

Отчёт по лабораторной работе №4

Работа с программными пакетами

Акунаева Антонина Эрдниевна

Содержание

1 Цель работы	5
2 Задание	6
3 Выполнение лабораторной работы	7
4 Выводы	20
Список литературы	21

Список иллюстраций

3.1 Определение списка репозиториев	7
3.2 Вывод списка репозиториев	8
3.3 Определение информации о nmap	9
3.4 Установка nmap	9
3.5 Установка nmap*	10
3.6 Удаление nmap	10
3.7 Определение групп пакетов dnf 1	11
3.8 Определение групп пакетов dnf 2	11
3.9 Установка RPM DevTools	12
3.10 Удаление RPM DevTools	12
3.11 История использования dnf	13
3.12 Установка rpm-пакета lynx	14
3.13 Установка lynx из пакета и информация о нём	15
3.14 Вывод списка файлов в пакете rpm	15
3.15 Вывод списка файлов с документацией пакета	16
3.16 Скрипты и удаление lynx	16

Список таблиц

1 Цель работы

Получить навыки работы с репозиториями и менеджерами пакетов. [1]

2 Задание

1. Изучите, как и в каких файлах подключаются репозитории для установки программного обеспечения; изучите основные возможности (поиск, установка, обновление, удаление пакета, работа с историей действий) команды `dnf` (см. раздел 4.4.1).
2. Изучите и повторите процесс установки/удаления определённого пакета с использованием возможностей `dnf` (см. раздел 4.4.1).
3. Изучите и повторите процесс установки/удаления определённого пакета с использованием возможностей `grpm` (см. раздел 4.4.2).

3 Выполнение лабораторной работы

4.4.1. Работа с репозиториями

Зайдём в аккаунт и терминал как суперпользователь. Перейдём в каталог репозиториев /etc/yum.repos.d и выведем на экран список репозиториев, а после содержимое каждого из файлов репозиториев через команду cat с подстановкой названия одного из 4-ёх имеющихся пакетов (рис. 3.1):

```
su -  
cd /etc/yum.repos.d  
ls  
cat rocky[name].repo
```

```
[aeakunaeva@aeakunaeva ~]$ su -  
Password:  
[root@aeakunaeva ~]# cd /etc/yum.repos.d  
[root@aeakunaeva yum.repos.d]# ls  
rocky-addons.repo  rocky-devel.repo  rocky-extras.repo  rocky.repo  
[root@aeakunaeva yum.repos.d]# cat rocky-addons.repo  
# rocky-addons.repo  
#  
# The mirrorlist system uses the connecting IP address of the client and the  
# update status of each mirror to pick current mirrors that are geographically  
# close to the client. You should use this for Rocky updates unless you are  
# manually picking other mirrors.  
#  
# If the mirrorlist does not work for you, you can try the commented out  
# baseurl line instead.  
  
[highavailability]  
name=Rocky Linux $releasever - High Availability  
mirrorlist=https://mirrors.rockylinux.org/mirrorlist?arch=$basearch&repo=HighAvailabilty-$releasever$rltype  
#baseurl=http://dl.rockylinux.org/$contentdir/$releasever/HighAvailability/$basearch/os/  
gpgcheck=1
```

Рис. 3.1: Определение списка репозиториев

Выведем список репозиториев (рис. 3.2):

```
dnf repolist
```

Список отображает ID репозитория и название - в нашем случае это три репозитория для Rocky Linux версии 9, стандартные установленные AppStream, BaseOS и Extras.

Выведем также список пакетов, имеющих где-либо в названии или описании *user*:

```
dnf search user
```

Получим список всех возможных названий установленных пакетов, имеющих *user* в названии - и, следующими по списку, в описании.

```
[root@aekunaeva yum.repos.d]# dnf repolist
repo id          repo name
appstream        Rocky Linux 9 - AppStream
baseos           Rocky Linux 9 - BaseOS
extras           Rocky Linux 9 - Extras
[root@aekunaeva yum.repos.d]# dnf search user
Last metadata expiration check: 13 days, 23:38:38 ago on Sat 13 Sep 2025 08:37:5
4 PM MSK.
===== Name & Summary Matched: user =====
gnome-user-docs.noarch : GNOME User Documentation
libuser.x86_64 : A user and group account administration library
libuser.i686 : A user and group account administration library
perl-User-pwent.noarch : By-name interface to Perl built-in user name resolver
samba-usershares.x86_64 : Provides support for non-root user shares
usermode.x86_64 : Tools for certain user account management tasks
usermode-gtk.x86_64 : Graphical tools for certain user account management tasks
userspace-rcu.x86_64 : RCU (read-copy-update) implementation in user-space
userspace-rcu.i686 : RCU (read-copy-update) implementation in user-space
util-linux-user.x86_64 : libuser based util-linux utilities
xdg-user-dirs.x86_64 : Handles user special directories
===== Name Matched: user =====
anaconda-user-help.noarch : Content for the Anaconda built-in help system
gnome-shell-extension-user-theme.noarch : Support for custom themes in GNOME
```

