

RPM Paket Yöneticisi

Devrim Gündüz

(devrim@gunduz.org)

Ali Erdinç Köroğlu

(info@erdinc.info)

<http://seminer.linux.org.tr>

<http://www.belgeler.org>

<http://www.gunduz.org/belgeler.php>

Giriş

- Bu seminerde, aşağıdaki konular anlatılacaktır:
 - RPM ile tarball arasındaki farklar
 - RPM dünyasına giriş : Sorgulama
 - Paket bütünlük kontrolü
 - Yeni paket kurma
 - Paket güncelleme
 - Paket kaldırmak (silme)
 - Diğer RPM işlemleri

Giriş

- Bir RPM içindeki herhangi bir dosyayı listelemek ve o dosyayı almak
- SRPM kavramı
- Spec dosyası nedir?
- Bir spec dosyası yazımı
- Spec dosyası kullanarak RPM yapma
- SRPM'i yeniden derleme

RPM nedir?

- “Tarball” kavramı
- Paket yöneticilerinin doğması
- RPM : RPM Package Manager (! Red Hat Package Manager)

RPM Nedir?

- RPM bize Őu avantajları sunar:
- Kurulu olan paketlerin listesinin grlmesi
- Hangi dizine hangi dosyanın kurulduđunun sorgulanması

RPM Nedir?

- Hangi dosyanın hangi pakete ait olduğunun sorgulanabilmesi
- Paket bağımlılıklarının takibi
- Paketlerin sistemden kaldırılabilmesi
- Kolay güncelleme imkanı
- ...

RPM ile tarball arasındaki farklar

- RPM'deki sürüm bilgisi
- Bağımlılık takibi

Sorgulama

- rpm -q
- rpm -qf
- rpm -qi
- rpm -qa
- rpm -qd
- rpm -qc
- rpm -ql

Dikkat: rpm -q -i != rpm -qi

Bütünlük kontrolü

- Neden bütünlük kontrolü?
- rpm -V
- rpm -Va
- # rpm -Va

S.5....T c /etc/pam.d/system-auth

SM5....T c /etc/sysconfig/rhn/up2date

S.5....T c /etc/sysconfig/rhn/up2date-uuid

Bütünlük kontrolü

Sıra No	Sembol	Anlamı
1	S	Dosya boyutu deęişmiş
2	M	Mod ya da izinleri deęişmiş
3	5	Checksum deęişmiş
4	D	Aygıt majör ya da minör numaraları deęişmiş
5	L	Kısayol deęişmiş
6	U	Kullanıcı sahiplięi deęişmiş
7	G	Grup sahiplięi deęişmiş
8	T	Son düzenlenme tarihi deęişmiş

Yeni Paket Kurma

- rpm -i
 - rpm -i postgresql-8.0.3-1PGDG.i686.rpm
- Paket adındaki her bir bölümün ayrı bir anlamı vardır.
- Yazılımın adı, sürüm ve alt sürüm numarası, mimari, ...
- Bütünlük kontrolü

Yeni Paket Kurma

- **-h** : Hash marking (kurulum düzey işaretini gösterir)
- **-v** : Verbose modu
- **--test** : Kurulumu sadece deneysel olarak yapar.
- **--force** : Zorla kurulum (dosya ve paket çakışmalarında kullanılabilir)
- **--nodeps** : Dependency (bağımlılık) önemsenmeden kurulum yapılır (Bu işlemi yaparken ne yaptığınızı bilmeniz ve ne yaptığınıza emin olmanız gerekmektedir. Aksi takdirde kurduğunuz yazılım çalışmayabilir.
- **--replacefiles** : Aynı dosyalardan varsa yerine yenisini yazar.

Yeni Paket Kurma

- RPM, aynı adlı paketlerin aynı sürümlerinin kurulmasına izin vermez. Ancak bunun bazı istisnaları olabilir. Kernel paketi bunlardan bir tanesidir.
- Birbirine bağımlı (dependent) paketleriniz varsa, bunların bağımlılığını çözmek için hepsini aynı anda yükleyebilirsiniz.

