

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

Файловые системы операционных систем

Отчет по лабораторной работе 6

по дисциплине «Системное программное обеспечение»

студента 3 курса группы ИВТ-б-о-202

Шор Константина Александровича

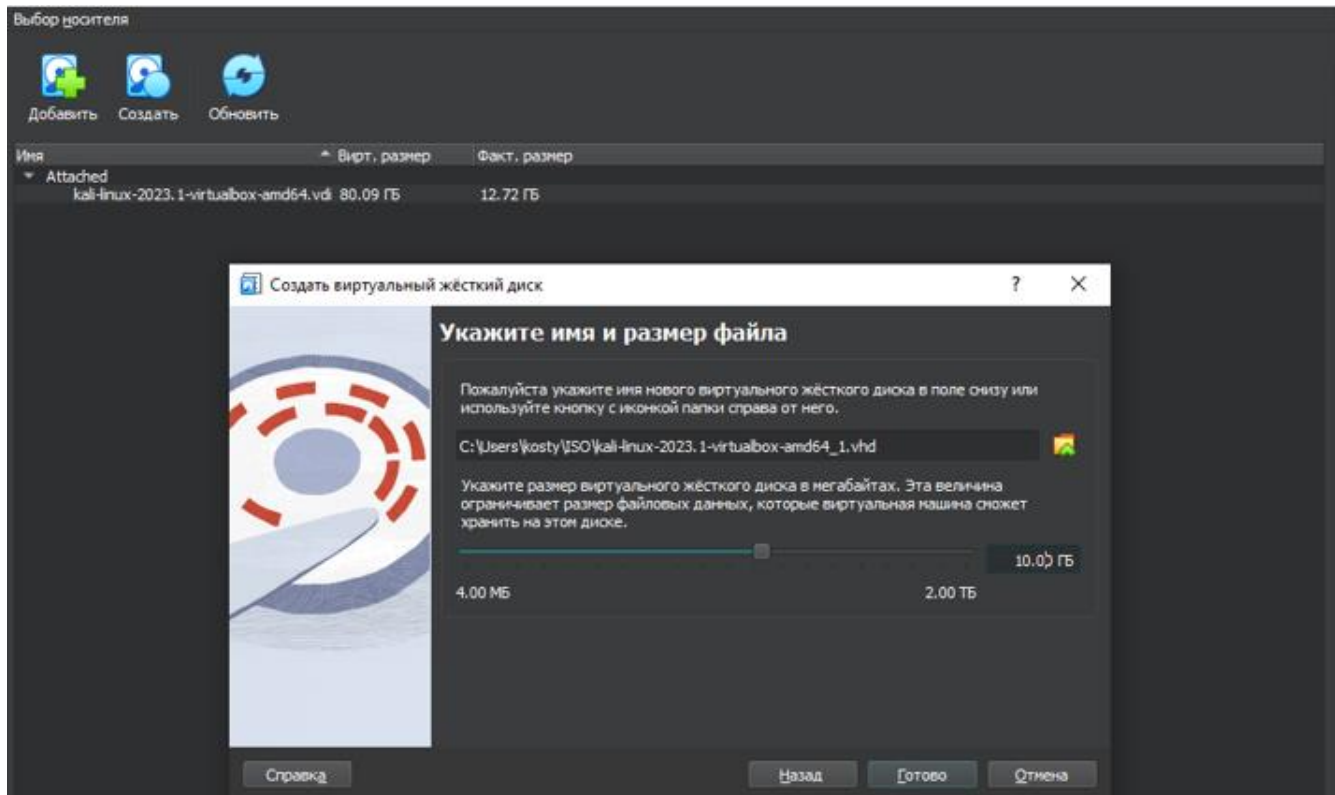
Направления подготовки 09.03.01 «Информатика и вычислительная техника»

Симферополь, 2023

Лабораторная работа №6. Файловые системы операционных систем

Цель работы: Получение навыков управления различными файловыми системами в среде операционной системы Linux.