Рис. 3.2: Вывод списка репозиториев

Установим nmap, изучив информацию о нём (рис. 3.3-рис. 3.5):

```
dnf search nmap
```

```
dnf info nmap
```

```
dnf install nmap
```

```
dnf install nmap\*
```

```
[root@aakunaeva yum.repos.d]# dnf search nmap
=====
Name Exactly Matched: nmap
=====
nmap.x86_64 : Network exploration tool and security scanner
=====
Name & Summary Matched: nmap
=====
nmap-ncat.x86_64 : Nmap's Netcat replacement
=====
[root@aakunaeva yum.repos.d]# dnf info nmap
Rocky Linux 9 - BaseOS          11 kB/s | 4.1 kB   00:00
Rocky Linux 9 - BaseOS          3.4 MB/s | 2.5 MB   00:00
Rocky Linux 9 - AppStream        13 kB/s | 4.5 kB   00:00
Rocky Linux 9 - AppStream        7.1 MB/s | 9.5 MB   00:01
Rocky Linux 9 - Extras           96 B/s  | 2.9 kB   00:31
Available Packages
Name      : nmap
Epoch     : 3
Version   : 7.92
Release   : 3.el9
Architecture : x86_64
Size      : 5.4 M
Source    : nmap-7.92-3.el9.src.rpm
Repository : appstream
Summary   : Network exploration tool and security scanner
URL       : http://nmap.org/
License   : Nmap
Description: Nmap is a utility for network exploration or security auditing.
```

Рис. 3.3: Определение информации о nmap

```
[root@aakunaeva yum.repos.d]# dnf install nmap
Last metadata expiration check: 0:00:34 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Dependencies resolved.
=====
Package      Architecture Version       Repository      Size
=====
Installing:
  nmap        x86_64      3:7.92-3.el9      appstream      5.4 M

Transaction Summary
=====
Install 1 Package

Total download size: 5.4 M
Installed size: 24 M
Is this ok [y/N]: y
Downloading Packages:
nmap-7.92-3.el9.x86_64.rpm          7.2 MB/s | 5.4 MB   00:00
-----
Total                                         4.9 MB/s | 5.4 MB   00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
```

Рис. 3.4: Установка nmap

```
[root@aekunaeva yum.repos.d]# dnf install nmap\*
Last metadata expiration check: 0:00:53 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Package nmap-3:7.92-3.el9.x86_64 is already installed.
Package nmap-ncat-3:7.92-3.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@aekunaeva yum.repos.d]# dnf remove nmap
Dependencies resolved.
=====
 Package      Architecture   Version       Repository    Size
 =====
 Removing:
  nmap        x86_64        3:7.92-3.el9     @appstream   24 M

Transaction Summary
=====
Remove 1 Package

Freed space: 24 M
Is this ok [y/N]: y
Running transaction check
Transaction check succeeded.
```

Рис. 3.5: Установка nmap*

Запись `dnf install nmap` отличается от `dnf install nmap|8` тем, что первая просто устанавливает пакет nmap, тогда как `nmap*` устанавливает все пакеты, связанные с nmap (в нашем случае nmap и ncat).

Удаление работает по тому же принципу. После установки удалим установленные пакеты (рис. 3.6):

`dnf remove nmap`

`dnf remove nmap\`

```
[root@aekunaeva yum.repos.d]# dnf remove nmap\*
Dependencies resolved.
=====
 Package      Architecture   Version       Repository    Size
 =====
 Removing:
  nmap-ncat   x86_64        3:7.92-3.el9     @AppStream   469 k

Transaction Summary
=====
Remove 1 Package

Freed space: 469 k
Is this ok [y/N]: y
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing          :                                1/1
  Running scriptlet: nmap-ncat-3:7.92-3.el9.x86_64  1/1
  Erasing           : nmap-ncat-3:7.92-3.el9.x86_64  1/1
```