Paket Güncelleme

- Paketin yeni bir sürümü
- rpm -U
- -v, -h, --test
- -U ile sürüm yükseltirken paketin sürüm numaraları kontrol edilir. Örneğin sistemde 8.0.1-1 sürümü kurulu iken 8.0.1-2 sürümüne geçiş yapılabilir. Eğer tersi bir durum söz konusu olursa uyarı verilir.

Paket Güncelleme

- Ancak eğer yine de bir alt sürüme geçmek istenirse, -U' ya `-oldpackage` parametresi eklenebilir:
- `# rpm -Uvh -oldpackage paketin.eski.sürümü`
- Uyarı: Her RPM paketi için güncelleme işlemi uygun sonuç vermeyebilir. Bir yazılımı güncellemeden önce ilgili yazılımın sürüm yükseltme/azaltma yönergelerini mutlaka okuyunuz. Aksi takdirde veri kaybı yaşayabilirsiniz.

Paket Kaldırmak (Silme)

- Paket kaldırma sürecinde, paket kurulma sürecinin benzeri ama tersi bir süreç izlenir. Öncelikle, o pakete bağımlı olan başka paket ya da paketlerin olup olmadığı RPM veritabanından kontrol edilir. Eğer bu kontrol sonucunda veritabanında paket(ler) bulunursa bunlar kullanıcıya bildirilir.

Paket Kaldırmak (Silme)

- . Eğer bağımlılıklarda sorun yoksa, *preun* (*pre-uninstall*) betiği (*script*) çalıştırılır. Örneğin bir sunucu paketi kaldırılmadan önce o sunucunun durdurulması gerekecektir. Ardından, değiştirilmiş ayar dosyaları, dosya adının sonuna *.rpmsave* yazılıp saklanır. Varsa *postun* (*post-uninstall*) betiği çalıştırılır (örneğin, bir *lib* paketinin kaldırılmasından sonra *ldconfig* çalıştırılabilir) Bu aşamalardan sonra paket sistemden kaldırılmış olur.

Paket Kaldırmak (Silmek)

- Kurulum sürecinde olduğu gibi --test parametresi ile paket silme işleminin sadece denenmesi sağlanabilir. Böylece olası sonuçlar önceden görülebilir. --nodeps parametresi de kurulum sürecindeki benzer olarak kaldıracağınız pakete olan bağımlılıkların gözardı edilmesini sağlar.

Paket Kaldırmak (Silme)

- Eğer `preun` ve `postun` betiklerinin çalıştırılmaması istenmiyorsa, `rpm` komutuna `-noscripts` parametresi verilir.

Diğer işlemler

- rpm ile yapılabilecek işlemler bunlarla sınırlı değildir. RPM' in altyapısına yönelik bazı parametreler de mevcuttur. --rebuilddb bunlardan birtanesidir. RPM veritabanının bir şekilde zarar görmesi durumunda çalıştırılabilir. --vv ile de verbose mode kullanılabilir.

- `# rpm --rebuilddb -vv`
- D: rebuilding database /var/lib/rpm into /var/lib/rpmrebuilddb.7984
- D: creating directory /var/lib/rpmrebuilddb.7984
- D: opening old database with dbapi 3
- D: opening db environment /var/lib/rpm/Packages joinenv
- ...
- Bir başka parametre de `--initdb`'dir. Bu parametre çok dikkatli kullanılmalıdır; zira tüm RPM veritabanı ilklendirilecektir!!!

RPM içindeki dosyalar ile ilgili işlemler

- rpm2cpio (=rpm -q1p)
- # rpm2cpio postgresql-8.0.1-2PGDG.i686.rpm
lcpio -t
- ./usr/bin/clusterdb
- ./usr/bin/createdb
- ./usr/bin/createlang
- ./usr/bin/createuser

RPM içindeki dosyalar ile ilgili işlemler

- `# rpm2cpio postgresql-8.0.1-2PGDG.i686.rpm`
`lcpio -ivd ./usr/bin/createuser`
`./usr/bin/createuser`
`19126 blocks`
- mevcut dizinimizin içindeki `usr/bin` dizini içinde `createuser` dosyası oluşturulacaktır.

