

Менеджер пакетов RPM

НКАбд-06-23

Улитина Мария Максимовна

Содержание

1	Введение	5
2	Менеджер пакетов RPM	6
2.1	Общая характеристика и основные понятия	6
2.2	Применение RPM	6
2.3	Преимущества RPM	7
2.4	Основы создания пакета в RPM	7
2.5	Проверка качества пакета	8
3	Выводы	9
4	Список литературы	10

Список иллюстраций

Список таблиц

1 Введение

В настоящее время распространение программного обеспечения связано с большим количеством файлов. Для управления файлами, связанными с программой используются специальные пакеты.

Система управления пакетами («менеджер пакетов» или «пакетный менеджер») — набор программного обеспечения, позволяющего управлять процессом установки, удаления, настройки и обновления различных компонентов программного обеспечения.

Такие системы зачастую используются в различных дистрибутивах операционной системы Linux и других UNIX-подобных операционных системах.

2 Менеджер пакетов RPM

2.1 Общая характеристика и основные понятия

RPM (RPM Package Manager) популярный формат пакетов программного обеспечения, а также программа для создания и управления этими пакетами.

Пакет программ — набор взаимосвязанных модулей, которые предназначены для решения задач определённого класса некоторой предметной области.

Существует два типа RPM пакетов:

1. source RPM (SRPM) - исходный;
2. binary RPM - бинарный.

Такие пакеты имеют одинаковый формат и инструменты для работы с ними, но обладают разным наполнением и служат для разных целей. SRPM содержит исходный код, дополнения к нему, а также SPEC файл, описывающий алгоритм преобразования исходного кода в binary RPM.

SPEC-файл содержит инструкции для создания RPM пакета.

2.2 Применение RPM

Пакетный менеджер RPM применяется для:

1. создание пакетов, пригодных для дистрибуции из файлов программ.
2. установки, обновления и удаления упакованных программ.

3. запроса детальной информации о пакетах программного обеспечения: установлены ли они или нет.
4. подтверждения целостности установленного программного обеспечения и результатов установки программного обеспечения.

2.3 Преимущества RPM

Пакетный менеджер RPM обладает рядом следующих полезных функций:

1. Возможность установки, удаления, обновления, а также подтверждения программных пакетов совместно с Yum.
2. Использование базы данных установленных пакетов для запроса информации о пакетах и подтверждение пакетов.
3. Использование метаданных для описания пакетов, их инструкций по установке и других параметров пакетов.
4. Возможность собрать исходные файлы программного обеспечения, патчи, инструкции установки в исходные и бинарные файлы.
5. Добавление пакетов в репозитории Yum.
6. Цифровая подпись пакетов с использованием ключей.

2.4 Основы создания пакета в RPM

При создании пакеты в RPM необходимо указать следующие параметры в преамбуле SPEC-файла: 1. Name - имя; 2. Version - версия; 3. Release - релиз; 4. Summary - краткое описание пакета; 5. License - лицензия. 6. URL - ссылка на дополнительную информацию о программе; 7. BuildArch - архитектура компьютера для запуска программы; 8. Requires - системные требования.

В основной части SPEC-файла содержатся следующие поля:

1. %description - полное описание программного обеспечения, содержащегося в пакете;

2. %prep - команда/команды, для подготовки ПО к сборке;
3. %build - команда/команды для самой сборки ПО;
4. %install - команда/команды, используемые для копирования необходимых в дальнейшем файлов, полученных при сборке;
5. %check - команда/команды для тестирования ПО;
6. %files - список файлов, которые будут установлены конечному пользователю программы;
7. %changelog - журнал изменений пакета.

Файл с вышеперечисленными полями необходимо сохранить в формате .spec.

После этого выполнить следующие команды:

```
$ rpmddev-setuptree $ rpmbuild -ba filename.spec
```

Команда rpmddev-setuptree создаст несколько необходимых для работы директорий.

rpmbuild создаст сам пакет RPM.

2.5 Проверка качества пакета

После создания пакета следует проверить его качество и работоспособность.

Основным инструментом для этого служит rpmlint.

rpmlint совершает следующие действия:

1. Улучшает поддерживаемость RPM;
2. Осуществляет проверку на ошибки и взаимосвязи компонентов пакета.

rpmlint может проверять бинарные, исходные и SPEC-файлы, поэтому полезен на всех этапах создания пакета.

3 Выводы

RPM - полезный пакетный менеджер, содержащий множество функций и параметров. В дополнение к своей функциональности он прост и понятен в использовании.

