

Практическая работа «Поиск файлов»

Команды **locate** и **find** используются для поиска файла в файловой системе. Хотя обе команды выполняют одинаковые задачи, каждая из них выполняет свою методику, со своими собственными отличительными преимуществами и недостатками. Команда **locate** использует базу данных, которая обновляется один раз в день с помощью «задания cron», а команда **find** выполняет поиск в действующей файловой системе.

Для установки программы **locate** выполните следующие команды:

Debian

```
apt-get install -y locate
```

Centos

```
yum install -y mlocate
```

Чтобы найти все файлы со словом «**hostname**» в имени файла, выполните следующую команду:

```
locate hostname
```

Чтобы найти все файлы со словом «**hostname**» в имени файла и сделать поиск нечувствительным к регистру, выполните следующую команду:

```
locate -i hostname
```

Чтобы найти файл «**hostname**» в каталоге **/etc**, а также перенаправить поток ошибок, выполните следующую команду:

```
find /etc -name hostname 2>/dev/null
```

Команда **find** имеет опции для поиска файлов по их временным меткам (см. лекц.). Чтобы найти все файлы в каталоге **/etc**, которые были изменены менее 2 дней назад, выполните следующую команду:

```
find /etc -mtime -2 2>/dev/null
```

Чтобы найти все файлы, которые были изменены менее 30 минут назад в каталоге **/etc**, выполните следующую команду:

```
find /etc -mmin -30 2>/dev/null
```

Чтобы найти файлы размером более 8 МБ, выполните следующую команду:

```
find / -size +8M 2>/dev/null
```

Чтобы найти все пустые файлы в домашнем каталоге пользователя, выполните следующие команды:

```
touch sample
```

```
find ~/ -empty 2>/dev/null
```

Чтобы найти все файлы, которые являются файлами каталога, в домашнем каталоге пользователя, выполните следующие команды:

```
mkdir dir1 dir2 dir3
```

```
find ~/ -type d 2>/dev/null
```

Чтобы отобразить счетчик всех файлов в структуре каталогов **/etc**, заканчивающихся на **.conf**, выполните следующую команду:

```
find /etc -name "*.conf" 2>/dev/null | wc -l
```

Чтобы ограничить глубину, которую ищет команда **find**, используйте параметр **-maxdepth**:

```
find /etc -name "*.conf" -maxdepth 1 2>/dev/null | wc -l
```

Чтобы найти все файлы, начинающиеся с «**host**», в структуре каталогов **/etc** и отобразить подробную информацию о файлах, используйте следующую команду:

```
find /etc -name "hosts*" -ls 2>/dev/null
```

Опция **-exec** используется для выполнения определенной команды, используя результаты поиска в качестве входных данных. Чтобы найти все файлы в вашем домашнем каталоге, которые заканчиваются на «**.conf**», и удалить их с помощью команды **rm**, выполните следующие команды:

```
cp /etc/*.conf .
```

```
ls
```

```
find ~ -name "*.conf" -exec rm {} \;
```

```
ls
```

Опция **-ok** используется для выполнения определенной команды, используя результаты поиска в качестве входных данных после подтверждения пользователем. Чтобы найти все файлы, заканчивающиеся словом «**.conf**», а затем удалить их в интерактивном режиме, выполните следующие команды:

```
cp /etc/*.conf .
ls
find ~ -name "*.conf" -ok rm {} \;
```

Другие примеры:

4. ИНВЕРТИРОВАНИЕ ШАБЛОНА

Найти файлы, которые не соответствуют шаблону:

```
find . -not -name "test*"
```

5. НЕСКОЛЬКО КРИТЕРИЕВ

Поиск командой find в Linux по нескольким критериям, с оператором исключения:

```
find . -name "test" -not -name "*.php"
```

Найдет все файлы, начинающиеся на test, но без расширения php. А теперь рассмотрим оператор ИЛИ:

```
find -name "*.html" -o -name "*.php"
```

6. НЕСКОЛЬКО КАТАЛОГОВ

Искать в двух каталогах одновременно:

```
find ./test ./test2 -type f -name "*.c"
```

7. ПОИСК СКРЫТЫХ ФАЙЛОВ

Найти скрытые файлы:

```
find ~ -type f -name ".*"
```

8. ПОИСК ПО РАЗРЕШЕНИЯМ

Найти файлы с определенной маской прав, например, 0664:

```
find . type f -perm 0664
```

Найти файлы с установленным флагом suid/guid:

```
find / -perm 2644
```

Или так:

```
find / -maxdepth 2 -perm /u=s
```

Поиск файлов только для чтения:

```
find /etc -maxdepth 1 -perm /u=r
```

Найти только исполняемые файлы:

```
find /bin -maxdepth 2 -perm /a=x
```

9. ПОИСК ФАЙЛОВ ПО ГРУППАМ И ПОЛЬЗОВАТЕЛЯХ

Найти все файлы, принадлежащие пользователю:

```
find . -user sergiy
```

Поиск файлов в Linux принадлежащих группе:

```
find /var/www -group developer
```

10. ПОИСК ПО ДАТЕ МОДИФИКАЦИИ

Поиск файлов по дате в Linux осуществляется с помощью параметра mtime. Найти все файлы модифицированные 50 дней назад:

```
find / -mtime 50
```

Поиск файлов в Linux открытых N дней назад:

```
find / -atime 50
```

Найти все файлы, модифицированные между 50 и 100 дней назад:

```
find / -mtime +50 -mtime -100
```

Найти файлы измененные в течении часа:

```
find . -cmin 60
```

11. ПОИСК ФАЙЛОВ ПО РАЗМЕРУ

Найти все файлы размером 50 мегабайт:

```
find / -size 50M
```

От пятидесяти до ста мегабайт:

```
find / -size +50M -size -100M
```

Найти самые маленькие файлы:

```
find . -type f -exec ls -s {} \; | sort -n -r | head -5
```

Самые большие:

```
find . -type f -exec ls -s {} \; | sort -n | head -5
```

12. ПОИСК ПУСТЫХ ФАЙЛОВ И ПАПКОК

```
find /tmp -type f -empty
```

```
$ find ~/ -type d -empty
```

13. ДЕЙСТВИЯ С НАЙДЕННЫМИ ФАЙЛАМИ

Для выполнения произвольных команд для найденных файлов используется опция -exec. Например, выполнить ls для получения подробной информации о каждом файле:

```
find . -exec ls -ld {} \;
```

Удалить все текстовые файлы в tmp

```
find /tmp -type f -name "*.txt" -exec rm -f {} \;
```

Удалить все файлы больше 100 мегабайт:

```
find /home/bob/dir -type f -name *.log -size +10M -exec rm -f {} \;
```