask 022
3. + cd /usr/src/redhat/BUILD
4. + LANG=C
5. + export LANG
6. + unset DISPLAY
7. + cd /usr/src/redhat/BUILD
8. + rm -rf haproxy-1.4.22
9. + /bin/gzip -dc /usr/src/redhat/SOURCES/haproxy-1.4.22.tar.gz
10. + tar -xf -
11. + STATUS=0
12. …………略………………
13. Wrote: /usr/src/redhat/SRPMS/haproxy-1.4.22-20130106\_hexun\_as5.src.rpm
14. Wrote: /usr/src/redhat/RPMS/x86\_64/haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
15. Wrote: /usr/src/redhat/RPMS/x86\_64/haproxy-debuginfo-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
16. Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.75459
17. + umask 022
18. + cd /usr/src/redhat/BUILD
19. + cd haproxy-1.4.22
20. + '[' /var/tmp/haproxy-1.4.22-root '!=' / ']'
21. + /bin/rm -rf /var/tmp/haproxy-1.4.22-root
22. + exit 0  #注意输出status为0 即创建过程错误

7、检验RPM包

7.1 使用 Mock 和 Koji 去测试 RPM 包

详见 [Fedora 新软件维护者指南](https://fedoraproject.org/wiki/Fedora_%E6%96%B0%E8%BD%AF%E4%BB%B6%E7%BB%B4%E6%8A%A4%E8%80%85%E6%8C%87%E5%8D%97?rd=Fedora%E9%8F%82%E6%9D%BF%EE%9A%9C%E6%B5%A0%E5%89%81%E6%B7%AE%E9%8E%B6%E3%82%88%E2%82%AC%E5%91%AE%E5%AF%9A%E9%8D%97?#.E4.BD.BF.E7.94.A8_Mock_.E5.92.8C_Koji_.E5.8E.BB.E6.B5.8B.E8.AF.95_RPM_.E5.8C.85)

7.2 列出RPM软件包内的文件信息

1. # rpm -qpl   /usr/src/redhat/RPMS/x86\_64/haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
2. /etc/haproxy
3. /etc/haproxy/haproxy.cfg
4. /etc/rc.d/init.d/haproxy
5. /usr/sbin/haproxy
6. /usr/share/doc/haproxy-1.4.22
7. /usr/share/doc/haproxy-1.4.22/CHANGELOG
8. /usr/share/doc/haproxy-1.4.22/TODO
9. /usr/share/doc/haproxy-1.4.22/acl-content-sw.cfg
10. /usr/share/doc/haproxy-1.4.22/architecture.txt
11. /usr/share/doc/haproxy-1.4.22/auth.cfg
12. /usr/share/doc/haproxy-1.4.22/build.cfg
13. /usr/share/doc/haproxy-1.4.22/configuration.txt
14. /usr/share/doc/haproxy-1.4.22/content-sw-sample.cfg
15. /usr/share/doc/haproxy-1.4.22/cttproxy-src.cfg
16. /usr/share/doc/haproxy-1.4.22/examples.cfg
17. /usr/share/doc/haproxy-1.4.22/haproxy-en.txt
18. /usr/share/doc/haproxy-1.4.22/haproxy-fr.txt
19. /usr/share/doc/haproxy-1.4.22/haproxy.cfg
20. /usr/share/doc/haproxy-1.4.22/option-http\_proxy.cfg
21. /usr/share/doc/haproxy-1.4.22/tarpit.cfg
22. /usr/share/doc/haproxy-1.4.22/test-section-kw.cfg
23. /usr/share/doc/haproxy-1.4.22/url-switching.cfg
24. /usr/share/man/man1/haproxy.1.gz

