

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ им. В. И. ВЕРНАДСКОГО»  
ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ  
Кафедра компьютерной инженерии и моделирования

**ОТЧЕТ ПО ПРАКТИЧЕСКОМУ ЗАДАНИЮ №16**  
**«Системы виртуализации в среде ОС GNU/ Linux. Наблюдение и аудит в ОС**  
**GNU/Linux.»**

Практическая работа  
по дисциплине «Системное программное обеспечение»  
студента 3 курса группы ИВТ-б-о-222(1)  
Гоголева Виктора Григорьевича

09.03.01 «Направление подготовки»

Симферополь, 2025

## 1) Изучите возможности команды qemu-img:

- Создайте образ виртуального жёсткого диска в папке /tmp/ размером 1.5GB в формате vmdk с именем disk\_base\_\$USER.vmdk
- \$USER переменная среды окружения в которой хранится логин текущего пользователя

```

~ /study/3_2/SystemP0 P main qemu-img create -f vmdk /tmp/disk_
base_$USER.vmdk 2.28G
Formatting '/tmp/disk_base_vicotr.vmdk', fmt=vmdk size=2448131359 compat6=c
ff hwversion=undefined
~ /study/3_2/SystemP0 P main ls /tmp/disk base vicotr.vmdk ✓
/tmp/disk_base_vicotr.vmdk
~ /study/3_2/SystemP0 P main cat /tmp/disk base vicotr.vmdk
KDMVH:

# Disk DescriptorFile
version=1
CID=94fa9f1
parentCID=ffffffff
createType="monolithicSparse"

# Extent description
RW 4781507 SPARSE "disk_base_vicotr.vmdk"

# The Disk Data Base
#DDB

ddb.virtualHWVersion = "4"
ddb.geometry.cylinders = "4743"
ddb.geometry.heads = "16"
ddb.geometry.sectors = "63"
ddb.adapterType = "ide"
ddb.toolsVersion = "2147483647"

```

Рисунок 1 — создание образа виртуального жесткого диска объемом 2.28

Гигибайт через утилиту qemu-img

```

qemu-img info /tmp/disk_base_vicotr.vmdk
image: /tmp/disk_base_vicotr.vmdk
file format: vmdk
virtual size: 2.28 GiB (2448131584 bytes)
disk size: 12 KiB
cluster_size: 65536
Format specific information:
  cid: 156215793
  parent cid: 4294967295
  create type: monolithicSparse
  extents:
    [0]:
      virtual size: 2448131584
      filename: /tmp/disk_base_vicotr.vmdk
      cluster size: 65536
      format:
Child node '/file':
  filename: /tmp/disk_base_vicotr.vmdk
  protocol type: file
  file length: 320 KiB (327680 bytes)
  disk size: 12 KiB

```

Рисунок — проверка что образ диска успешно создан объемом хранилища  
2.28 Гиббита

(с) Измените формат образа на qcow2, изменив также расширение файла

```

qemu-img convert -f vmdk -O qcow2 /tmp/disk_base_$USER.vmdk /tmp/disk_
base_$USER.qcow2
cat /tmp/disk_base_vicotr.qcow2
QFiphWdirty bitcorrupt bitexternal data filecompression typeextended L2 entriesla
zy refcountsbitmapsraw external data%

```

Рисунок 2 — конвертирование vmdk формата образа в qcow2

(d) Увеличьте размер образа диска до 7Gb

```

❏ ~ ➤ qemu-img resize /tmp/disk_base_$USER.qcow2 7G
Image resized.
❏ ~ ➤

```

Рисунок 3 — изменение размера образа до 7 Гиббайт

```

❏ ~/study/3_2/SystemP0 ➤ main !1 ?1 ➤ qemu-img create -f qcow2 -o
backing_file=/tmp/disk_base_$USER.qcow2 /tmp/disk_$USER.qcow2
qemu-img: /tmp/disk_vicotr.qcow2: Backing file specified without backing fo
rmat
Detected format of qcow2.

