www.altlinux.ru

Приложение для управления *rpm*-пакетами и контроля целостности операционной системы

Техническое задание

Альт Линукс — 2012

1 Общие требования

Система для управления *rpm*-пакетами должна состоять из нескольких отдельных утилит, набор которых детально описывается в последующих главах. При этом существует ряд общих требований, предъявляемых к любому приложению из всего множества. Ниже приводится их подробный перечень.

Поддержка протоколирования. Любая операция, приводящая к изменению состояния системы, должна поддерживать развёрнутое протоколирование всех совершаемых действий. Накапливаемый журнал необходимо снабжать как пометками, облегчающими чтение и поиск данных системным администратором, так и машинночитаемой информацией с целью обеспечения автоматизированной возможности возврата ОС к предыдущему состоянию в случае неудачного завершения транзакции. Записи должны разделяться по степени критичности содержания (уровни “отладка”, “основная информация”, “предупреждение”, “ошибка”, “критическая ошибка”). Для всех утилит требуется наличие ключа `-verbose`, повышающего детальность информации, выводимой на стандартный поток вывода.

Поддержка локализации. Система должна поддерживать механизм перевода интерактивных сообщений на национальные языки. Требование локализации не распространяется на службу протоколирования и утилиту построения индекса репозитория.

Поддержка средств контроля за искажением информации. Конечный пользователь системы должен иметь возможность выполнять проверку контрольной суммы и цифровой подписи для множества пакетов, полученных из удалённых репозиториев, и для служебной информации о них. Контроль необходимо выполнять на основе цифровой подписи автора автоматически во время работы. При обнаружении ошибки требуется уведомить об этом пользователя и добавить соответствующую информацию в журнал. Уведомление должно указывать, обнаружено ли только несовпадение контрольной суммы, найдена ли неправильная цифровая подпись или подпись выполнена неизвестным (просроченным) ключом.

Минимизация риска аварийного завершения операции. При любой

операции, предполагающей внесение изменений в состояние ОС, необходимо предварительно провести максимально доступное количество проверок с целью предотвращения неуспешного результата работы. Начало модификации системы допускается только в случае, если все необходимые данные доставлены на локальный компьютер, проверена их целостность, устранена возможность файловых конфликтов (см. гл. 4), и доступно достаточное количество свободного дискового пространства.

Поддержка параллельных вычислений. Все операции, предполагающие высокую вычислительную нагрузку на процессор, по мере возможности должны поддерживать параллельный режим работы. Разделение общей задачи на несколько процессов или потоков позволит задействовать несколько ядер в многоядерных системах, что способно значительно ускорить получение результата. Желательно, чтобы количество процессов или потоков в вычислении не превышало физическое количество ядер системы. Это требование не распространяется на процедуру загрузки данных по сети.

Кэширование данных из удалённых источников. При загрузке пакетов для установки из удалённых репозиториев в сети полученные файлы должны сохраняться в специально отведённом каталоге на случай повторного использования. При этом требуется наличие конфигурационных параметров, задающих положение каталога для хранения в системе и его максимальной размер. При превышении заданного порога размера кэша пакеты необходимо удалять с учётом давности последнего обращения.

2 Основные операции

Основное содержание этой главы представляет описание операций, производящих изменение в состоянии ОС. Назначение изменений определяется командой пользователя, но общий порядок работы у всех операций одинаковый. Он состоит из следующих этапов:

1. Анализ команды пользователя, просмотр загруженной индексной информации о подключённых репозиториях и поиск решения, представленного в виде списков пакетов для установки, для обновления и для удаления.

2. Загрузка необходимых пакетов из удалённых репозиториев.

3. Внесение изменений в ОС.

Реализация каждой команды должна содержать чёткое разделение приведённых этапов, позволяя пропускать или отменять любой из них, кроме первого. Требуется оснастить реализацию механизмом вызова внешних обработчиков завершения каждого этапа. Результат их работы должен указывать на необходимость продолжения или отмены дальнейших действий.