7.3 列出RPM软件包的描述信息

1. # rpm -qpi   /usr/src/redhat/RPMS/x86\_64/haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
2. Name        : haproxy                      Relocations: (not relocatable)
3. Version     : 1.4.22                            Vendor: (none)
4. Release     : 20130106\_hexun\_as5            Build Date: Mon 07 Jan 2013 09:36:21 AM CST
5. Install Date: (not installed)               Build Host: study02
6. Group       : System Environment/Daemons    Source RPM: haproxy-1.4.22-20130106\_hexun\_as5.src.rpm
7. Size        : 1275757                          License: GPL
8. Signature   : (none)
9. URL         : http://haproxy.1wt.eu/
10. Summary     : HA-Proxy is a TCP/HTTP reverse proxy for high availability environments
11. Description :
12. HA-Proxy is a TCP/HTTP reverse proxy which is particularly suited for high
13. availability environments. Indeed, it can:
14. - route HTTP requests depending on statically assigned cookies
15. - spread the load among several servers while assuring server persistence
16. through the use of HTTP cookies
17. - switch to backup servers in the event a main one fails
18. - accept connections to special ports dedicated to service monitoring
19. - stop accepting connections without breaking existing ones
20. - add/modify/delete HTTP headers both ways
21. - block requests matching a particular pattern
22. It needs very little resource. Its event-driven architecture allows it to easily
23. handle thousands of simultaneous connections on hundreds of instances without
24. risking the system's stability.

7.4 rpm方式安装验证

1. [root@study01 tools]# rpm -ivh haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
2. Preparing...                ########################################### [100%]
3. 1:haproxy                ########################################### [100%]
4. [root@study01 tools]# ls
5. haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
6. [root@study01 tools]# ls /etc/haproxy/ #安装目录及主配置文件
7. haproxy.cfg
8. [root@study01 tools]# ll /etc/rc.d/init.d/haproxy  #服务启动文件
9. -rwxr-xr-x 1 root root 2553 Jan 7 09:36 /etc/rc.d/init.d/haproxy
10. [root@study01 tools]# ll /etc/init.d/haproxy
11. -rwxr-xr-x 1 root root 2553 Jan 7 09:36 /etc/init.d/haproxy
12. [root@study01 tools]# rpm -qf /etc/init.d/haproxy  #查看软件所属软件包
13. haproxy-1.4.22-20130106\_hexun\_as5

至此定制RPM包已经制作完毕，接下来我们就可以把常用的应用软件源码包都制作为rpm包放入yum仓库，以后部署软件就很便捷了。

**8、将定制的RPM加入yum仓库**

所需工具createrepo 、yum-arch

详见很早以前写的yum server搭建文章

《CentOS基于 http服务 搭建yum 源 服务器》<http://bbs.linuxtone.org/thread-4439-1-1.html>

8.1 在yum server上配置

1. [root@hxinstall 5ASU8]# cd /opt/ftp/pub/os/Linux/Centos/x86\_64/Centos5.8/
2. [root@hxinstall Centos5.8]# ls
3. dvd install
4. # cd RPMS.hexun/
5. # ls
6. haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm

8.2 创建repodata 数据库

包含rpm 包的详细信息，依赖关系等

1. [root@hxinstall RPMS.hexun]# createrepo /opt/ftp/pub/os/Linux/Centos/x86\_64/Centos5.8/RPMS.hexun
2. 1/1 - haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm
3. Saving **Primary** metadata
4. Saving file lists metadata
5. Saving other metadata

8.3 RPM包分析

在目录下产生 heaers 目录

1. # yum-arch -l /opt/ftp/pub/os/Linux/Centos/x86\_64/Centos5.8/RPMS.hexun
2. # ls
3. haproxy-1.4.22-20130106\_hexun\_as5.x86\_64.rpm headers repodata

注意：

每添加新的软件包，都需要执行此两个命令使其生效。

8.4 yum 客户端安装定制版HAProxy

8.4.1 配置yum

添加定制软件包yum仓库

1. cat << EOF >>  /etc/yum.repos.d/CentOS-Base.repo
2. [hexun]
3. **name**= hexun rpm
4. baseurl=http://10.0.251.154/pub/os/Linux/Centos/x86\_64/Centos5.8/RPMS.hexun
5. gpgcheck=0
6. EOF