```

Рисунок 4 — создание дочернего образа disk\_vicotr.qcow2 на основе образа disk\_base\_vicotr.qcow2

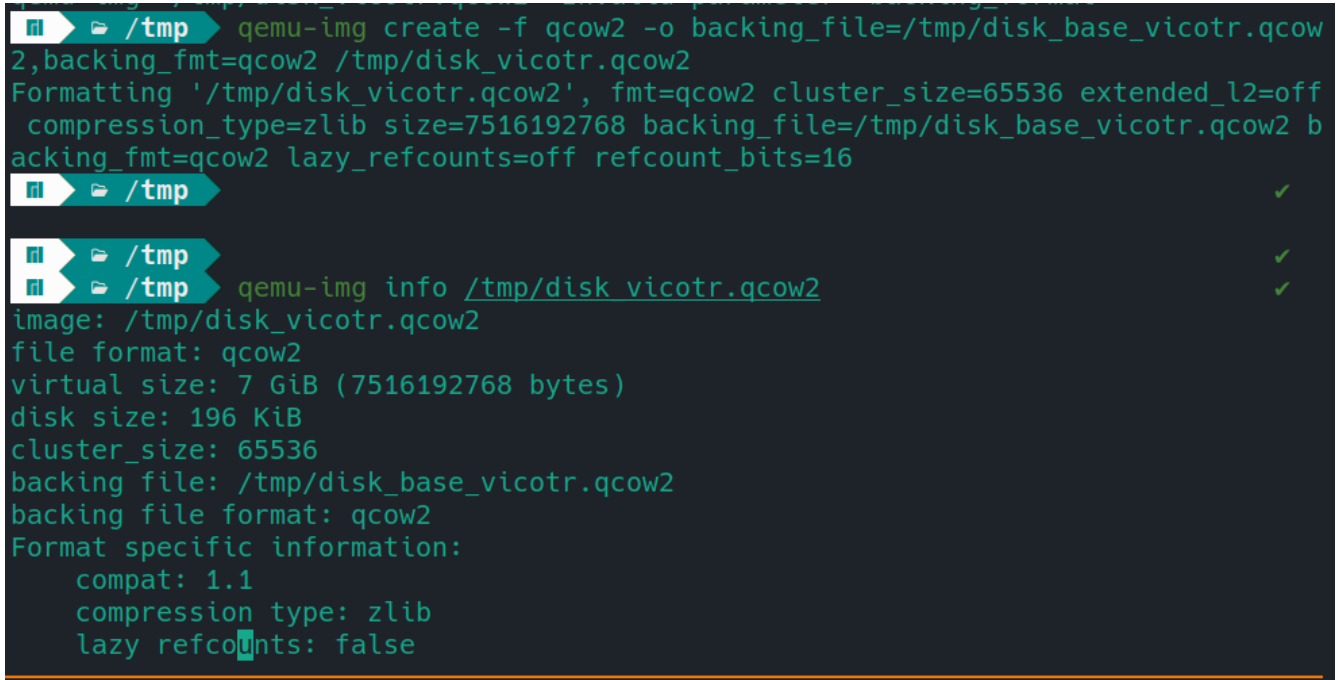
```

❏ ~ ➤ qemu-img info /tmp/disk_base_vicotr.qcow2
image: /tmp/disk_base_vicotr.qcow2
file format: qcow2
virtual size: 7 GiB (7516192768 bytes)
disk size: 200 KiB
cluster_size: 65536
Format specific information:
  compat: 1.1
  compression type: zlib
  lazy refcounts: false
  refcount bits: 16
  corrupt: false
  extended l2: false
Child node '/file':
  filename: /tmp/disk_base_vicotr.qcow2
  protocol type: file
  file length: 256 KiB (262656 bytes)
  disk size: 200 KiB
❏ ~ ➤

```

Рисунок 5 — проверка что размер образа диска изменился с 2.28 Гиббайт на 7 Гиббайта

(е) С помощью `qemu-img` создайте целевой (дочерний) образ диска, базирующийся на образе диска, созданном на предыдущем этапе. Образ в формате `qcow2` должен называться `disk_${USER}.qcow2` и располагаться в директории `/tmp/`



```

/tmp ➤ qemu-img create -f qcow2 -o backing_file=/tmp/disk_base_vicotr.qcow2,backing_fmt=qcow2 /tmp/disk_vicotr.qcow2
Formatting '/tmp/disk_vicotr.qcow2', fmt=qcow2 cluster_size=65536 extended_l2=off
compression_type=zlib size=7516192768 backing_file=/tmp/disk_base_vicotr.qcow2 b
acking_fmt=qcow2 lazy_refcounts=off refcount_bits=16

/tmp ➤

/tmp ➤ qemu-img info /tmp/disk_vicotr.qcow2
image: /tmp/disk_vicotr.qcow2
file format: qcow2
virtual size: 7 GiB (7516192768 bytes)
disk size: 196 KiB
cluster_size: 65536
backing file: /tmp/disk_base_vicotr.qcow2
backing file format: qcow2
Format specific information:
  compat: 1.1
  compression type: zlib
  lazy refcounts: false
  