Рис. 3.6: Удаление nmap

Нужно установить RPM Development Tools, для этого перед установкой получим информацию о группах пакетов dnf (рис. 3.7-рис. 3.8):

```
dnf groups list  
LANG=C dnf groups list  
dnf groups info "RPM Development Tools"
```

```
[root@aeakunaeva yum.repos.d]# dnf groups list  
Last metadata expiration check: 0:02:08 ago on Sat 27 Sep 2025 08:18:21 PM MSK.  
Available Environment Groups:  
  Server  
  Minimal Install  
  Workstation  
  Custom Operating System  
  Virtualization Host  
Installed Environment Groups:  
  Server with GUI  
Installed Groups:  
  Container Management  
  Development Tools  
  Headless Management  
Available Groups:  
  Legacy UNIX Compatibility  
  Console Internet Tools  
  .NET Development  
  Graphical Administration Tools  
  Network Servers  
  RPM Development Tools  
  Scientific Support  
  Security Tools
```

Рис. 3.7: Определение групп пакетов dnf 1

```
[root@aeakunaeva yum.repos.d]# LANG=C dnf groups list  
Last metadata expiration check: 0:02:30 ago on Sat Sep 27 20:18:21 2025.  
Available Environment Groups:  
  Server  
  Minimal Install  
  Workstation  
  Custom Operating System  
  Virtualization Host  
Installed Environment Groups:  
  Server with GUI  
Installed Groups:  
  Container Management  
  Development Tools  
  Headless Management  
Available Groups:  
  Legacy UNIX Compatibility  
  Console Internet Tools  
  .NET Development  
  Graphical Administration Tools  
  Network Servers  
  RPM Development Tools  
  Scientific Support
```

Рис. 3.8: Определение групп пакетов dnf 2

Установим теперь группу RPM Development Tools (рис. 3.9):

```
dnf groupinstall "RPM Development Tools"
```

```
[root@aekunaeva yum.repos.d]# dnf groups info "RPM Development Tools"
Last metadata expiration check: 0:03:02 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Group: RPM Development Tools
  Description: Tools used for building RPMs, such as rpmbuild.
  Mandatory Packages:
    redhat-rpm-config
    rpm-build
  Default Packages:
    rpmdevtools
  Optional Packages:
    rpmlint
[root@aekunaeva yum.repos.d]# dnf groupinstall "RPM Development Tools"
Last metadata expiration check: 0:03:30 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Dependencies resolved.
=====
 Package           Arch      Version       Repository      Size
 =====
Installing group/module packages:
 rpmdevtools        noarch    9.5-1.el9     appstream      75 k
Installing dependencies:
 python3-argcomplete noarch    1.12.0-5.el9   appstream      61 k
Installing Groups:
 RPM Development Tools
```

Рис. 3.9: Установка RPM DevTools

Удалим группы пакетов RPM Development Tools и убедимся через ls, что он не установлен (рис. 3.10):

```
dnf groupremove "RPM Development Tools"
```

```
[root@aekunaeva yum.repos.d]# dnf groupremove "RPM Development Tools"
Dependencies resolved.
=====
 Package           Arch      Version       Repository      Size
 =====
Removing:
 rpmdevtools        noarch    9.5-1.el9     @appstream     195 k
Removing unused dependencies:
 python3-argcomplete noarch    1.12.0-5.el9   @appstream     219 k
Removing Groups:
 RPM Development Tools

Transaction Summary
=====
Remove 2 Packages
```

```
Freed space: 414 k
Is this ok [y/N]: y
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
```

Рис. 3.10: Удаление RPM DevTools

Определим историю использования команды dnf и получим семь строк за всё время использования. Нам нужно отменить 6-ое действие (удаление RPM

Development Tools), но, т.к. это является седьмым действием в нашем случае, а шестым - его установка, ничего не выйдет, потому вручную установим снова RPM Development Tools, как было сделано ранее (рис. 3.11):

dnf history

```
dnf history undo 6
```

Рис. 3.11: История использования dnf

4.4.2. Использование grp

Установим rpm-пакет lynx, предварительно выведя список пакетов lynx. Установим, добавим ключ *-downloadonly* для исключительно установки пакета без дальнейших манипуляций (рис. 3.12):

```
dnf list lynx
```

```
dnf install lynx --downloadonly
```

```
[root@aakunaeva yum.repos.d]# dnf list lynx
Last metadata expiration check: 0:12:26 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Available Packages
lynx.x86_64 2.8.9-20.el9 appstream
[root@aakunaeva yum.repos.d]# dnf install lynx --downloadonly
Last metadata expiration check: 0:12:45 ago on Sat 27 Sep 2025 08:18:21 PM MSK.
Dependencies resolved.
=====
 Package Architecture Version Repository Size
 =====
 Installing:
 lynx x86_64 2.8.9-20.el9 appstream 1.5 M

Transaction Summary
=====
Install 1 Package

Total download size: 1.5 M
Installed size: 6.1 M
DNF will only download packages for the transaction.
Is this ok [y/N]: y
Downloading Packages:
lynx-2.8.9-20.el9.x86_64.rpm 4.9 MB/s | 1.5 MB 00:00
```