СОЗДАНИЕ ВИРТУАЛЬНОГО ДИСКА



```
root@kali: /home/kali
File Actions Edit View Help
(root@kali)-[/home/kali]
# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda 8:0 0 80.1G 0 disk
└─sda1 8:1 0 80.1G 0 part /
sdb 8:16 0 10G 0 disk
sr0 11:0 1 1024M 0 rom

(root@kali)-[/home/kali]
# fdisk -l
Disk /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

РАЗМЕТКА ДИСКА

```
root@kali: /home/kali
File Actions Edit View Help

Disk: /dev/sdb
Size: 10 GiB, 10737418240 bytes, 20971520 sectors
Label: gpt, identifier: F30F16D8-28D2-C74C-9343-EFD58A6EBAE4

Device      Start      End      Sectors    Size Type
/dev/sdb1    2048       2099199  2097152    1G Linux filesystem
/dev/sdb2    2099200    4196351  2097152    1G Linux filesystem
/dev/sdb3    4196352    6293503  2097152    1G Linux filesystem
/dev/sdb4    6293504    8390655  2097152    1G Linux filesystem
/dev/sdb5    8390656    10487807 2097152    1G Linux filesystem
/dev/sdb6    10487808   12584959 2097152    1G Linux filesystem
/dev/sdb7    12584960   14682111 2097152    1G Linux filesystem
/dev/sdb8    14682112   16779263 2097152    1G Linux filesystem
/dev/sdb9    16779264   18876415 2097152    1G Linux filesystem
>> /dev/sdb10 18876416   20969471 2093056    1022M Linux swap

Partition UUID: CDE593A4-F8F7-4B42-BCDE-0ED8B404A73E
Partition type: Linux swap (0657FD6D-A4AB-43C4-84E5-0933C84B4F4F)

[ Delete ] [ Resize ] [ Quit ] [ Type ] [ Help ] [ Write ]
[ Dump ]
```

```
(root@kali)-[/home/kali]
# fdisk -l
Disk /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: F30F16D8-28D2-C74C-9343-EFD58A6EBAE4

Device      Start      End      Sectors    Size Type
/dev/sdb1    2048       2099199  2097152    1G Linux filesystem
/dev/sdb2    2099200    4196351  2097152    1G Linux filesystem
/dev/sdb3    4196352    6293503  2097152    1G Linux filesystem
/dev/sdb4    6293504    8390655  2097152    1G Linux filesystem
/dev/sdb5    8390656    10487807 2097152    1G Linux filesystem
/dev/sdb6    10487808   12584959 2097152    1G Linux filesystem
/dev/sdb7    12584960   14682111 2097152    1G Linux filesystem
/dev/sdb8    14682112   16779263 2097152    1G Linux filesystem
/dev/sdb9    16779264   18876415 2097152    1G Linux filesystem
/dev/sdb10   18876416   20969471 2093056    1022M Linux swap
```

```
(root@kali)-[/home/kali]
# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0     0 80.1G  0 disk
└─sda1       8:1     0 80.1G  0 part /
sdb          8:16     0   10G  0 disk
├─sdb1       8:17     0    1G  0 part
├─sdb2       8:18     0    1G  0 part
├─sdb3       8:19     0    1G  0 part
├─sdb4       8:20     0    1G  0 part
├─sdb5       8:21     0    1G  0 part
├─sdb6       8:22     0    1G  0 part
├─sdb7       8:23     0    1G  0 part
├─sdb8       8:24     0    1G  0 part
├─sdb9       8:25     0    1G  0 part
└─sdb10      8:26     0 1022M  0 part
sr0         11:0     1 1024M  0 rom

(root@kali)-[/home/kali]
#
```

СОЗДАНИЕ ФАЙЛОВОЙ СИСТЕМЫ

EXT2

```
(root@kali)-[/home/kali]
# mkfs.ext2 /dev/sdb1
mke2fs 1.46.6 (1-Feb-2023)
/dev/sdb1 contains a ext2 file system
    created on Mon May 22 09:30:23 2023
Proceed anyway? (y,N) y
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 764de8a8-79ed-4f35-9d9b-fc42d277d195
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
    "the quieter you become, the more you are able to hear"

(root@kali)-[/home/kali]
# tune2fs -l /dev/sdb1 | grep features
Filesystem features:      ext_attr resize_inode dir_index filetype sparse_sup
er large_file
```

Не поддерживает журналирование

EXT3

```
(root@kali)-[/home/kali]
# mkfs.ext3 /dev/sdb2 -j
mke2fs 1.46.6 (1-Feb-2023)
/dev/sdb2 contains a ext3 file system
    created on Mon May 22 09:24:22 2023
Proceed anyway? (y,N) y
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 6ccac6d4-7a58-43f8-8d2e-33ad1c8d6f63
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

"the quieter you become, the more you are able to hear"

(root@kali)-[/home/kali]
# tune2fs -l /dev/sdb2 | grep features
Filesystem features:      has_journal ext_attr resize_inode dir_index filetype
sparse_super large_file
```

По умолчанию

EXT4