8.4.2 YUM安装自定制HAProxy 软件包

1. # yum clean all
2. # yum  install  haproxy -y
3. Loaded plugins: fastestmirror
4. Determining fastest mirrors
5. ……略…………
6. hexun                                                       |  951 B     00:00
7. hexun/**primary**                                               | 1.2 kB     00:00
8. hexun                                                                          1/1
9. Setting up Install Process
10. Resolving Dependencies
11. --> Running transaction check
12. ---> Package haproxy.x86\_64 0:1.4.22-20130106\_hexun\_as5 set to be updated
13. ………略………
14. Running **Transaction**
15. Installing     : haproxy                                                     1/1
16. Installed:  #yum已经安装成功
17. haproxy.x86\_64 0:1.4.22-20130106\_hexun\_as5

8.4.3 卸载定制版软件包

1. # yum  remove  haproxy   -y
2. Loaded plugins: fastestmirror
3. Setting up Remove Process
4. Resolving Dependencies
5. --> Running transaction check
6. ---> Package haproxy.x86\_64 0:1.4.22-20130106\_hexun\_as5 set to be erased
7. --> Finished Dependency Resolution
8. Dependencies Resolved
9. ===================================================================================
10. Package       Arch         Version                          Repository       **Size**
11. ===================================================================================
12. Removing:
13. haproxy       x86\_64       1.4.22-20130106\_hexun\_as5        installed       1.2 M
14. **Transaction** Summary
15. ===================================================================================
16. Remove        1 Package(s)
17. Reinstall     0 Package(s)
18. Downgrade     0 Package(s)
19. Downloading Packages:
20. Running rpm\_check\_debug
21. Running **Transaction** Test
22. Finished **Transaction** Test
23. **Transaction** Test Succeeded
24. Running **Transaction**
25. Erasing        : haproxy                                                     1/1
26. Removed:  #已经成功卸载
27. haproxy.x86\_64 0:1.4.22-20130106\_hexun\_as5
28. Complete!

9、制作RPM报错记录

9.1 build包报错找不到源码包

1. # rpmbuild -v -ba SPECS/haproxy.spec
2. error: File /usr/src/redhat/SOURCES/haproxy-1.4.22.tar.gz: **No** such file or directory

解决：将对应版本源码包放入对应目录

1. # cd /usr/src/redhat/SOURCES/
2. [root@study02 SOURCES]# wget http://haproxy.1wt.eu/download/1.4/src/haproxy-1.4.22.tar.gz

9.2 build时报依赖关系错误

1. # rpmbuild -v -ba SPECS/haproxy.spec
2. error: Failed build dependencies:
3. pcre-devel **is** needed **by** haproxy-1.4.22-20130106\_hexun\_as5.x86\_64

解决：在build rpm包的系统上安装pcre-devel

1. # yum -y install pcre-devel

10、制作RPM包涉及到的命令用法

10.1 命令rpmbuild

用法：

1. Usage: rpmbuild [ <specfile> | <tarball> | <source package> ]

常用参数：

-bb 只编译二进制rpm包

-bs 只编译源码rpm包(src.rpm)

-ba 同时编译二进制和源码rpm包(src.rpm)

