Создание и управление локальными репозиториями в НАЙС ОС

Введение

Локальные репозитории позволяют хранить и управлять пакетами программного обеспечения на собственных серверах, обеспечивая быстрый и надежный доступ к ним. Это особенно полезно в корпоративных средах, где требуется централизованное управление программным обеспечением и минимизация зависимости от внешних интернет-ресурсов. В данной документации мы рассмотрим процесс создания и управления локальными репозиториями в НАЙС ОС, используя пакетные менеджеры `dnf` и `tdnf`.

Основы репозиториев в НАЙС ОС

Репозитории в НАЙС ОС состоят из коллекций пакетов и метаданных, которые описывают эти пакеты. Пакеты представляют собой архивы программного обеспечения, обычно в формате RPM (Red Hat Package Manager), а метаданные включают информацию о версиях, зависимостях и других характеристиках пакетов.

Основные команды для работы с репозиториями

Пакетные менеджеры `dnf` и `tdnf` предоставляют несколько основных команд для работы с репозиториями:

- `dnf install` — установка пакета.
- `dnf update` — обновление пакетов.
- `dnf remove` — удаление пакета.
- `dnf repolist` — отображение списка репозиториев.
- `dnf clean all` — очистка кэша репозиториев.

```
sudo dnf install package_name
sudo dnf update
sudo dnf remove package_name
sudo dnf repolist
sudo dnf clean all
```

Создание локального репозитория

Для создания локального репозитория потребуется выполнить следующие шаги:

1. Подготовка директории для репозитория

Создайте директорию, в которой будут храниться пакеты и метаданные репозитория:

```
sudo mkdir -p /var/www/html/localrepo
```

2. Копирование пакетов в директорию репозитория

Скопируйте RPM пакеты, которые вы хотите добавить в локальный репозиторий, в созданную директорию:

```
sudo cp /path/to/packages/*.rpm /var/www/html/localrepo/
```

3. Генерация метаданных репозитория

Для генерации метаданных используйте утилиту `createrepo`:

```
sudo dnf install createrepo
sudo createrepo /var/www/html/localrepo
```

4. Настройка веб-сервера для доступа к репозиторию

Для предоставления доступа к репозиторию по HTTP настроим веб-сервер, например Apache:

```
sudo dnf install httpd
sudo systemctl start httpd
sudo systemctl enable httpd
```

Добавьте конфигурацию для нового репозитория:

```
sudo nano /etc/httpd/conf.d/localrepo.conf
```

Добавьте следующие строки в файл конфигурации:

```
Alias /localrepo /var/www/html/localrepo

    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
```

Перезапустите Apache для применения изменений:

© 2026 NiceSOFT — z.niceos.ru

```
sudo systemctl restart httpd
```

Добавление локального репозитория на клиентскую машину

Теперь, когда локальный репозиторий настроен и доступен, добавим его на клиентскую машину.

1. Создание файла репозитория

Создайте файл с настройками нового репозитория в директории `/etc/yum.repos.d/`:

```
sudo nano /etc/yum.repos.d/localrepo.repo
```

Добавьте следующую конфигурацию в файл:

```
[localrepo]
name=Local Repository
baseurl=http://server_ip/localrepo/
enabled=1
gpgcheck=0
```

2. Очистка кэша и обновление метаданных

Очистите кэш и обновите метаданные репозиториев:

```
sudo dnf clean all
sudo dnf repolist
```

Управление локальным репозиторием

Управление локальным репозиторием включает добавление новых пакетов, обновление существующих и удаление старых пакетов.

1. Добавление новых пакетов

Для добавления новых пакетов в локальный репозиторий скопируйте их в директорию репозитория и обновите метаданные:

```
sudo cp /path/to/new/packages/*.rpm /var/www/html/localrepo/
sudo createrepo --update /var/www/html/localrepo
```

© 2026 NiceSOFT — z.niceos.ru

2. Удаление старых пакетов

Для удаления старых пакетов просто удалите их из директории репозитория и обновите метаданные:

```
sudo rm /var/www/html/localrepo/oldpackage.rpm
sudo createrepo --update /var/www/html/localrepo
```

3. Автоматизация обновления репозитория

Для автоматизации обновления репозитория можно использовать планировщик задач cron:

```
sudo crontab -e
```

Добавьте следующую строку для обновления репозитория каждую ночь в 2:00:

```
0 2 * * * /usr/bin/createrepo --update /var/www/html/localrepo
```

Подписывание пакетов и репозиториев

Для обеспечения безопасности и целостности пакетов и репозиториев можно использовать подписи GPG (GNU Privacy Guard).