```

Рисунок 6 — создание образа виртуального диска на основе базового и проверка через утилиту `qemu-img info` что он связан с базовым

2) Определите поддерживается ли гипервизор KVM на вашем оборудовании как описано в предыдущей главе (для тестов можно использовать файл CD-ROM `/var/qemu/OS/ubuntu14.iso`). Если KVM поддерживается, в дальнейшем используйте его при работе с VM.

3) Запустите виртуальную машину `qemu` с необходимыми параметрами:  
Количество процессоров 1

- Оперативная память 512Mb
- Тип эмулируемой видеокарты `std`

- Образ жёсткого диска образ, созданный вами на предыдущем этапе лабораторной работы (целевой) • Файл CD-ROM /var/qemu/OS/xubuntu14.iso
- Сеть пользовательская сеть
- Проброс портов: порт хост-компьютера = 8080) порт виртуальной машины = 80
- Включите отображение меню выбора устройства для загрузки
- Таймаут отображения меню 10 секунд
- Дополнительные опции:

```
e ).
[terminal icon] /var/qemu/OS sudo wget https://mirror.yandex.ru/ubuntu-releases/22
.04/ubuntu-22.04.5-desktop-amd64.iso
--2025-05-11 00:03:37-- https://mirror.yandex.ru/ubuntu-releases/22.04/ubu
ntu-22.04.5-desktop-amd64.iso
Загружен сертификат CA «/etc/ssl/certs/ca-certificates.crt»
Распознаётся mirror.yandex.ru (mirror.yandex.ru)... 213.180.204.183
Подключение к mirror.yandex.ru (mirror.yandex.ru)|213.180.204.183|:443... с
оединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: 4762707968 (4,4G) [application/octet-stream]
Сохранение в: «ubuntu-22.04.5-desktop-amd64.iso»

ubuntu-22.04.5-des 100%[=====>] 4,44G 9,79MB/s за 8m 53s

2025-05-11 00:12:32 (8,52 MB/s) - «ubuntu-22.04.5-desktop-amd64.iso» сохрaн
ён [4762707968/4762707968]
```

Рисунок 7 — скачал образ ubuntu 22.04 с зеркала yandex

```
qemu-system-x86_64 \
-enable-kvm \
-cpu host \
-smp 1 \
-m 512M \
-vga std \
-hda "/tmp/disk_${USER}.qcow2" \
-cdrom /var/qemu/OS/ubuntu-22.04.5-desktop-amd64.iso \
-net user,hostfwd=tcp::8080-:80 \
-boot menu=on \
-serial none \
-monitor telnet:127.0.0.1:10023,server,nowait
qemu-system-x86_64: warning: hub 0 with no nics
```

QEMU

Machine View

GNU GRUB version 2.06

```
Try or Install Ubuntu
Ubuntu (safe graphics)
*OEM install (for manufacturers)
Test memory
```

Use the ↑ and ↓ keys to select which entry is highlighted.  
Press enter to boot the selected OS, 'e' to edit the commands  
before booting or 'c' for a command-line.