SRPM kavramı

- “Source” RPM
- src.rpm
- Yeniden derlenebilir, özelleştirilebilir.
- Platforma özel paket yapılabilir.

SRPM

- spec dosyası
- /usr/src/redhat/SPECS
- /usr/src/redhat/RPMS
 - /usr/src/redhat/RPMS/i{3,4,5,6}86, noarch, x86_64
- /usr/src/redhat/SOURCES
- /usr/src/redhat/BUILD

SRPM

- rpmbuild!
- Parametreleri:
 - rpmbuild -ba -bb -bp -bc -bi -bl -bs ... spec.dosyası
 - rpmbuild -ta -tb -tp -tc -ti -tl -ts kaynakkod.tar.gz
 - rpmbuild -rebuild

Spec dosyası

- Paketleme sürecini gösterir.
- Paket ile ilgili bilgiler içerir.
- Olası yamaları içerir.
- Derleme parametrelerini içerir.
- Kurulumdan önce ya da sonra çalıştırılabilecek betikleri (script) içerir.
- Değişiklik bilgilerini (changelog) içerir.

Örnek bir spec dosyası - Slony-1

- `%{!?perltools:%define perltools 1 }`
-
- **Summary:** A "master to multiple slaves" replication system with cascading and failover.
- **Name:** slony1
- **Version:** 1.1.0
- **Release:** beta3
- **License:** Berkeley/BSD

Örnek bir spec dosyası - Slony-1

- Group: Applications/Databases
- URL: <http://slony.info/>
- Packager: Devrim Gunduz
<devrim@PostgreSQL.org>
- Source0: slony1-`{version}`.`{release}`.tar.gz
- Buildroot: `{_tmppath}`/`{name}`-`{version}`-root
- BuildRequires: postgresql-devel
- Requires: postgresql-server

Örnek bir spec dosyası - Slony-1

- Patch1: slony-installation.sgml.patch
- Patch3: slony-bestpractices.sgml.patch
- Patch5: slony-concepts.sgml.patch
- Patch7: slony-usingslonik.sgml.patch

Örnek bir spec dosyası - Slony-1

- **%description**
- Slony-I will be a "master to multiple slaves" replication system with cascading and failover.
- The big picture for the development of Slony-I is to build a master-slave system that includes all features and capabilities needed to replicate large databases to a reasonably limited number of slave systems.
- Slony-I is planned as a system for data centers and backup sites, where the normal mode of operation is that all nodes are available

Örnek bir spec dosyası - Slony-1

- **%prep**
- %setup -q -n slony1-%{version}-%{release}
- %patch1 -p0
- %patch3 -p0
- %patch5 -p0
- %patch7 -p0

Örnek bir spec dosyası - Slony-1

- **%build**
- autoconf
- CFLAGS="\$\${CFLAGS:-%optflags}" ; export CFLAGS
- CXXFLAGS="\$\${CXXFLAGS:-%optflags}" ; export CXXFLAGS
- CPPFLAGS="\$\${CPPFLAGS} -I%{_includedir}/et" ; export CPPFLAGS
- CFLAGS="\$\${CFLAGS} -I%{_includedir}/et" ; export CFLAGS%

Örnek bir spec dosyası - Slony-1

- `./configure --bindir=/usr/bin --libdir % {_libdir} --includedir % {_includedir}/pgsql \`
- `%if %perltools`
- `--with-perltools==% {_bindir} \`
- `%endif`
- `--datadir % {_datadir}/pgsql --sysconfdir=/etc
 --exec-prefix=/usr/bin --with-pglibdir=% {_libdir}/
pgsql`

Örnek bir spec dosyası - Slony-1

- `%install`
- `rm -rf $RPM_BUILD_ROOT`
- `mkdir -p $RPM_BUILD_ROOT/etc`
- `mkdir -p $RPM_BUILD_ROOT%{_datadir}/pgsql`
- `mkdir -p $RPM_BUILD_ROOT/%{_libdir}/pgsql`
- `make DESTDIR=$RPM_BUILD_ROOT install`