```
5B Volume
(root@kali)-[/home/kali]
# mkfs.ext4 /dev/sdb3 -j
mke2fs 1.46.6 (1-Feb-2023)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 64aa513a-a423-4897-afdd-e4b6d6019558
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

(root@kali)-[/home/kali]
# tune2fs -l /dev/sdb3 | grep features
Filesystem features:      has_journal ext_attr resize_inode dir_index filetype
extent 64bit flex_bg sparse_super large_file huge_file dir_nlink extra_isize
metadata_csum
```

По умолчанию

XFS

```
(root@kali)-[/home/kali]
# mkfs.xfs /dev/sdb4
meta-data=/dev/sdb4          isize=512    agcount=4, agsize=65536 blks
                             =          sectsz=512   attr=2, projid32bit=1
                             =          crc=1        finobt=1, sparse=1, rmapbt=0
                             =          reflink=1     bigtime=1 inobtcount=1 nnext64=0
data      =                  bsize=4096   blocks=262144, imaxpct=25
                             =          sunit=0      swidth=0 blks
naming    =version 2         bsize=4096   ascii-ci=0, ftype=1
log        =internal log    bsize=4096   blocks=16384, version=2
                             =          sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none             extsz=4096   blocks=0, rtextents=0
```

По умолчанию

BTRFS

```
(root@kali)-[/home/kali]
# mkfs.btrfs /dev/sdb5 -f
btrfs-progs v6.2
See http://btrfs.wiki.kernel.org for more information.

NOTE: several default settings have changed in version 5.15, please make sure
this does not affect your deployments:
- DUP for metadata (-m dup)
- enabled no-holes (-O no-holes)
- enabled free-space-tree (-R free-space-tree)

Label: (null)
UUID: 302bfffac-4153-4e43-b152-73045f681cc7
Node size: 16384
Sector size: 4096
Filesystem size: 1.00GiB
Block group profiles:
  Data: single 8.00MiB
  Metadata: DUP 51.19MiB
  System: DUP 8.00MiB
SSD detected: no
Zoned device: no
Incompat features: extref, skinny-metadata, no-holes
Runtime features: free-space-tree
Checksum: crc32c
Number of devices: 1
Devices:
  ID      SIZE  PATH
  1      1.00GiB /dev/sdb5

1.1 GB Volume

(root@kali)-[/home/kali]
# btrfs filesystem show /dev/sdb5
Label: none  uuid: 302bfffac-4153-4e43-b152-73045f681cc7
      Total devices 1 FS bytes used 144.00KiB
      devid  1 size 1.00GiB used 126.38MiB path /dev/sdb5
```

По умолчанию

ZFS

```
(root@kali)-[/mnt]
# zpool create -f pool1 /dev/sdb6
Defaulting to 4K blocksize (ashift=12) for '/dev/sdb6'
invalid vdev specification
the following errors must be manually repaired:
/dev/sdb6 is part of active pool 'pool1'

(root@kali)-[/mnt]
# zpool scrub pool1

(root@kali)-[/mnt]
# zpool list
NAME      SIZE  ALLOC   FREE      CAP  DEDUP  HEALTH  ALTROOT
pool1    1016M   964K  1015M       0%  1.00x  ONLINE   -

(root@kali)-[/mnt]
# zfs create pool1/files

(root@kali)-[/mnt]
# zfs create pool1/media

(root@kali)-[/mnt]
# zpool list
NAME      SIZE  ALLOC   FREE      CAP  DEDUP  HEALTH  ALTROOT
pool1    1016M   1.22M  1015M       0%  1.00x  ONLINE   -

(root@kali)-[/mnt]
# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
pool1                984K   983M   128K   /pool1
pool1/data           112K   983M   112K   /pool1/data
pool1/files          112K   983M   112K   /pool1/files
pool1/media          112K   983M   112K   /pool1/media
pool1/mediaclear     112K   983M   112K   /pool1/mediaclear
```

По умолчанию

REISERS