Рисунок 8 — запуск эмулятора qemu с параметрами из ТЗ

```

/var/qemu/OS telnet 127.0.0.1 10023
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
QEMU 9.2.3 monitor - type 'help' for more information
(qemu) info cpus
* CPU #0: thread_id=30971
(qemu) savevm running_stage
info registers
unknown command: 'info'
(qemu) info regusters
unknown command: 'info regusters'
(qemu) info registers

CPU#0
RAX=000000003914af7d RBX=0000000000000027 RCX=0000000000000000 RDX=000000000000004d
RSI=0000000000000000 RDI=00000004d39074833 RBP=ffff9c8ac0013e78 RSP=ffff9c8ac0013e78
R8 =0000000000000000 R9 =000000000001b774d R10=0000000000000000 R11=0000000000000000
R12=0000000000001d844 R13=00000000000000001 R14=000000000001d8a8 R15=0000000000000000
RIP=ffffffff8b986750 RFL=00000246 [---Z-P-] CPL=0 II=0 A20=1 SMM=0 HLT=0
ES =0000 0000000000000000 ffffffff 00c00100
CS =0010 0000000000000000 ffffffff 00a09b00 DPL=0 CS64 [-RA]
SS =0000 0000000000000000 ffffffff 00c00100
DS =0000 0000000000000000 ffffffff 00c00100
FS =0000 0000000000000000 ffffffff 00c00100
GS =0000 fffff8d3bdf40000 ffffffff 00c00100
LDT=0000 0000000000000000 ffffffff 00c00000
TR =0040 fffffe43cad4a000 00004087 00008b00 DPL=0 TSS64-busy
GDT= fffffe43cad48000 0000007f
IDT= fffffe0000000000 00000fff
CR0=80050033 CR2=ffff8d3bd4001000 CR3=000000001323c001 CR4=00370ef0
DR0=0000000000000000 DR1=0000000000000000 DR2=0000000000000000 DR3=0000000000000000
DR6=00000000ffff0fff DR7=00000000000000400
EFER=00000000000000d01
FCW=037f FSW=0000 [ST=0] FTW=00 MXCSR=00001f80
FPR0=0000000000000000 0000 FPR1=0000000000000000 0000
FPR2=0000000000000000 0000 FPR3=0000000000000000 0000
FPR4=0000000000000000 0000 FPR5=0000000000000000 0000
FPR6=0000000000000000 0000 FPR7=0000000000000000 0000
YMM00=0000000000000000 0000000000000000 89775500fb384c3b 882902f60cc1817d
YMM01=0000000000000000 0000000000000000 30111ee1084bd815 df841e43005bb23a
YMM02=0000000000000000 0000000000000000 109120a87839d4f5 88cf47ca0e48e382
YMM03=0000000000000000 0000000000000000 0163e72f86141075 8d8865b7d6654968
YMM04=0000000000000000 0000000000000000 0000000000000000 0000000000000000
YMM05=0000000000000000 0000000000000000 0000000000000000 0000000000000000
YMM06=0000000000000000 0000000000000000 0000000000000000 0000000000000000
YMM07=0000000000000000 0000000000000000 0000000000000000 00000000a7c5796e
YMM08=0000000000000000 0000000000000000 480000008a000000 e200000052000000

```

Рисунок 9 — подключение к монитору виртуальной машины по протоколу telnet и вывод информации о процессоре и его регистрах



```

(qemu) info cpus
* CPU #0: thread_id=30971
(qemu) info network
hub 0
  \ hub0port0: #net019: index=0,type=user,net=10.0.2.0,restrict=off
(qemu) info blocks
unknown command: 'info blocks'
(qemu) info block
ide0-hd0 (#block134): /tmp/disk_vicotr.qcow2 (qcow2)
  Attached to:      /machine/unattached/device[5]
  Cache mode:      writeback
  Backing file:     /tmp/disk_base_vicotr.qcow2 (chain depth: 1)

ide1-cd0 (#block554): /var/qemu/OS/ubuntu-22.04.5-desktop-amd64.iso (raw, read-only)
  Attached to:      /machine/unattached/device[6]
  Removable device: not locked, tray closed
  Cache mode:      writeback

floppy0: [not inserted]
  Attached to:      /machine/unattached/device[15]
  Removable device: not locked, tray closed

sd0: [not inserted]
  Removable device: not locked, tray closed
(qemu) █