Рис. 3.12: Установка rpm-пакета lynx

Найдём каталог с установленным пакетом lynx */var/cache/dnf/appstream-25485261a76941d3/packages* и перейдём в него, установим lynx указанной версии и далее определим его местоположение (*/usr/bin*). Определим местоположение lynx, его пакетов и получим информацию о нём (название, версия, описание и пр.) через -qi (рис. 3.13):

```
find /var/cache/dnf/ -name lynx*
rpm -Uvh lynx-<версия>.rpm
```

```
which lynx
rpm -qf $(which lynx)
rpm -qi lynx
```

```
[root@aeakunaeva yum.repos.d]# find /var/cache/dnf/ -name lynx*
/var/cache/dnf/appstream-25485261a76941d3/packages/lynx-2.8.9-20.el9.x86_64.rpm
[root@aeakunaeva yum.repos.d]# cd /var/cache/dnf/appstream-25485261a76941d3/packages
[root@aeakunaeva packages]# rpm -Uhv lynx-2.8.9-20.el9.x86_64.rpm
Verifying... ################################################ [100%]
Preparing... ################################################ [100%]
Updating / installing...
  1:lynx-2.8.9-20.el9 ################################################ [100%]
[root@aeakunaeva packages]# which lynx
/usr/bin/lynx
[root@aeakunaeva packages]# rpm -qf $(which lynx)
lynx-2.8.9-20.el9.x86_64
[root@aeakunaeva packages]# rpm -qi lynx
Name        : lynx
Version     : 2.8.9
Release     : 20.el9
Architecture: x86_64
Install Date: Sat 27 Sep 2025 08:35:09 PM MSK
Group       : Unspecified
Size        : 6356615
License     : GPLv2
Signature   : RSA/SHA256, Tue 16 Apr 2024 02:13:04 AM MSK, Key ID 702d426d350d27
```

Рис. 3.13: Установка lynx из пакета и информация о нём

Выведем список всех файлов в пакете (рис. 3.14):

```
rpm -ql lynx
```

```
[root@aeakunaeva packages]# rpm -ql lynx
/etc/lynx-site.cfg
/etc/lynx.cfg
/etc/lynx.lss
/usr/bin/lynx
/usr/lib/.build-id
/usr/lib/.build-id/5a
/usr/lib/.build-id/5a/06d3c4e1a8f4e4a2a1a5305737084bab728b85
/usr/share/doc/Lynx
/usr/share/doc/lynx/INSTALLATION
/usr/share/doc/Lynx/README
/usr/share/doc/lynx/docs
/usr/share/doc/lynx/docs/CHANGES2.3
/usr/share/doc/lynx/docs/CHANGES2.4
/usr/share/doc/lynx/docs/CHANGES2.5
/usr/share/doc/lynx/docs/CHANGES2.6
/usr/share/doc/lynx/docs/CHANGES2.7
/usr/share/doc/lynx/docs/CHANGES2.8
/usr/share/doc/lynx/docs/CRAWL.announce
/usr/share/doc/lynx/docs/FM.announce
/usr/share/doc/lynx/docs/IBMPC-Charsets.announce
/usr/share/doc/lynx/docs/README.TRST
/usr/share/doc/lynx/docs/README.chartrans
```

Рис. 3.14: Вывод списка файлов в пакете rpm

Выведем перечень файлов с документацией пакета (рис. 3.15):

```
[root@aakunaeva packages]# rpm -qd lynx
/usr/share/doc/lynx/INSTALLATION
/usr/share/doc/lynx/README
/usr/share/doc/lynx/docs/CHANGES2.3
/usr/share/doc/lynx/docs/CHANGES2.4
/usr/share/doc/lynx/docs/CHANGES2.5
/usr/share/doc/lynx/docs/CHANGES2.6
/usr/share/doc/lynx/docs/CHANGES2.7
/usr/share/doc/lynx/docs/CHANGES2.8
/usr/share/doc/lynx/docs/CRAWL.announce
/usr/share/doc/Lynx/docs/FM.announce
/usr/share/doc/lynx/docs/IBMPC-Charsets.announce
/usr/share/doc/lynx/docs/README.TRST
/usr/share/doc/lynx/docs/README.chartrans
/usr/share/doc/lynx/docs/README.cookies
/usr/share/doc/lynx/docs/README.defines
/usr/share/doc/lynx/docs/README.metrics
/usr/share/doc/lynx/docs/README.options
/usr/share/doc/lynx/docs/README.release
/usr/share/doc/lynx/docs/README.rootcerts
/usr/share/doc/lynx/docs/README.ssl
/usr/share/doc/lynx/docs/README.sslcerts
/usr/share/doc/lynx/docs/SOCKETSHR.announce
```