```
(root@kali)-[/home/kali/OS]
# mkfs.reiserfs /dev/sdb7
mkfs.reiserfs 3.6.27

Guessing about desired format.. Kernel 6.1.0-kali5-amd64 is running.
Format 3.6 with standard journal
Count of blocks on the device: 262144
Number of blocks consumed by mkreiserfs formatting process: 8219
Blocksize: 4096
Hash function used to sort names: "r5"
Journal Size 8193 blocks (first block 18)
Journal Max transaction length 1024
inode generation number: 0
UUID: 528db063-aa85-440c-a27a-d1a4d2e18e92
ATTENTION: YOU SHOULD REBOOT AFTER FDISK!
          ALL DATA WILL BE LOST ON '/dev/sdb7'!
Continue (y/n):y
Initializing journal - 0%....20%....40%....60%....80%....100%
Syncing..ok
ReiserFS is successfully created on /dev/sdb7.
```

По умолчанию

FAT32

```
(root@kali)-[/home/kali/OS]
# mkfs.vfat /dev/sdb8
mkfs.fat 4.2 (2021-01-31)
```

Нет

NTFS

```
(root@kali)-[/home/kali/OS]
# mkfs.ntfs /dev/sdb9
Cluster size has been automatically set to 4096 bytes.
Initializing device with zeroes: 100% - Done.
Creating NTFS volume structures.
mkntfs completed successfully. Have a nice day.
```

По умолчанию

Раздел подкачки

```
(root@kali)-[/home/kali/OS]
# mkswap /dev/sdb10
Setting up swapspace version 1, size = 1022 MiB (1071640576 bytes)
no label, UUID=e0e69278-c129-4d9c-93f7-eee278880c90

(root@kali)-[/home/kali/OS]
# swapon -s
Filename      1.1 GB Volume      Type      Size      Used      Priority
/swapfile     file      1048572    247580    -2

(root@kali)-[/home/kali/OS]
# swapon -U e0e69278-c129-4d9c-93f7-eee278880c90

(root@kali)-[/home/kali/OS]
# swapon -s
Filename      1.1 GB Volume      Type      Size      Used      Priority
/swapfile     file      1048572    247324    -2
/dev/sdb10    partition 1046524    0         -3
```

МОНТИРОВАНИЕ

```
(root@kali)-[/mnt]
# mount /dev/sdb1 /mnt/ext2

(root@kali)-[/mnt]
# mount /dev/sdb2 /mnt/ext3

(root@kali)-[/mnt]
# mount /dev/sdb3 /mnt/ext4

(root@kali)-[/mnt]
# mount /dev/sdb4 /mnt/xfs

(root@kali)-[/mnt]
# mount /dev/sdb5 /mnt/btrfs

(root@kali)-[/mnt]
# mount /dev/sdb6 /mnt/zfs
mount: /mnt/zfs: unknown filesystem type 'zfs_member'.
dmesg(1) may have more information after failed mount system call.

(root@kali)-[/mnt]
# mount /dev/sdb7 /mnt/reiserfs

(root@kali)-[/mnt]
# mount /dev/sdb8 /mnt/fat32

(root@kali)-[/mnt]
# mount /dev/sdb9 /mnt/ntfs
```

```
(root@kali)-[/mnt]
# zfs mount

(root@kali)-[/mnt]
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
pool1	984K	983M	128K	/pool1
pool1/data	112K	983M	112K	/pool1/data
pool1/files	112K	983M	112K	/pool1/files
pool1/media	112K	983M	112K	/pool1/media
pool1/mediaclear	112K	983M	112K	/pool1/mediaclear

```
(root@kali)-[/mnt]
#
```

```
(root@kali)-[/mnt]
```

```
# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sda	8:0	0	80.1G	0	disk	
└─sda1	8:1	0	80.1G	0	part	/
sdb	8:16	0	10G	0	disk	
├─sdb1	8:17	0	1G	0	part	/mnt/ext2
├─sdb2	8:18	0	1G	0	part	/mnt/ext3
├─sdb3	8:19	0	1G	0	part	/mnt/ext4
├─sdb4	8:20	0	1G	0	part	/mnt/xfs
├─sdb5	8:21	0	1G	0	part	/mnt/btrfs
├─sdb6	8:22	0	1G	0	part	
├─sdb7	8:23	0	1G	0	part	/mnt/reiserfs
├─sdb8	8:24	0	1G	0	part	/mnt/fat32
├─sdb9	8:25	0	1G	0	part	/mnt/ntfs
└─sdb10	8:26	0	1022M	0	part	[SWAP]
sr0	11:0	1	1024M	0	rom	

Перезагрузили систему и вывели перемонтированные файловые системы