Зависимости, указанные как необходимые только для выполнения дополнительных скриптов установки/удаления пакета, удовлетворяются при работе соответствующих операций. Установленные таким образом дополнительные пакеты после завершения внесения изменений из ОС не удаляются автоматически, но могут быть удалены вручную.

2.1 Операция установки и обновления пакетов

Операции установки и обновления пакетов имеют единый интерфейс пользователя. В качестве основного параметра указывается множество пакетов, каждый из которых может быть задан одним из следующих способов:

- по имени;
- по имени с ограничением версии, включая возможность даунгрейда;
- по ссылке на файл пакета, включая возможность загрузки из сети;
- по одному из *provides* пакета (см. гл. 4), включая возможность указания ограничения версии.

При указании более одного пакета операция отменяется полностью для всех пакетов, если обнаружена проблема установки хотя бы одного из них. При вызове операции необходимо наличие возможности запрета пользователем рассмотрения решений, включающих установку некоторого пакета. Предполагается, что два пакета с одинаковыми именами, эпохами, версиями, релизами и отметками времени их создания являются одним и тем же пакетом. Конкретный пакет для установки выбирается на основе параметров приоритета репозитория, в котором он расположен (см. гл. 5). Только в случае указания точной версии или имени файла происходит однозначный выбор пакета для установки. В противном случае операция производит выбор подходящего варианта в соответствии с правилами, описаными ниже.

При указании множества пакетов для установки или обновления допускается возможность использования регулярных выражений. С этой целью требуется наличие аргумента командной строки, разрешающего обработку ввода пользователя как регулярного выражения, а также определяющего, должны ли регулярные выражения обрабатываться на множестве имён пакетов или с учётом известного списка всех *provides*. Все обнаруженные допустимые подстановки передаются на вход операции так, как если бы пользователь сделал перечисление вручную.

В стандартном поведении операция должна производить поиск решений с учётом экономии дискового пространства пользователя. Дополнительно требуется наличие параметра командной строки, запрещающего удаление уже установленных пакетов из системы.

2.1.1 Порядок выбора пакета по явному запросу

При установке пакета, перечисленного в команде пользователя, поиск подходящего кандидата ведётся следующим образом:

1. Если пользователь указал имя файла или запросил установку на основе *URL*, то устанавливается запрошенный пакет без рассмотрения дополнительных вариантов, описанных ниже. Если запрошенный пакет уже установлен, но установленная версия новее запрошенной, то операция не отменяется.

2. Если существует пакет, имя которого точно соответствует запрошенному, то выбирается его самая свежая версия, удовлетворяющая ограничениям. Прочие варианты, описанные ниже, не рассматриваются.

3. Если существуют только пакеты, подходящие под запрос на основе записей *provides*, то выбирается только один из них следующим образом:

   - если пользователь наложил ограничение версии, то рассматриваются только пакеты, соответствующие записи *provides* в которых содержат информацию о версии. Если таких пакетов несколько, то в указанном порядке последовательно выполняется обработка предустановленного списка приоритетов *provides* (см. гл. 2.6), сортировка по версии записи *provides* и сортировка по основному имени до тех пор, пока один из методов не исключит неоднозначность выбора;

- если пользователь ограничения версии не наложил, но все подходящие записи *provides* имеют информацию о версии, выбор осуществляется таким же способом, как описано в предыдущем пункте;
- если пользователь ограничения версии не наложил, и не все подходящие записи *provides* имеют информацию о версии, то выбор производится на основе предустановленного списка предпочтений или, если запись в списке предпочтений отсутствует, путём выбора последнего элемента в списке имён пакетов после сортировки по возрастанию.

