

ПРАКТИЧЕСКАЯ РАБОТА № 6

РАБОТА В СРЕДЕ SHELL с командой Chroot и dd

Целью данной работы является получение практических навыков и умений по работе с утилитой управления процессом монтирования корневого каталога **Chroot** и копирования бинарных потоков данных **dd**.

Теоретические сведения по работе с утилитой Chroot

Изменение корня это процесс изменения видимой на диске корневой директории (и текущего запуска процессов) на другую корневую директорию. Когда вы изменили корневую директорию на другую, вы больше не имеете доступа к файлам и командам за пределами этой директории. Эта директория становится подобна заключению пользователя в клетку. Изменение корневой директории обычно нужно для таких задач как переустановка GRUB или сброс забытого пароля и чаще осуществляется при помощи LiveCD или LiveUSB в монтируемый раздел, содержащий установленную систему.

Вы должны загрузиться с любой работающей среды Linux (например, с LiveCD или USB flash disk). Для работы с chroot потребуются привилегии суперпользователя.

Убедитесь, что архитектура среды Linux, с которой вы загрузились соответствует архитектуре системы, с которой вы будете работать (т.е. i686, x86_64). Вы можете просмотреть архитектуру среды командой: `uname -m`

Монтирование каталогов

Чтобы просмотреть все имеющиеся дисковые разделы и их типы введите:

```
# fdisk -l
```

Создайте директорию, куда вы хотели бы подмонтировать устройство или раздел:

```
# mkdir /mnt/arch
```

```
# mount /dev/sda3 /mnt/arch
```

Изменение корневой директории

Подмонтируйте временные файловые системы:

```
# mount -t proc none /mnt/arch/proc
```

```
# mount -t sysfs sys /mnt/arch/sys
```

```
# mount -o bind /dev /mnt/arch/dev
```

Монтируйте другие разделы, если нуждаетесь в них (такие как /boot, /var, /usr). Например:

```
# mount /dev/sda1 /mnt/arch/boot
```

После выхода из изолированной среды, вы сможете отмонтировать все разделы одной командой. Это позволит безопасно отключить систему.

Если при нахождении в среде chroot вам нужна будет работа в сети, скопируйте информацию о DNS в новый корневой каталог:

```
# cp -L /etc/resolv.conf etc/resolv.conf
```

Переходим в новую среду:

```
# chroot /mnt/arch /bin/bash
```

Если вы увидите ошибку "chroot: cannot run command '/bin/bash': Exec format error" это может означать, что архитектуры не совпадают.

При работе с GRUB в изолированной среде нужно быть уверенным, что каталог /etc/mtab содержит актуальную информацию:

```
# grep -v rootfs /proc/mounts > /etc/mtab
```

Следующие шаги:

```
# source /etc/profile
# export PS1="(chroot) $PS1"
```

Обслуживание системы:

Вот то, что вы можете сделать в изолированной среде:

- Обновить или откатить пакеты
- Пересобрать образ initrd
- Сбросить забытый пароль
- Исправить /etc/fstab
- Переустановить GRUB

Выход из chroot окружения:

Когда вы закончите работу, выйдите из chroot, введя команду exit.

Теперь отмонтируйте устройства и каталоги, которые вам больше не нужны:

```
# umount {proc,sys,dev,boot...}
```

Наконец отмонтируйте ваш жёсткий диск:

```
# cd /
# umount /mnt/arch
```

Если вы видите ошибку подобную этой: "/mnt (или другой каталог) is busy", вы можете узнать причину (используя lsof) или принудительно отмонтировать каталог:

```
# umount -f /mnt
```

После этого вы сможете безопасно отключить систему.

Теоретические сведения по работе с утилитой dd

Утилита dd linux, которая просто выполняет копирование данных из одного места в другое на двоичном уровне. Она может скопировать CD/DVD диск, раздел на диске или даже целый жесткий диск.

Утилита просто переносит по одному блоку данных указанного размера с одного места в другое. Поскольку в Linux все, в том числе, устройства, считается файлами, вы можете переносить устройства в файлы и наоборот.

С помощью различных опций утилиты можно повлиять на размер блока, а это, в свою очередь, уже влияет на скорость работы программы. Далее мы рассмотрим основные опции утилиты и ее возможности.