```
└─# mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=968048k,nr_inodes=242012,mode=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=201460k,mode=755,inode64)
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,inode64)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=15602)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
ramfs on /run/credentials/systemd-sysctl.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
ramfs on /run/credentials/systemd-sysusers.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
ramfs on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
/dev/sdb1 on /mnt/ext2 type ext2 (rw,relatime)
/dev/sdb2 on /mnt/ext3 type ext3 (rw,relatime)
/dev/sdb3 on /mnt/ext4 type ext4 (rw,relatime)
/dev/sdb9 on /mnt/ntfs type fuseblk (rw,relatime,user_id=0,group_id=0,allow_other,blksize=4096)
/dev/sdb5 on /mnt/btrfs type btrfs (rw,relatime,space_cache=v2,subvolid=5,subvol=/)
/dev/sdb7 on /mnt/reiserfs type reiserfs (rw,relatime)
/dev/sdb8 on /mnt/fat32 type vfat (rw,relatime,fmask=0022,dmask=0022,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=remount-ro)
/dev/sdb4 on /mnt/xfs type xfs (rw,relatime,attr2,inode64,logbufs=8,logbsize=32k,noquota)
ramfs on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,nosuid,nodev,noexec,relatime)
sunrpc on /run/rpc_pipefs type rpc_pipefs (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=201456k,nr_inodes=50364,mode=700,uid=1000,gid=1000,inode64)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)

(root@kali)-[/home/kali]
└─# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0 80.1G  0 disk
└─sda1       8:1    0 80.1G  0 part /
sdb          8:16   0   10G  0 disk
├─sdb1       8:17   0    1G  0 part /mnt/ext2
├─sdb2       8:18   0    1G  0 part /mnt/ext3
├─sdb3       8:19   0    1G  0 part /mnt/ext4
├─sdb4       8:20   0    1G  0 part /mnt/xfs
├─sdb5       8:21   0    1G  0 part /mnt/btrfs
├─sdb6       8:22   0    1G  0 part
├─sdb7       8:23   0    1G  0 part /mnt/reiserfs
├─sdb8       8:24   0    1G  0 part /mnt/fat32
├─sdb9       8:25   0    1G  0 part /mnt/ntfs
└─sdb10      8:26   0 1022M  0 part [SWAP]
sr0         11:0    1 1024M  0 rom
```

Свап файл

```
(root@kali)-[/home/kali/OS]
# dd if=/dev/zero of=/home/kali/OS/swap bs=100M count=5
5+0 records in
5+0 records out
524288000 bytes (524 MB, 500 MiB) copied, 0.812716 s, 645 MB/s

(root@kali)-[/home/kali/OS]
# mkswap /home/kali/OS/swap
mkswap: /home/kali/OS/swap: insecure permissions 0644, fix with: chmod 0600 /home/kali/OS/swap
Setting up swapspace version 1, size = 500 MiB (524283904 bytes)
no label, UUID=a188600a-0244-45dd-af93-dcdf64608f8

(root@kali)-[/home/kali/OS]
# swapon /home/kali/OS/swap
swapon: /home/kali/OS/swap: insecure permissions 0644, 0600 suggested.

(root@kali)-[/home/kali/OS]
# swapon -s
"the quieter you become, the more you are able to hear"


| Filename           | Type      | Size    | Used | Priority |
|--------------------|-----------|---------|------|----------|
| /swapfile          | file      | 1048572 | 0    | -2       |
| /dev/sdb10         | partition | 1046524 | 0    | -3       |
| /home/kali/OS/swap | file      | 511996  | 0    | -4       |



(root@kali)-[/home/kali/OS]
#
```

СКРИПТЫ

1. 15 Протестировать все разделы для операций чтения маленьких файлов (16кб), цикл не менее 100 раз

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_15.sh
#!/bin/bash

fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xfs"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

file_size="16k"
count_reads=100

echo "" > task_15.log

for partition in "${fail_system[@]}; do
    dd if=/dev/zero of="$partition/testfile" bs="$file_size" count=1

    start_time=$(date +%s.%6N)
    dd if="$partition/testfile" of=/dev/null bs="$file_size" count=100
    end_time=$(date +%s.%6N)

    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_15.log

    rm "$partition/testfile"
done
```

```
(root@kali)-[/home/kali/OS/Laba_7]
# cat task_15.log

/mnt/ext2 took .002101 mks
/mnt/ext3 took .002678 mks
/mnt/ext4 took .001807 mks
/mnt/xfs took .001803 mks
/mnt/btrfs took .001775 mks
/mnt/reiserfs took .001772 mks
/mnt/fat32 took .001863 mks
/mnt/ntfs took .002001 mks
```