2.1.2 Порядок выбора пакета при разрешении *requires*

При установке пакетов в рамках обработки зависимостей требуется просмотр известного множества записей *obsoletes*. При этом требуемый пакет может быть заменён на пакет с соответствующей записью *obsoletes* по следующим правилам:

1. Замена производится только в случае, если пакет не установлен или ограничение версии в записи *requires* не соответствует версии установленного пакета. При этом рассматриваются только реальные имена пакетов и их версий без учёта имеющихся записей *provides*.
2. Пакет с записью *obsoletes* должен содержать также запись *provides*, в которой имя пакета совпадает с именем пакета в записи *obsoletes*. Если запись *provides* отсутствует, запись *obsoletes* пропускается[1].
3. Если запись *requires*, по которой устанавливается пакет, содержит ограничение версии, то запись *provides* также должна содержать указание версии. Если указания версии в записи *provides* нет или версия не подходит под требование *requires* замена пакета не допускается.
4. Если обнаружено несколько подходящих пакетов для замены, то выбирается только один из них на основе записи *provides*, как это описано ниже.

Если требуемый пакет уже установлен (включая возможность наличия в системе пакета с подходящей записью *provides*), и его версия соответствует требованию версии *requires*, то никакие действия с ним не совершаются. Если

[1] Предполагается, что под действие записи *obsoletes* попадают только реальные имена пакетов, но не записи *provides*. В противном случае, требования наличия одновременно записей *obsoletes* и *provides* будет приводить к ситуации, в которой пакеты обновляют сами себя.

пакет не установлен или его версия является неподходящей, и не производилась замена пакета на основе записи *obsoletes*, то выполняются следующие действия:

1. Если пакет уже установлен, его версия является неподходящей, но подходящая версия доступна в репозиториях, то производится его обновление.
2. Если пакет уже установлен, его версия является неподходящей, и все доступные версии также не подходят, то рассматривается множество пакетов на основе *provides* с последующей установкой нового варианта. Причём уже установленный пакет из системы не удаляется.
3. Если пакет не установлен, но существует его кандидат, имя которого соответствует заданному, ему отдаётся приоритет, и устанавливается его подходящая самая свежая версия. Существующие пакеты, содержащие одноимённые *provides*, не рассматриваются.
4. Если запись *requires* имеет ограничение версии, то рассматриваются только пакеты, соответствующие записи *provides* в которых содержат информацию о версии. Если таких пакетов несколько, то в указанном порядке последовательно выполняется обработка предустановленного списка приоритетов *provides* (см. гл. 2.6), сортировка по версии записи *provides* и сортировка по основному имени до тех пор, пока один из методов не исключит неоднозначность выбора.
5. Если запись *requires* не имеет ограничений версии, но все подходящие записи *provides* имеют информацию о версии, выбор осуществляется таким же способом, как описано в предыдущем пункте.
6. Если запись *requires* не имеет ограничений версии, и не все подходящие записи *provides* имеют информацию о версии, то выбор производится на основе предустановленного списка предпочтений или, если запись в списке предпочтений отсутствует, путём выбора последнего элемента в списке имён пакетов после сортировки по возрастанию.

2.2 Операция удаления пакетов

При выполнении операции удаления указывается множество пакетов для обработки. Допускается использование только настоящих имён пакетов,

provides не рассматриваются. Операция производит изменения, после которых ни один из указанных пакетов не может присутствовать в ОС, но допускается установка дополнительных, если это требуется зависимостями. Ситуация вызова операции для отсутствующего пакета ошибочной не считается.

Для увеличения точности определения желаемого результата требуется возможность указания пользователем необходимости присутствия в системе некоторого пакета после окончания работы. Требуется наличие опции командной строки, запрещающей установку новых пакетов. Допускается использование регулярных выражений, которые применяются к именам (без учёта *provides*) всех установленных пакетов.

2.3 Операция переустановки пакетов

Операция переустановки одного или нескольких пакетов производится для исправления повреждённой ОС. В ходе работы пакет удаляется и заново устанавливается, сохраняя версию. При отсутствии необходимого пакета в кэше производится его загрузка из доступных репозиториев.