```

Рисунок 10- вывод информации о блочных устройствах и сети виртуальной машины

## ВТОРАЯ ЧАСТЬ ЗАДАНИЯ:

### Б) Auditd

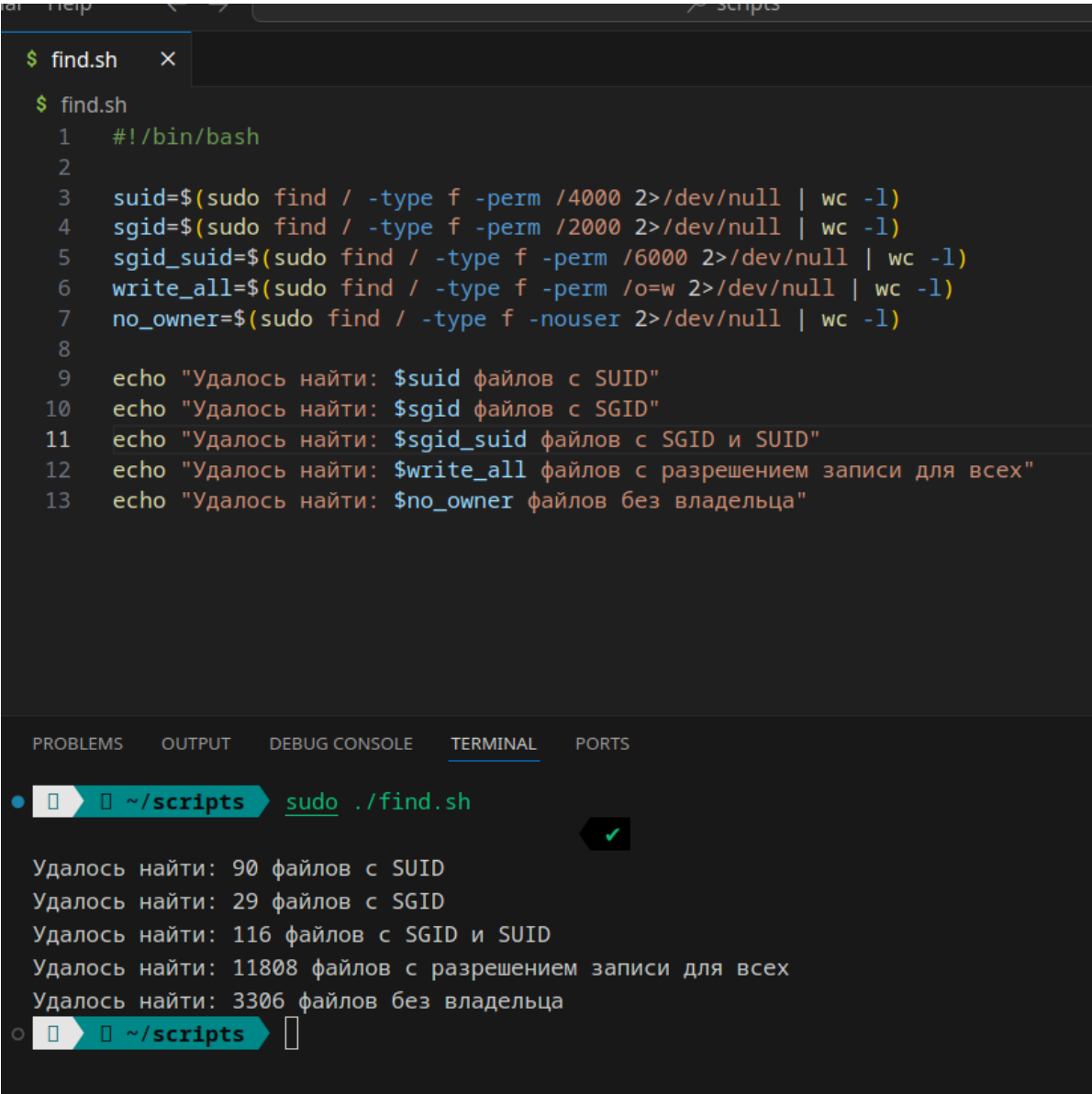
1. Узнайте список всех пользователей Linux
2. Получите вывод только имён пользователей в системе
3. Узнайте список всех подключенных пользователей к системе в данный момент времени

```
cat /etc/passwd | wc -l
43
cut -d: -f1 /etc/passwd
root
nobody
dbus
bin
daemon
mail
ftp
http
systemd-coredump
systemd-network
systemd-oom
systemd-journal-remote
systemd-resolve
systemd-timesync
tss
uidd
alpm
dnsmasq
_talkd
polkitd
rpc
rpcuser
avahi
git
nm-openconnect
nm-openvpn
ntp
openvpn
rtkit
sddm
vicotr
colord
gdm
geoclue
brltty
flatpak
gnome-remote-desktop
libvirt-qemu
qemu
saned
usbmux
www
gluster
who
vicotr tty2 2025-05-11 15:16 (:0)
vicotr pts/0 2025-05-11 15:16 (:0)
vicotr pts/1 2025-05-11 15:16 (:0)
vicotr pts/2 2025-05-11 15:16 (tmux(1969).%0)
```