2. 16Протестировать все файловые системы для операций чтения больших файлов

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_16.sh
#!/bin/bash

fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xfs"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

file_size="500M"

echo "" > task_16.log

for partition in "${fail_system[@]}; do
    dd if=/dev/zero of="$partition/testfile" bs="$file_size" count=1

    start_time=$(date +%s.%6N)
    dd if="$partition/testfile" of=/dev/null bs="$file_size" count=100
    end_time=$(date +%s.%6N)

    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_16.log

    rm "$partition/testfile"
done
```

```
(root@kali)-[/home/kali/OS/Laba_7]
# cat task_16.log

/mnt/ext2 took .180637 mks
/mnt/ext3 took .179480 mks
/mnt/ext4 took .140054 mks
/mnt/xfs took .175682 mks
/mnt/btrfs took .219337 mks
/mnt/reiserfs took .219066 mks
/mnt/fat32 took .185229 mks
/mnt/ntfs took .319330 mks
```

3. 17Протестировать операции записи для маленьких файлов

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_17.sh *
#!/bin/bash

fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xfs"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

file_size="16k"

echo "" > task_17.log

for partition in "${fail_system[@]}; do
    start_time=$(date +%s.%6N)
    dd if=/dev/zero of="$partition/testfile" bs="$file_size" count=100
    end_time=$(date +%s.%6N)
    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_17.log
    rm "$partition/testfile"
done
```

```
(root@kali)-[/home/kali/OS/Laba_7]
# cat task_17.log

/mnt/ext2 took .004187 mks
/mnt/ext3 took .004632 mks
/mnt/ext4 took .003327 mks
/mnt/xfs took .002662 mks
/mnt/btrfs took .002889 mks
/mnt/reiserfs took .003964 mks
/mnt/fat32 took .005763 mks
/mnt/ntfs took .025588 mks
```

4. 18Протестировать операции записи для больших файлов

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_18.sh *
fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xfs"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

file_size="500M"

echo "" > task_18.log

for partition in "${fail_system[@]}; do

    start_time=$(date +%s.%6N)
    dd if=/dev/zero of="$partition/testfile" bs="$file_size" count=100
    end_time=$(date +%s.%6N)

    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_18.log

    rm "$partition/testfile"

done
```

```
(root@kali)-[/home/kali/OS/Laba_7]
# cat task_18.log

/mnt/ext2 took 1.714617 mks
/mnt/ext3 took 1.553655 mks
/mnt/ext4 took 1.575909 mks
/mnt/xfs took 1.562939 mks
/mnt/btrfs took 1.582234 mks
/mnt/reiserfs took 2.691580 mks
/mnt/fat32 took 3.166142 mks
/mnt/ntfs took 13.382168 mks
```


5. 19Создать структуру каталогов с не менее чем 1000 подкаталогов в каждой файловой системе, замерить время создания.

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_19.sh
#!/bin/bash

fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xfs"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

echo "" > task_19.log

for partition in "${fail_system[@]"; do

    start_time=$(date +%s.%6N)
    for i in {0000..1000}; do
        mkdir -p "$partition/dir/$i"
    done
    end_time=$(date +%s.%6N)
    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_19.log

    rm -r "$partition/dir"
done
```

```
(root@kali)-[/home/kali/OS/Laba_7]
# cat task_19.log

/mnt/ext2 took 1.018321 mks
/mnt/ext3 took 1.050899 mks
/mnt/ext4 took 1.050512 mks
/mnt/xfs took 1.022357 mks
/mnt/btrfs took .991985 mks
/mnt/reiserfs took .978087 mks
/mnt/fat32 took 1.137906 mks
/mnt/ntfs took 1.314042 mks
```