2.4 Операция обновления ОС

Операция обновления ОС соответствует операции обновления пакетов с указанием множества имён всех установленных пакетов. Сохраняется возможность пробного запуска работы только с выводом списка изменений на поток стандартного вывода или с загрузкой пакетов из репозиториев в сети. В случае обнаружения невозможности выполнения операции пользователю необходимо предоставить подробный отчёт с описанием проблемы и предложить сделать дополнительные указания для её разрешения.

2.5 Операция исправления целостности ОС

Операция контроля целостности ОС должна выполнить для каждого пакета проверку удовлетворения его зависимостей и конфликтов с другими пакетами. Процедуру исправления найденных ошибок требуется выполнять следующим образом:

1. Для каждой пары конфликтующих пакетов пользователю необходимо предоставить запрос для выбора пакета для удаления из ОС.

2. Дополнить список пакетов для удаления списком пакетов для установки, добавляя в него все неудовлетворённые зависимости.

3. Запустить операцию исправления ошибок, причём в ходе работы должны одновременно учитываться и пакеты для установки, и пакеты для удаления. В случае невозможности нахождения решения пользователю необходимо предоставить подробное описание проблемы с предложением установить или удалить часть пакетов вручную, после чего повторить операцию исправления целостности.

2.6 Дополнительные возможности

Требуется наличие ряда возможностей, модифицирующих поведение операций, изменяющих состояние ОС. Они подразумевают указание пользователем дополнительной информации, которая должна сохраняться в конфигурационных файлах.

Указание приоритетов разрешения *provides*. Пользователь должен иметь возможность составления списка приоритетного выбора пакетов при разрешении *provides*. Выбирается первый элемент, доступный для установки. В случае ошибки оставшиеся элементы не просматриваются, даже если с их использованием существует допустимое решение[2].

Указание списка пакетов для удержания. Пользователь должен иметь возможность указания списков пакетов, для которых запрещаются операции установки, обновления или удаления. Три списка должны быть раздельными и храниться в конфигурационных файлах системы, допуская использование регулярных выражений. В случае явного указания пользователем выполнить операцию, противоречащую содержимому одного из списков, необходимо выдать запрос на подтверждение продолжения работы.

Указание списка "важных" пакетов. Пользователь или администратор должен иметь возможность определения списка пакетов, удаление которых приводит ОС в неработоспособное состояние. Пакеты из указанного списка не могут удаляться в ходе поиска решений, а в случае явного указания их пользователем в операции удаления необходимо запросить

[2]Ограничение вызвано чрезмерной трудоёмкостью алгоритма поиска замыкания на основе *SAT solver*.

подтверждение, что пользователь понимает риск и последствия удаления запрошенного пакета.

Помимо приведённых функций пользователь должен иметь возможность запомнить некоторое состояние ОС для возврата к нему после выполнения установки пакетов для экспериментов.

3 Информационные операции

Наряду с операциями, предполагающими внесение изменений в ОС, необходимо наличие нескольких информационных операций, предоставляющих различные данные по запросу пользователя. Все информационные операции должны выводить результат с учётом всех имеющихся в репозиториях пакетов, а не ограничиваться множеством установленных.

Вывод подробной информации о пакете. Пользователь должен иметь возможность получения подробной информации о пакете, включающей имя, версию (эпоху, версию и релиз), архитектуру, автора (*packager*), *URL* проекта, лицензию, имя пакета с исходными текстами, однострочное и развёрнутое описание.

Поиск пакета. Пользователь должен иметь возможность поиска пакета по его имени (включая обработку регулярных выражений) по однострочному описанию и подробному описанию, указывая, какие из перечисленных полей необходимо просматривать.

Поиск пакетов, предоставляющих некоторый *provides*. Пользователь должен иметь возможность запроса списка всех пакетов, содержащих указанный *provides*. При запросе ограничения версии не указывается, но система выводит список подходящих пакетов с уточнением версии соответствующего *provides*.