Синтаксис утилиты очень прост:

```
$ dd if=источник_копирования of=место_назначения параметры
```

С помощью параметра if вам нужно указать источник, откуда будут копироваться блоки, это может быть устройство, например, /dev/sda или файл - disk.img. Далее, с помощью параметра of необходимо задать устройство или файл назначения. Другие параметры имеют такой же синтаксис, как if и of.

Рассмотрим дополнительные параметры:

bs - указывает сколько байт читать и записывать за один раз;

cbs - сколько байт нужно записывать за один раз;

count - скопировать указанное количество блоков, размер одного блока указывается в параметре bs;

conv - применить фильтры к потоку данных;

ibs - читать указанное количество байт за раз;

obs - записывать указанное количество байт за раз;

seek - пропустить указанное количество байт в начале устройства для чтения;

skip - пропустить указанное количество байт в начале устройства вывода;

status - указывает насколько подробным нужно сделать вывод;

iflag, oflag - позволяет задать дополнительные флаги работы для устройства ввода и вывода, основные из них: nosync, nofollow.

Например, чтобы сохранить образ диска в файл можно использовать такую команду:

```
sudo dd if=/dev/sr0 of=~ /CD.iso bs=2048 conv=noerror
```

Фильтр поетгор позволяет отключить реагирование на ошибки. Далее, вы можете создать образ жесткого диска или раздела на нем и сохранить этот образ на диск. Только смотрите не сохраните на тот же жесткий диск или раздел, чтобы не вызвать рекурсию:

```
dd if=/dev/sda of=~/.disk.img
```

Главная

Терминал

Команда dd Linux

Команда dd Linux

Терминал Март 29, 2017 9 admin

Довольно часто системным администраторам приходится копировать различные двоичные данные. Например, иногда может понадобиться сделать резервную копию жесткого диска, создать пустой файл, заполненный нулями для организации пространства подкачки или другой виртуальной файловой системы.

Для решения всех этих задач используется утилита dd linux, которая просто выполняет копирование данных из одного места в другое на двоичном уровне. Она может скопировать CD/DVD диск, раздел на диске или даже целый жесткий диск. В этой статье мы рассмотрим что из себя представляет команда dd linux, основные ее опции и параметры, а также как ею пользоваться.

Содержание статьи:

Как работает команда dd?

Команда dd

Как пользоваться dd?

Выводы

Как работает команда dd?

Сначала нужно понять как работает команда dd и что она делает. Фактически, это аналог утилиты копирования файлов cp только для блочных данных. Утилита просто переносит по одному блоку данных указанного размера с одного места в другое. Поскольку в Linux все, в том числе, устройства, считается файлами, вы можете переносить устройства в файлы и наоборот.

С помощью различных опций утилиты можно повлиять на размер блока, а это, в свою очередь, уже влияет на скорость работы программы. Далее мы рассмотрим основные опции утилиты и ее возможности.

Команда dd

Синтаксис утилиты достаточно необычен, но в то же время очень прост, после того как вы его запомните и привыкнете:

```
$ dd if=источник_копирования of=место_назначения параметры
```

С помощью параметра `if` вам нужно указать источник, откуда будут копироваться блоки, это может быть устройство, например, `/dev/sda` или файл - `disk.img`. Далее, с помощью параметра `of` необходимо задать устройство или файл назначения. Другие параметры имеют такой же синтаксис, как `if` и `of`.

Теперь давайте рассмотрим дополнительные параметры:

- `bs` - указывает сколько байт читать и записывать за один раз;
- `cbs` - сколько байт нужно записывать за один раз;
- `count` - скопировать указанное количество блоков, размер одного блока указывается в параметре `bs`;
- `conv` - применить фильтры к потоку данных;
- `ibs` - читать указанное количество байт за раз;
- `obs` - записывать указанное количество байт за раз;
- `seek` - пропустить указанное количество байт в начале устройства для чтения;
- `skip` - пропустить указанное количество байт в начале устройства вывода;
- `status` - указывает насколько подробным нужно сделать вывод;
- `iflag, oflag` - позволяет задать дополнительные флаги работы для устройства ввода и вывода, основные из них: `nocache`, `nofollow`.

Это были все основные опции, которые вам могут понадобиться. Теперь перейдем ближе к практике и рассмотрим несколько примеров как пользоваться утилитой `dd linux`.

Как пользоваться `dd`?