Рисунок 11 — выполнение задания 1,2,3




4. С помощью команды `find` найдите в корневом каталоге файлы:

- имеющие атрибуты SUID;
- имеющие атрибуты SGID;
- имеющие атрибуты SGID и SUID;
- файлы, которые разрешено модифицировать всем;
- файлы, не имеющие владельца



```
$ find.sh
$ find.sh
1  #!/bin/bash
2
3  suid=$(sudo find / -type f -perm /4000 2>/dev/null | wc -l)
4  sgid=$(sudo find / -type f -perm /2000 2>/dev/null | wc -l)
5  sgid_suid=$(sudo find / -type f -perm /6000 2>/dev/null | wc -l)
6  write_all=$(sudo find / -type f -perm /o=w 2>/dev/null | wc -l)
7  no_owner=$(sudo find / -type f -nouser 2>/dev/null | wc -l)
8
9  echo "Удалось найти: $suid файлов с SUID"
10 echo "Удалось найти: $sgid файлов с SGID"
11 echo "Удалось найти: $sgid_suid файлов с SGID и SUID"
12 echo "Удалось найти: $write_all файлов с разрешением записи для всех"
13 echo "Удалось найти: $no_owner файлов без владельца"
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

•   ~/scripts `sudo ./find.sh` 

Удалось найти: 90 файлов с SUID  
 Удалось найти: 29 файлов с SGID  
 Удалось найти: 116 файлов с SGID и SUID  
 Удалось найти: 11808 файлов с разрешением записи для всех  
 Удалось найти: 3306 файлов без владельца




○   ~/scripts 

Рисунок 12 – скрипт для поиска файлов с различными правами в системе

5. С помощью команды `id user_name` посмотрите список основной и дополнительных групп пользователей. Найдите дополнительные группы `floppy`, `cdrom` и `plugdev`, дающие право использовать сменные машинные носители `/etc/cdrom`, `/etc/fd0` и т.д. для бесконтрольного блочного копирования данных.

```

~/.scripts$ id -gn
vicotr
~/.scripts$ id
uid=1000(vicotr) gid=1000(vicotr) группы=1000(vicotr),3(sys),90(network),98(power),953(docker),960(libvirt),986(uucp),991(lp),998(wheel)
~/.scripts$ cat /etc/group | grep -E 'floppy|cdrom|plugdev'
floppy:x:94:

```

Рисунок 13 - выполнение задания на работу с группами

Зарегистрируйте нового пользователя и добавьте его в разные группы, выведите список существующих пользователей и группы, проверьте наличие нового пользователя

```

~/.scripts$ sudo useradd -m -G docker,www,ttty newuser
~/.scripts$ sudo passwd newuser
Новый пароль:
Повторите ввод нового пароля:
passwd: пароль успешно обновлён
~/.scripts$ sudo usermod -aG games,ftp newuser
~/.scripts$ cat /etc/passwd | grep newuser
newuser:x:1001:1001::/home/newuser:/usr/bin/bash
~/.scripts$ groups newuser
newuser : newuser tty ftp games docker www
~/.scripts$ cat /etc/group | grep newuser
tty:x:5:brltty,newuser
ftp:x:11:newuser
games:x:50:newuser
docker:x:953:vicotr,newuser
www:x:952:newuser
newuser:x:1001:

```

Рисунок 14 – выполнение заданий на работу с пользователем/группой

7. С помощью команды `md5sum` вычислите и запишите контрольную сумму для одного из файлов в каталоге `/home/`

```

~/scripts md5sum ~/.tmux.conf > checksum.txt
~/scripts cat checksum.txt
58ea0cf9f8106b0cca17834f1fd3f72 /home/vicotr/.tmux.conf

```

Рисунок 15 – выполнение задания 7

8. С помощью команды `md5sum` вычислите и запишите в файл контрольную сумму всех файлов в каталоге `/bin`.