6. 203 Замерить время поиска по созданной структуре для каждой файловой системы

```
root@kali: /home/kali/OS/Laba_7
File Actions Edit View Help
GNU nano 7.2 task_20.sh
#!/bin/bash

fail_system=(
    "/mnt/ext2"
    "/mnt/ext3"
    "/mnt/ext4"
    "/mnt/xf"
    "/mnt/btrfs"
    "/mnt/reiserfs"
    "/mnt/fat32"
    "/mnt/ntfs"
)

echo "" > task_20.log

for partition in "${fail_system[@]"; do

    for i in {0000..1000}; do
        mkdir -p "$partition/dirs/dir${i}"
    done

    start_time=$(date +%s.%6N)
    find "$partition/dirs" -type d -name "dir1000"
    end_time=$(date +%s.%6N)

    result_time=$(echo "$end_time - $start_time" | bc)
    echo "$partition took $result_time mks" >> task_20.log

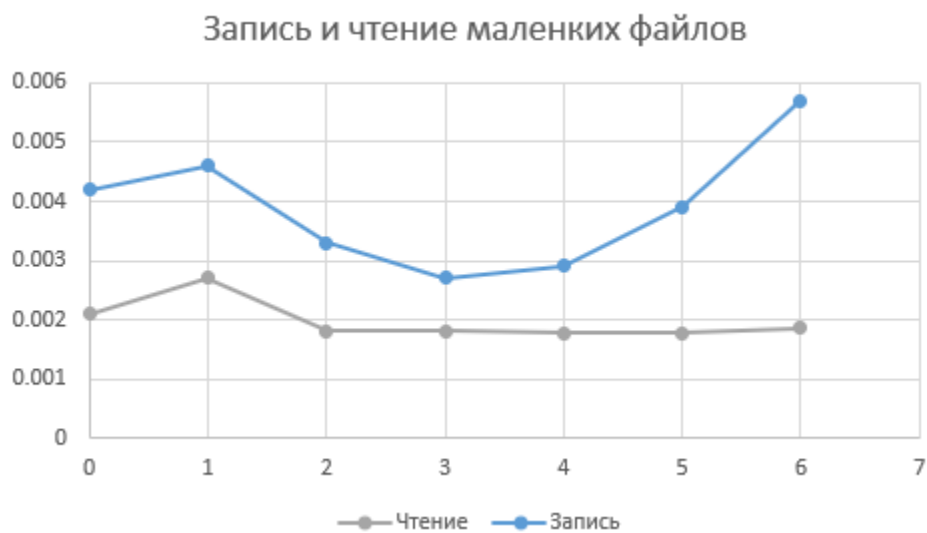
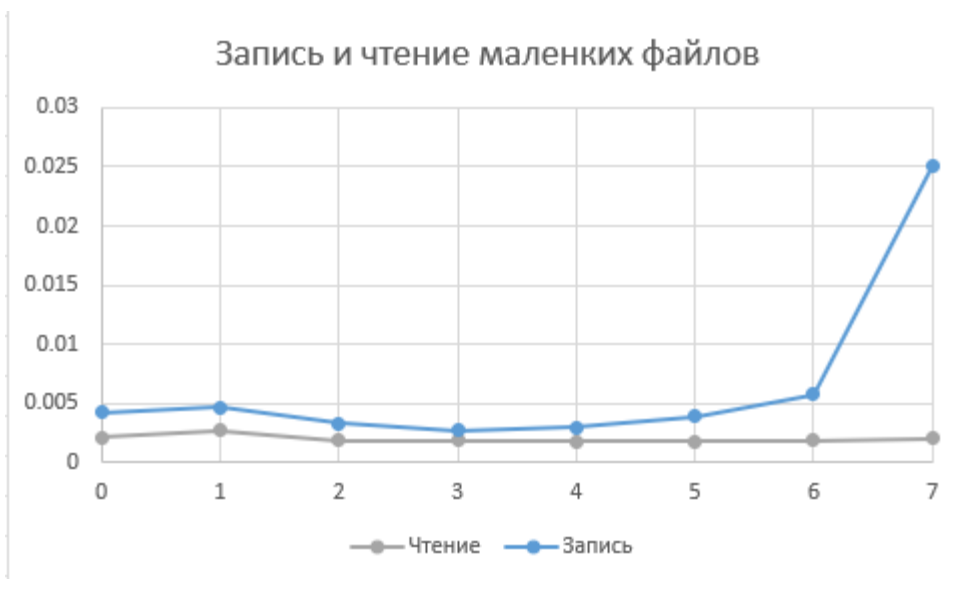
    rm -r "$partition/dirs"
done
```