Обычные пользователи используют команду `dd` чаще всего для создания образов дисков DVD или CD. Например, чтобы сохранить образ диска в файл можно использовать такую команду:

```
sudo dd if=/dev/sr0 of=~ /CD.iso bs=2048 conv=noerror
```

Фильтр `noerror` позволяет отключить реагирование на ошибки. Далее, вы можете создать образ жесткого диска или раздела на нем и сохранить этот образ на диск. Только смотрите не сохраните на тот же жесткий диск или раздел, чтобы не вызвать рекурсию:

```
dd if=/dev/sda of=~ /disk.img
```

В вашей домашней папке будет создан файл с именем disk1.img, который в будущем можно будет развернуть и восстановить испорченную систему. Чтобы записать образ на жесткий диск или раздел достаточно поменять местами адреса устройств:

```
dd of=~/.disk.img if=/dev/sda
```

Очень важная и полезная опция - это bs. Она позволяет очень сильно влиять на скорость работы утилиты. Этот параметр позволяет установить размер одного блока при передаче данных. Здесь нужно задать цифровое значение с одним из таких модификаторов формата:

- c - один символ;
- b - 512 байт;
- kB - 1000 байт;
- K - 1024 байт;
- MB - 1000 килобайт;
- M - 1024 килобайт;
- GB - 1000 мегабайт;
- G - 1024 мегабайт.

Команда dd linux использует именно такую систему, она сложная, но от этого никуда не деться. Ее придется понять и запомнить. Например, 2b - это 1 килобайт, и 1k, это тоже 1 килобайт, 1M - 1 мегабайт. По умолчанию утилита использует размер блока - 512 байт. Например, чтобы ускорить копирование диска можно брать блоки размером по 5 мегабайт. Для этого применяется такая команда:

```
dd if=/dev/sda of=~/.disk.img bs=5M
```

Следующий параметр - это count. С помощью него можно указать сколько блоков необходимо скопировать. Например, мы можем создать файл размером 512 мегабайт, заполнив его нулями из /dev/zero или случайными цифрами из /dev/random:

```
sudo dd if=/dev/zero of=file.img bs=1M count=512
```

Обратите внимание, что этот параметр указывает не размер в мегабайтах, а всего лишь количество блоков. Поэтому, если вы укажете размер блока 1b, то для создания файла размером 1Кб нужно взять только два блока. С помощью этого параметра также можно сделать резервную копию таблицы разделов MBR. Для этого скопируем в файл первые 512 байт жесткого диска:

```
sudo dd if=/dev/sda of=mbr.img bs=1b count=1
```

Для восстановления используйте обычную команду развертывания образа на диск.

Если образ диска слишком большой, можно перенаправить весь вывод нестандартный поток вывода утилиты gzip:

```
dd if =/dev/sda2 | bzip2 disk.img.bz2
```

Также можно использовать утилиту dd linux для копирования файлов, хотя это и не является ее прямым предназначением:

```
dd if=/home/sergiy/test.txt of=/home/sergiy/test1.txt
```

Для копирования, например, образа загрузочного диска с ОС Ubuntu на флеш накопитель достаточно выполнить команду

```
dd if=xubuntu-16.04.5-desktop-amd64.iso of=/dev/sdc bs=4096 status=progress oflag=nocache
```

Однако надо помнить, что при неправильном использовании данной утилиты все данные в хранилище могут быть безвозвратно уничтожены. Например, команда dd linux пишет данные на диск непосредственно в двоичном виде, это значит, что записываются нули и единицы. Они переопределяют то, что было раньше размещено на устройстве для записи. Поэтому чтобы стереть диск вы можете просто забить его нулями из /dev/zero.

```
sudo dd if=/dev/zero of=/dev/sdb
```

Такое использование dd приводит к тому что весь диск будет полностью стерт.

Задание на выполнение практической работы

1. Получите из репозитория в Интернет дистрибутив выбранной вами ОС на базе ядра Linux в виде образа компакт диска формата iso.
2. Создайте с помощью утилиты dd загрузочную флешку из полученного образа.
3. Запустите в пробном однопользовательском режиме с флеш накопителя установленную на нее ОС на базе ядра Linux.
4. Смените корневой каталог текущей загруженной ОС на корневой каталог установленной в предыдущих работах ОС Linux.
5. Оформите отчет и представьте выполненную работу на защиту преподавателю.

Вопросы к защите практической работы

- 1) Что понимается под термином “корневой каталог”?
- 2) Поясните общую структуру команды chroot.
- 3) Поясните общую структуру команды dd.