1. Создание ключа GPG

Создайте пару ключей GPG:

```
gpg --gen-key
```

Следуйте инструкциям для создания ключа. После создания ключа экспортируйте публичный ключ:

```
gpg --export -a "Your Name" > RPM-GPG-KEY-yourname
```

© 2026 NiceSOFT — z.niceos.ru

2. Подпись пакетов

Подпишите пакеты с помощью утилиты `rpm`:

```
rpm --addsign /var/www/html/localrepo/*.rpm
```

3. Настройка репозитория для проверки подписи

Скопируйте публичный ключ на клиентскую машину и импортируйте его:

```
sudo cp RPM-GPG-KEY-yourname /etc/pki/rpm-gpg/
sudo rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-yourname
```

Обновите файл конфигурации репозитория, чтобы включить проверку подписи:

```
sudo nano /etc/yum.repos.d/localrepo.repo
```

Добавьте или измените строки:

```
gpgcheck=1
gpgkey=http://server_ip/localrepo/RPM-GPG-KEY-yourname
```

Создание и управление зеркалами репозиториев

Для повышения доступности и скорости загрузки пакетов можно создать зеркала репозиториев.

1. Синхронизация репозиториев

Для синхронизации репозиториев используйте утилиту `reposync`:

```
sudo dnf install yum-utils
sudo reposync -r base -p /var/www/html/mirrorrepo/
```

2. Настройка веб-сервера для зеркала

Настройте веб-сервер для предоставления доступа к зеркалу:

© 2026 NiceSOFT — z.niceos.ru

```
sudo nano /etc/httpd/conf.d/mirrorrepo.conf
```

Добавьте следующие строки в файл конфигурации:

```
Alias /mirrorrepo /var/www/html/mirrorrepo

    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
```

Перезапустите Apache для применения изменений:

```
sudo systemctl restart httpd
```

3. Настройка клиента для использования зеркала

Обновите файл конфигурации репозитория на клиентской машине:

```
sudo nano /etc/yum.repos.d/localrepo.repo
```

Измените `baseurl` на URL зеркала:

```
baseurl=http://mirror_ip/mirrorrepo/
```

Мониторинг и обслуживание репозиториев

Регулярный мониторинг и обслуживание репозиториев поможет обеспечить их стабильную работу и актуальность пакетов.

1. Проверка доступности репозиториев

Используйте команды `dnf repolist` и `dnf check-update` для проверки доступности и актуальности репозиториев:

```
sudo dnf repolist
sudo dnf check-update
```

© 2026 NiceSOFT — z.niceos.ru

2. Логирование и аудит

Настройте логирование для отслеживания доступа к репозиторию и изменений в нем. Используйте стандартные журналы веб-сервера и системы:

```
sudo tail -f /var/log/httpd/access_log
sudo tail -f /var/log/httpd/error_log
```

3. Очистка устаревших пакетов

Регулярно очищайте устаревшие и неиспользуемые пакеты из репозитория:

```
sudo find /var/www/html/localrepo/ -name '*.rpm' -mtime +30 -exec rm {} \;
sudo createrepo --update /var/www/html/localrepo
```

Заключение

Создание и управление локальными репозиториями в НАЙС ОС позволяет централизовать управление программным обеспечением, улучшить доступность и безопасность, а также снизить зависимость от внешних источников. Следуя приведенным инструкциям, вы сможете настроить локальные репозитории, управлять пакетами и обеспечивать их актуальность. Регулярный мониторинг и обслуживание репозиториев поможет поддерживать их стабильную работу и своевременное обновление пакетов.

© 2026 NiceSOFT — z.niceos.ru