-bp执行到%prep段，解开tar包然后把所有的补丁文件合并而生成一个完整的具最新功能的源文件

10.2 命令rpm

rpm 命令以前的linux版本较常用，但是解决软件包依赖关系就比较麻烦，不如yum便捷。

用法：

1. Usage: rpm [**OPTION**...]

常用的rpm参数组合：

1. （1）rpm -qx file\_name, x={f,i,l,a,p...}, file\_name可以是命令名、动态库名称、配置文件名等等。
2. 使用此命令的前提：相应的rpm包已经安装。
3. rpm -qf file：查询文件file是哪个rpm包中的；rpm -qf `which your\_cmd`, rpm -qf `locate file\_name`
4. rpm -qi rpm\_name：查看指定rpm包安装的详细信息；
5. rpm -ql installed\_rpm\_name：列出已经安装的rpm包中包含了哪些文件及他们的安装路径。如rpm -ql iptraf
6. 用以下选项与 -q 连用来指明要查询哪些软件包的信息。这些选项被称之为“软件包指定选项”：
7. -a 查询所有已安装的软件包。
8. -f <file> 将查询包含有文件 <file>的软件包。
9. -p <packagefile> 查询软件包文件名为 <packagefile>的包。
10. 有几种方式来指定查询软件包时所显示的信息。 以下选项可通过读取rpm包头部的办法显示rpm包的信息，这样的选项被称作“信息选择选项”：
11. -i 显示软件包信息，如描述、发行号、大小、编译日期、安装日期、硬件平台、以及其它一些各类信息。
12. -l 列出软件包中包含的文件。（列出已经安装的rpm包中包含了哪些文件及他们的安装路径）
13. -s 显示软件包中所有文件的状态。
14. -d 列出被标注为文档的文件 (如，man 手册、 info 信息、README，等等) 。
15. -c 列出被标注为配置文件的文件。这些文件是需要在安装完毕后加以定制的，如 (sendmail.cf, passwd, inittab, 等) 。
16. 如果要在执行上述选项的同时，显示文件列表, 可以同时使用 -v 命令行选项，就能得出与 ls -l 格式类似的输出。
17. （2）查看未安装的rpm/src.rpm包中包含的文件列表
18. － 在本地暂时只能使用（4）中提供的方法；
19. － 通过rpmfind.net等网站进行查询；
20. （3）rpm -ivh xxx.rpm：重新安装；（和-Uvh相比，建议用-ivh）
21. rpm -ivh --relocate /=/tmp/test/ xxx.rpm
22. （4）rpm2cpio xxx.rpm/xxx.src.rpm：将rpm解压为cpio归档；
23. rpm2cpio xxx.rpm/xxx.src.rpm | cpio -idmv (rpm2cpio xxx.rpm | cpio --extract --make-directories)
24. 参数-i(或--extract)表示提取文件； v表示指示执行进程；-d(或--make-directory)表示根据包中文件原来的路径建立目录；m表示保持文件的更新时间。

10.3 命令 yum

目前yum命令在安装软件包时很常用，这里不再一一列举用法参数。

常用命令可参考<http://bbs.linuxtone.org/thread-4439-1-1.html>

11、参考资料

RPM 打包技术与典型 SPEC 文件分析

<http://www.ibm.com/developerworks/cn/linux/l-rpm/?S_TACT=105AGX52&S_CMP=tech-51CTO>

常用的rpm参数组合

<http://www.xdays.info/rpm%E5%8C%85%E5%88%B6%E4%BD%9C.html#file>

RPM使用笔记

<http://rdc.taobao.com/blog/cs/?p=43#sec-7>

How to create an RPM package/zh-cn

<https://fedoraproject.org/wiki/How_to_create_an_RPM_package/zh-cn#.E6.96.B0.E5.BB.BA.E4.B8.80.E4.B8.AA.E7.A9.BA.E7.99.BD.E7.9A.84_.spec_.E6.96.87.E4.BB.B6>

Fedora 新软件维护者指南

<https://fedoraproject.org/wiki/Fedora_%E6%96%B0%E8%BD%AF%E4%BB%B6%E7%BB%B4%E6%8A%A4%E8%80%85%E6%8C%87%E5%8D%97?rd=Fedora%E9%8F%82%E6%9D%BF%EE%9A%9C%E6%B5%A0%E5%89%81%E6%B7%AE%E9%8E%B6%E3%82%88%E2%82%AC%E5%91%AE%E5%AF%9A%E9%8D%97?#.E4.BD.BF.E7.94.A8_Mock_.E5.92.8C_Koji_.E5.8E.BB.E6.B5.8B.E8.AF.95_RPM_.E5.8C.85>