Örnek bir spec dosyası - Slony-1

- `install -m 0755 src/backend/slony1_funcs.so
$RPM_BUILD_ROOT%{_libdir}/pgsql/slony1_funcs.so`
- `install -m 0755 src/xxid/xxid.so $RPM_BUILD_ROOT%{_libdir}/
pgsql/xxid.so`
- `install -m 0755 src/backend/*.sql $RPM_BUILD_ROOT%
{_datadir}/pgsql/`
- `install -m 0755 src/xxid/*.sql $RPM_BUILD_ROOT%{_datadir}/
pgsql/`
- `install -m 0755 tools/*.sh $RPM_BUILD_ROOT%{_bindir}/`
- `install -m 0755 share/slon.conf-sample $RPM_BUILD_ROOT%
{_sysconfdir}/slon.conf`

Örnek bir spec dosyası - Slony-1

- `%if %perltools`
- `cd tools`
- `make DESTDIR=$RPM_BUILD_ROOT install`
- `/bin/rm -rf altperl/*.pl altperl/ToDo altperl/README
altperl/Makefile altperl/CVS`
- `mv altperl/slony_tools.conf-sample
$RPM_BUILD_ROOT%{_sysconfdir}/slony_tools.conf`
- `/bin/rm -f $RPM_BUILD_ROOT%{_sysconfdir}/
slony_tools.conf-sample`

Örnek bir spec dosyası - Slony-1

- `install -m 0755 altperl/* $RPM_BUILD_ROOT%{_bindir}/`
- `install -m 0755 altperl/slony-tools.pm`
`$RPM_BUILD_ROOT%{_libdir}/pgsql/`
- `/bin/rm -f $RPM_BUILD_ROOT%{_libdir}/slony-tools.pm`
- `/bin/rm -f $RPM_BUILD_ROOT%{_bindir}/slony-tools.pm`
- `%endif`

Örnek bir spec dosyası - Slony-1

- %clean
- `rm -rf $RPM_BUILD_ROOT`
- %post
- `echo "Slony1 has been successfully installed.
Before running slony,"`
- `echo "be sure to edit /etc/slony.conf-sample and
rename it to"`
- `echo "/etc/slony.conf"`

Örnek bir spec dosyası - Slony-1

- %files
- %defattr(-,root,root,-)
- %doc COPYRIGHT doc/adminguide doc/concept doc/howto doc/implementation doc/support
- %{_bindir}/*
- %{_libdir}/pgsql/slony1_funcs.so
- %{_libdir}/pgsql/xxid.so
- %{_datadir}/pgsql/*.sql
- %{_sysconfdir}/slon.conf

Örnek bir spec dosyası - Slony-1

- `%if %perltools`
- `%{_libdir}/pgsql/slony-tools.pm`
- `%{_sysconfdir}/slony_tools.conf`
- `%endif`

Örnek bir spec dosyası - Slony-1

- %changelog
- * Mon Apr 10 2005 Devrim Gunduz <devrim@PostgreSQL.org>
postgresql-slony1-engine
- - More fixes on RPM builds
- * Thu Apr 04 2005 Devrim Gunduz <devrim@PostgreSQL.org>
postgresql-slony1-engine
- - Fix RPM build errors, regarding to tools/ .
- * Thu Apr 02 2005 Devrim Gunduz <devrim@PostgreSQL.org>
postgresql-slony1-engine
- - Added docs to installed files list.
- - Fixed doc problems
- - Updated the spec file

Belgenin güncel hali

- <http://www.gunduz.org/belgeler.php>
- <http://seminer.linux.org.tr>

LKD Seminerleri

- Seminer programı, seminer notları ve ayrıntılı bilgiler için;

<http://seminer.linux.org.tr>

adresini ziyaret edebilirsiniz.

RPM Paket Yöneticisi

Devrim Gündüz

(devrim@gunduz.org)

Ali Erdiñç Körođlu

(info@erdinc.info)

<http://seminer.linux.org.tr>

<http://www.belgeler.org>

<http://www.gunduz.org/belgeler.php>