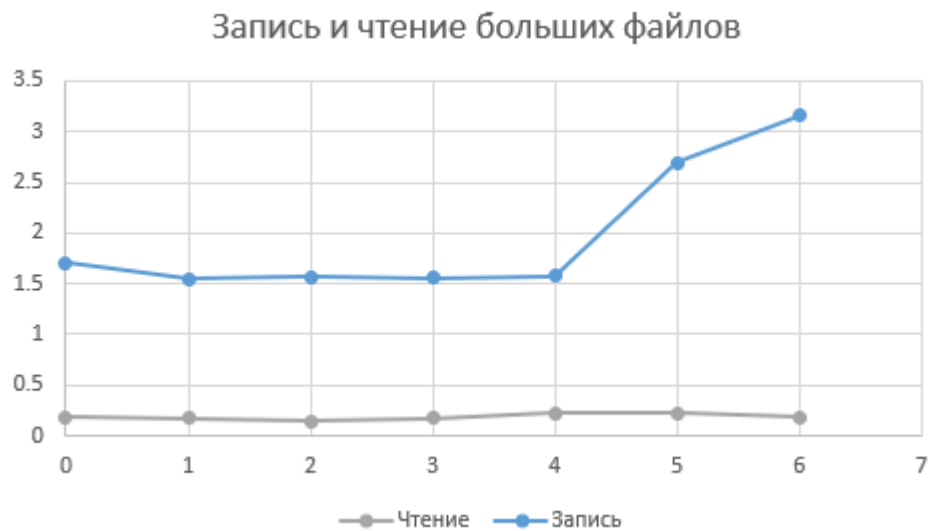
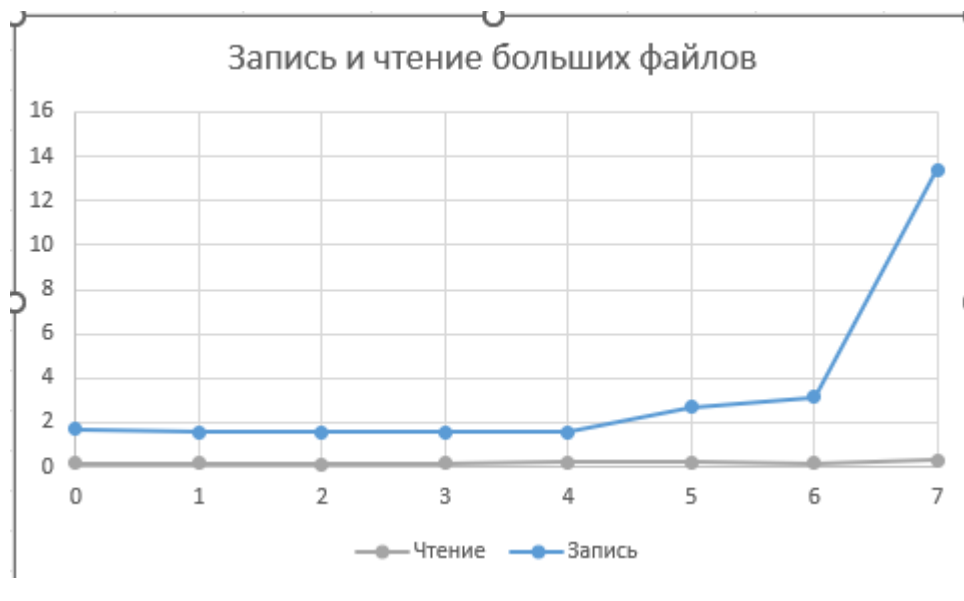
```
(root@kali)-[/home/kali/OS/Laba_7]
# ./task_20.sh
/mnt/ext2/dirs/dir1000
/mnt/ext3/dirs/dir1000
/mnt/ext4/dirs/dir1000
/mnt/xf/dirs/dir1000
/mnt/btrfs/dirs/dir1000
/mnt/reiserfs/dirs/dir1000
/mnt/fat32/dirs/dir1000
/mnt/ntfs/dirs/dir1000

(root@kali)-[/home/kali/OS/Laba_7]
# cat task_20.log

/mnt/ext2 took .011792 mks
/mnt/ext3 took .012039 mks
/mnt/ext4 took .012764 mks
/mnt/xf took .011751 mks
/mnt/btrfs took .012871 mks
/mnt/reiserfs took .010239 mks
/mnt/fat32 took .012908 mks
/mnt/ntfs took .236815 mks
```

21. Представить результаты каждого эксперимента по замеру производительности операции в файловой системе в виде графика





Вывод: В ходе данной лабораторной работе я создал виртуальный диск, на котором потом произвёл разметку диска с помощью cfdisk. Затем произвёл создания файловых систем: ext2, ext3, ext4, xfs, btrfs, zfs, fat32 и ntfs. После создания файловых систем – смонтировал их. Настроил автоматическое монтирование при запуске системы в fstab. Также создал и подключил файл подкачки. Написал на bash скрипты, которые тестируют запись и чтение маленьких и больших файлов. И скрипт, который создаёт и производит поиск 1000 каталогов. Измерил время выполнения всех скриптов на разных файловых системах и предоставил графики для отображения статистики.