```

bin_checksums.txt
1 b77fc17c383534d64016f074ca981dfd /bin/ddcmon
2 b566f2b5363fb0ee65fe8a2e5c8e9466 /bin/lvm
3 948946206b75137c573deb90de48b143 /bin/envvars
4 5f7b1975501efd267db88803a70d83d6 /bin/xdg-settings
5 667316a97bb8b89ff61e4f3eb63b4790 /bin/nf-ct-list
6 001614c2cb3defd35c00a293ae95ede9 /bin/setpriv
7 23756588166845236e426329b7609fe2 /bin/qemu-loongarch64
8 afbfaf8535229af3a64838e7f1df4232 /bin/appstream-util
9 62f5f941aa6a21282c5f029e835c799c /bin/genccode
10 3540b47fa15fd510c50496f9c9f7f920 /bin/g-lensfun-update-data
11 a83f9594f762091d2302e571d22b5916 /bin/di-edid-decode
12 25f552e876c1ac51b8e535d61f9e5504 /bin/qmlimportscanner
13 9697e4425a4479144f82cf4faabc85a8 /bin/guile-config
14 7d26abafd847a4cf239ffeee3898e49d /bin/m4
15 534e10158625072ef866f2ee78c9e213 /bin/x86_64-pc-linux-gnu-gcc-ranlib
16 352bb112a09cb1bc5141a7eccf27aeb6 /bin/zforce
17 d44721b986e7db1e1d035f9a3f41792b /bin/virt-pki-query-dn
18 70e6504c3ef135227ba68b0f7b67dc76 /bin/ps2pdfwr
19 fe3ee3e4d63bff58364d7028b3fe4419 /bin/qemu-xtensaeb
20 a3e194959he406h4ehh603r453741 /bin/tv_device

~/scripts sudo find /bin/ -type f -print0 | xargs -0 md5sum > bin_checksums.txt
md5sum: /bin/mount.nfs: Отказано в доступе
md5sum: /bin/groupmems: Отказано в доступе

~/scripts

```

Рисунок 16 – выполнение задания 8

9. Снова с помощью команды `md5sum` вычислите и запишите в файл контрольную сумму всех файлов в каталоге `/bin` и добавьте какие-нибудь символы в конце файла, после сравните обе суммы

```

~/scripts sudo find /bin/ -type f -print0 | xargs -0 md5sum > bin_checksums_9task.txt
md5sum: /bin/mount.nfs: Отказано в доступе
md5sum: /bin/groupmems: Отказано в доступе

~/scripts echo "Gogolev VG 16 Laba example string" >> bin_checksums_9task.txt
~/scripts diff bin_checksums.txt bin_checksums_9task.txt
3161a3162
> Gogolev VG 16 Laba example string

```

Рисунок 17 – выполнение задания 9

```

~/scripts sudo find /usr/share -type f -name "**doc*" -exec cp {} /tmp/docs/ \;
~/scripts ls -la /tmp/docs
итого 4976
drwxr-xr-x  2 root root 11760 мая 11 17:16 .
drwxrwxrwt 29 root root   860 мая 11 17:16 ..
-rw-r--r--  1 root root 39676 мая 11 17:16 advanced.docbook
-rw-r--r--  1 root root 2185 мая 11 17:16 analog-input-dock-mic.conf
-rw-r--r--  1 root root 3037 мая 11 17:16 andoc.tmac
-rw-r--r--  1 root root 1244 мая 11 17:16 application-vnd.oasis.opendocument.chart.svg
-rw-r--r--  1 root root 1725 мая 11 17:16 application-vnd.oasis.opendocument.database.svg
-rw-r--r--  1 root root 1908 мая 11 17:16 application-vnd.oasis.opendocument.formula.svg
-rw-r--r--  1 root root 5305 мая 11 17:16 application-vnd.oasis.opendocument.formula-template.svg
-rw-r--r--  1 root root 4990 мая 11 17:16 application-vnd.oasis.opendocument.presentation-template.svg
-rw-r--r--  1 root root 5163 мая 11 17:16 application-vnd.oasis.opendocument.spreadsheet-template.svg
-rw-r--r--  1 root root 5040 мая 11 17:16 application-vnd.oasis.opendocument.text-template.svg
-rw-r--r--  1 root root 15249 мая 11 17:16 application-vnd.oasis.opendocument.web-template.svg

```

Рисунок 18 – выполнение задания на поиск файлов содержащих в названии “doc” и копирование в /tmp/docs

10. Найдите в папке /usr/share, включая подкаталоги, простые файлы “doc” и скопируйте найденное в папку /tmp/docs/