Рис. 3.15: Вывод списка файлов с документацией пакета

```
rpm -qd lynx
```

Наглядно ознакомимся с документацией *lynx* через *man*. Затем выведем местоположение и список скриптов пакета. Запустим в другом терминале *lynx*, введя команду *lynx*, чтобы убедиться, что всё было установлено корректно. После удаляем пакет *lynx*. Проверим корректность удаления через *ls* (рис. 3.16):

```
man lynx
```

```
rpm -qc lynx
```

```
rpm -q --scripts lynx
```

```
rpm -e lynx
```

```
ls
```

```
[root@aakunaeva packages]# man lynx
[root@aakunaeva packages]# rpm -qc lynx
/etc/lynx-site.cfg
/etc/lynx.cfg
/etc/lynx.lss
[root@aakunaeva packages]# rpm -q --scripts lynx
[root@aakunaeva packages]# rpm -e lynx
[root@aakunaeva packages]# ls
lynx-2.8.9-20.el9.x86_64.rpm
[root@aakunaeva packages]# rpm -e lynx*
error: package lynx-2.8.9-20.el9.x86_64.rpm is not installed
```

Рис. 3.16: Скрипты и удаление *lynx*

lynx

``` ([рис. @fig:017]):

![Текстовый браузер lynx] (image/17.PNG){#fig:017 width=70%}

**\*\*Установка dnsmasq\*\***

Установим dnsmasq, предварительно выведя список пакетов ([рис. @fig:018]):

```
dnf list dnsmasq dnf install dnsmasq
```

![Установка dnsmasq] (image/18.PNG){#fig:018 width=70%}

Определим расположение пакетов установленных пакетов rpm и самого dnsmasq. Выведем информацию о пакете dnsmasq и ознакомимся с ней ([рис. @fig:019]):

```
which dnsmasq rpm -qf $(which dnsmasq) rpm -qi dnsmasq
```

![Получение списков по dnsmasq 1] (image/19.PNG){#fig:019 width=70%}

Получим также дополнительную информацию в списках о всех файлах пакета, файлах документации и т.д.

```
rpm -ql dnsmasq rpm -qd dnsmasq man dnsmasq rpm -qc dnsmasq
```

![Получение списков по dnsmasq 2] (image/20.PNG){#fig:020 width=70%}

Выведем скрипты пакетов dnsmasq. Скрипты наглядно показывают использование нескольких скриптов (1-больше) для взаимодействия с пакетами dnsmasq ([рис. @fig:021]):

```
rpm -q --scripts dnsmasq
```

- 1 - установка и обновление пакетов
- 0 - удаление пакетов без обновления
- <=1 - обновление пакетов без удаления

![Скрипты dnsmasq] (image/21.PNG){#fig:021 width=70%}

После удаляем пакеты dnsmasq и проверяем успешность через ls ([рис. @fig:022]):

```
rpm -e dnsmask
```

![Скрипты и удаление dnsmasq] (image/22.PNG){#fig:022 width=70%}

# Контрольные вопросы

1. \*\*Какая команда позволяет вам искать пакет rpm, содержащий файл useradd?\*\*

```
rpm -qf [filename]
```

2. \*\*Какие команды вам нужно использовать, чтобы показать имя группы dnf, которая соде

```
dnf groupinfo dnf grouplist
```

3. \*\*Какая команда позволяет вам установить rpm, который вы загрузили из Интернета и к

```
rpm -i [packagename].rpm
```

4. \*\*Вы хотите убедиться, что пакет rpm, который вы загрузили, не содержит никакого оп

`rpm -pq --scripts [packagename]`

5. \*\*Какая команда показывает всю документацию в rpm?\*\*

`rpm -qd`

6. \*\*Какая команда показывает, какому пакету rpm принадлежит файл?\*\*

`rpm -qf “`

## **4 Выводы**

Я получила навыки работы с репозиториями и менеджерами пакетов.

# **Список литературы**

1. Кулабов Д.С. Работа с программными пакетами. ТУИС РУДН.