Поиск нарушений целостности множества пакетов в репозиториях. Пользователь должен иметь возможность анализа содержимого подключенных репозиториев и получения списка пакетов, установка которых невозможна по причине отсутствия пакетов, удовлетворяющих некоторые зависимости. Список должен содержать имя пакета и соответствующую зависимость, порождающую ошибку.

Вывод бинарных пакетов на основе пакета с исходными текстами. Пользователь должен иметь возможность получения списка пакетов, которые были созданы путём сборки указанного пакета с исходными текстами. Требуется возможность указания нескольких пакетов с исходными текстами одновременно.

Вывод информации о зависимостях и конфликтах пакета. Пользователь должен иметь возможность просмотра информации о всех зависимостях и конфликтах указанного пакета, дополненной перечислением пакетов, удовлетворяющих каждому пункту выводимых данных. Требуется явно помечать зависимости, разрешение которых невозможно удовлетворить.

Вывод зависимых пакетов. Пользователь должен иметь возможность просмотра списка пакетов, зависимых от указанного. Другими словами, требуется вывести все пакеты, к удалению которых приведёт удаление указанного. Необходимо иметь возможность просмотреть как список пакетов, зависимых напрямую, так и список косвенно зависимых пакетов, т. е. включая транзитивные замыкания. Анализироваться должны только пакеты, установленные в системе.

Вывод информации о зависимостях между репозиториями. Пользователь должен иметь возможность контроля целостности содержимого некоторого репозитория (каждый пакет может быть установлен в систему без подключения сторонних репозиториев) и обнаружения необходимости задействования пакетов из других репозиториев.

4 Структура пакета

Описанные выше операции должны корректно обрабатывать структуру *rpm*-пакетов и придерживаться основных принципов их поведения во время установки и удаления. Информация о версии пакета подразумевает наличие следующих компонентов:

- Эпоха;
- Версия;
- Релиз.

Эпоха представляет из себя целое неотрицательное число. Версия и релиз могут быть произвольными наборами символов, за исключением использования символа "-". При сравнении версии различных пакетов система должна использовать функцию `rpmRangesOverlap()` из состава *librpm* и избегать собственной обработки. При ссылке на некоторую версию другого пакета указание эпохи и релиза необязательно, что подразумевает их произвольное значение.

Пакет может предоставлять функциональность других пакетов, указывая информацию об этом в тэге *provides*. Имя *provides* допускает произвольное значение, не обязательно совпадающее с именем какого-либо существующего пакета и является, скорее, соглашением, что пакет обладает некоторой совместимостью. Для *provides* допускается указание подмножества версии. Неявными *provides* считаются имена всех файлов, хранимых в пакете.

Следующие типы отношений допускаются на множестве пакетов:

Requires: пакет требует обязательное наличие другого пакета, указанного по его имени или по одному из его *provides*. Допускается указание подмножества версии требуемого пакета. В случае указания ограничения версии под *requires* может подходить *provides* только дополненный информацией о версии.

Conflicts: пакет запрещает наличие другого пакета, указанного по его имени или по одному из его *provides*. Допускается указание подмножества версии конфликтуемого пакета. В случае указания ограничения версии под *conflicts* может подходить *provides* только дополненный информацией о версии.

Obsoletes: Пакет может указывать, что является обновлением некоторого множества пакетов. При установке такого пакета все пакеты, обновлением которых он является, удаляются из ОС. Попытка их установки после установки обновляющего пакета приводит к ошибке типа "установлена более свежая версия". Допускается указание подмножества версии обновляемых пакетов. В случае указания ограничения версии под *obsoletes* может подходить *provides* только дополненный информацией о версии.

Установка двух пакетов является невозможной, если для них обнаружены файловые конфликты. Файловыми конфликтами считаются:

- хранение файлов с одинаковыми именами, но с различной *md5*-суммой или с различными атрибутами (права доступа, идентификаторы владельца и группы, отметка времени создания);
- хранение каталогов с одинаковыми именами, но с разными атрибутами (права доступа, идентификаторы владельца и группы, отметка времени создания).