```

~/scripts sudo find /usr/share/ -type f -name "**doc*" -print0 | xargs -0 cp /tmp/docs
cp: цель '/usr/share/doc/HTML/et/kioworker6/man/index.docbook': Это не каталог
cp: цель '/usr/share/doc/qt6/config/exampleurl-qtspeech.qdocconf': Это не каталог
~/scripts sudo find /usr/share/ -type f -name "**doc*" -print0 | xargs -0 cp /tmp/docs/
cp: цель '/usr/share/doc/HTML/et/kioworker6/man/index.docbook': Это не каталог
cp: цель '/usr/share/doc/qt6/config/exampleurl-qtspeech.qdocconf': Это не каталог

```

Рисунок 19 – почему-то решение через -print0 -xargs -0 не работает((

11. Установите пакет auditd для мониторинга событий операционной системы и записи их в журналы событий
12. Просмотрите статус службы auditd
13. Запустите службу auditd
14. Выведите абсолютно все события аудита за день
15. Выведите результаты аудита по времени
16. Установите пакет figlet
17. Запустите figlet таким образом, чтобы на экране отобразилась ваша фамилия и группа

```

• [ ] [ ] ~/scripts sudo systemctl enable --now auditd.service
Created symlink '/etc/systemd/system/multi-user.target.wants/auditd.service' → '/usr/lib/systemd/system/auditd.service'.
• [ ] [ ] ~/scripts sudo systemctl status auditd.service
• auditd.service - Security Audit Logging Service
  Loaded: loaded (/usr/lib/systemd/system/auditd.service; enabled; preset: disabled)
  Active: active (running) since Sun 2025-05-11 17:22:26 MSK; 21s ago
  Invocation: e467c2dced44b32b5d9cf07cd8750d9
  Docs: https://github.com/linux-audit/audit-documentation (ctrl + click)
        https://github.com/linux-audit/audit-documentation
  Process: 41248 ExecStart=/usr/bin/auditd (code=exited, status=0/SUCCESS)
  Main PID: 41249 (auditd)
  Tasks: 2 (limit: 9233)
  Memory: 668K (peak: 1.6M)
  CPU: 17ms
  CGroup: /system.slice/auditd.service
          └─41249 /usr/bin/auditd

```

Рисунок 20 – запуск и проверка службы auditd.service

```

[ ] [ ] ~/scripts sudo ausearch -ts today | head
-----
time->Sun May 11 17:22:26 2025
type=DAEMON_START msg=audit(1746973346.661:131): op=start ver=4.0.3 format=enriched kernel=6.1
2.25-1-MANJARO auid=4294967295 pid=41249 uid=0 ses=4294967295 res=success
-----
time->Sun May 11 17:22:26 2025
type=BPF msg=audit(1746973346.661:14): prog-id=64 op=LOAD
-----
time->Sun May 11 17:22:26 2025
type=BPF msg=audit(1746973346.661:15): prog-id=31 op=UNLOAD
-----
[ ] [ ] ~/scripts sudo aureport -t
PIPE|0 ✓

Log Time Range Report
=====
/var/log/audit/audit.log: 11.05.2025 17:22:26.661 - 11.05.2025 17:27:25.033
[ ] [ ] ~/scripts [ ]

```

Рисунок 21 – вывод всех событий аудита за день и за время 5 минут

## Ausearch

Назначение: Поиск событий в логах аудита.

Основные ключи:

-m (или --message) — фильтр по типу события (например, LOGIN, USER\_AUTH, SYSCALL).

-k (или --key) — поиск по ключу аудита (например, ausearch -k my\_script).

-ui — поиск по UID пользователя.

-sc — поиск по системному вызову (например, open, execve).

-ts — фильтр по времени (-ts today, -ts "05/11/2025 17:22:26").

### Aureport

Назначение: Генерация отчетов на основе логов аудита.

Основные ключи:

-t (или --log-time) — показывает временной диапазон логов.

-u — отчет по действиям пользователей.

-l — отчет по событиям входа/выхода.



Рисунок 22 – запустил figlet так чтобы вывело мою фамилию и группу