Все записи о конфликтах в пакете применяются только к другим пакетам, т. е. допускается установка в систему самоконфликтующих пакетов.

5 Репозитории пакетов

Источниками пакетов для установки в ОС должны служить репозитории, размещённые на ресурсах в сети или на съёмных носителях информации. Каждый репозиторий представляет из себя организованную специальным образом структуру каталогов и файлов, содержащую:

- бинарные пакеты для установки в ОС;
- пакеты с исходными текстами (необязательно);
- вспомогательную информацию для индексирования содержимого репозитория.

Структура вспомогательной информации не должна подразумевать возможность получения списка файлов в каталогах репозитория, поскольку такая функция отсутствует в некоторых сетевых протоколах, таких как, например, *HTTP*. Необходимо включение дополнительной информации, позволяющей выполнять проверку *md5*-суммы и электронной цифровой подписи. Индексные данные пакетов должны содержать всю необходимую информацию для работы вышеописанных операций, допуская доставку пакета на локальный компьютер только в случае его установки или по явному запросу пользователя.

Построение вспомогательной информации, предназначенной для хранения данных о наборе пакетов в репозитории, необходимо подготавливать при помощи специальной утилиты. Утилита должна проводить обзор множества пакетов из указанного каталога и сохранять созданные данные в предложенном пользователем месте. Требуется иметь возможность отбирать пакеты для внесения в индекс на основе заданного регулярного выражения.

В ходе построения вспомогательной информации файлы пакетов явно добавляются как *provides*. В силу избыточного их количества поведение этой функции должно ограничиваться. Необходимо предусмотреть следующие её режимы:

1. Регистрировать в качестве *provides* только файлы, для которых найдены соответствующие записи *requires*, *conflicts* и *obsoletes*. Такое поведение возможно, если характер репозитория подразумевает полноту множества хранимых пакетов, и использование других источников маловероятно.
2. Регистрация в качестве *provides* только файлов из указанного списка каталогов.
3. Регистрация всех файлов.

Репозиторий подключается к системе путём его регистрации в конфигурационных файлах. Индексная информация о пакетах после подключения загружается и регистрируется по явной команде пользователя. При указании репозитория необходимо иметь возможность назначить репозиторию приоритет в виде целого неотрицательного числа, на основе которого будет приниматься решение о выборе кандидата для установки при наличии одного и того же пакета в нескольких репозиториях. Необходимо наличие возможности управления множеством подключённых репозиториев в автоматическом режиме при помощи утилит командной строки.

6 Конфигурационные возможности

Работа операций для манипуляции пакетами и получения информации о них должна регулироваться набором конфигурационных файлов. Множество конфигурационных файлов разделяется на следующие группы:

- основные конфигурационные параметры;
- подключенные репозитории пакетов с указанием приоритетов их использования;
- зарегистрированные ключи авторов для проверки цифровых подписей;
- необходимые конфигурационные параметры для дополнительных возможностей, описанных в разд. 2.6.

Желательно, чтобы параметры каждой группы были вынесены в отдельный файл с возможностью сборки его окончательного варианта на основе фрагментов, размещённых в отдельном каталоге. Каждая утилита из состава комплекта должна быть снабжена параметром командной строки, указывающим положение конфигурационной информации, а также параметром, позволяющим выполнить чтение конфигурации и вывести его на стандартный поток вывода (возможно, в формате самого конфигурационного файла). При пробном чтении конфигурационной информации в выводе необходимо отображать полный набор параметров, включая не заданные пользователем в конфигурационных файлах, с их значениями по умолчанию.

Основные параметры должны включать в себя:

- текущую архитектуру системы;
- параметры загрузки файлов из сети (прокси-сервер и пр.);
- набор опций командной строки для каждой операции, которые должны быть неявно использованы при всех запусках;
- параметры кэширования пакетов, загруженных из сети (положение каталога для файлов и его максимальный размер).