



★ 4.84

Оценка

1619.1

Рейтинг

Selectel

IT-инфраструктура для бизнеса



AndreiYemelianov

20 янв 2015 в 16:13

Xargs: многообразие вариантов использования

🕒 7 мин 👁 128K

Блог компании Selectel



Об утилите `xargs` написано очень много – что можно написать еще? Но если, что называется, копнуть поглубже, то выясняется, что во многих публикациях излагаются лишь самые основы, но нет главного: не объясняется, как можно применять `xargs` в реальной практике. Статей с разбором сложных и нетривиальных вариантов применения этого весьма полезного для системного администратора инструмента, к сожалению, очень

мало. Именно поэтому мы написали свою статью и постарались включить в нее как можно больше примеров использования `xargs` для решения различных проблем.

Сначала мы рассмотрим принцип работы `xargs` и разберем примеры попроще, а затем перейдем к разбору сложных и интересных кейсов.

Вспоминаем основы

Принцип работы `xargs` можно описать следующим образом: программа берет данные из стандартного ввода или из файла, разбивает их в соответствии с указанными параметрами, а затем передает другой программе в качестве аргумента.

В общем виде синтаксис команды `xargs` можно представить так:

```
[команда_генератор_списка] | xargs [опции_xargs] [команда]
```

Рассмотрим, как все это работает, на материале простых и хрестоматийных примеров.

Удаление файлов

Одна из самых частых ситуаций, в которых используется `xargs` – удаление файлов, найденных при помощи команды `find`.

Представим себе следующую ситуацию: имеется директория, в которой хранится большое количество файлов. Из нее нужно удалить файлы определенного типа (в нашем примере – файлы с расширением `*.sh`). Чтобы осуществить эту операцию, нужно передать `xargs` вывод команды `find`, и к файлам с указанным расширением будет применена команда `rm`:

```
$ ls
one.sh one.py two.sh two.py

$ find . -name "*.sh"| xargs rm -rf

$ ls
one.py two.py
```

Отметим, что операцию удаления файлов можно осуществить и без `xargs`, а с помощью команды

```
$ find . -name "*.sh" -exec rm -rf '{}' \
```

Описанный способ не сработает, если в имени одного из удаляемых файлов содержится пробел. Имя, состоящее из двух слов, разделенных пробелом, не будет воспринято как единое целое.

Проиллюстрируем это следующим примером:

```
$ ls
new file.sh one.sh one.py two.sh two.py

$ find . -name "*.sh"| xargs rm -rf

$ ls
new file.sh one.py two.py
```

Как видим, файл, в имени которого имеется пробел, не был удалён.

Чтобы решить эту проблему, используется опция `print0` для команды `find` и опция `-0` для команды `xargs`. Она заменяет стандартный разделитель (перенос строки на ноль-символ `(\x0)`), который и означает конец хранимой строки:

```
$ find . -name "*.sh" -print0 | xargs -0 rm -rf
```

`Xargs` может также помочь, например, быстро удалить все временные файлы, имеющие расширение `tmp`:

```
$ find /tmp -name "*.tmp"| xargs rm
```

Сжатие файлов

Сжать все файлы в текущей директории с помощью `gzip` можно, введя следующую команду:

```
$ ls | xargs -p -l gzip
```

Рассмотрим еще один пример: сжатие с помощью `tar` всех файлов с расширением `*.pl`:

```
$ find . -name "*.pl" | xargs tar -zcf pl.tar.gz
```

Переименование файлов

С помощью `xargs` можно осуществлять массовое переименование файлов. Представим себе, что у нас есть группа файлов с расширением `*.txt`, и нам нужно заменить это расширение на `*.sql`. Это можно сделать при помощи `xargs` и потокового текстового редактора `sed`:

```
$ ls | sed -e "p;s/.txt$/.sql/" | xargs -n2 fmv
```

В результате ее выполнения на консоль будет выведен список переименованных файлов.

С помощью `xargs` можно также добавлять к дополнительным элементам к именам файлов (например, дату):

```
$ ls | xargs -I FILE mv {} <...>--{}
```

Вместо `<...>` можно подставить всё, что угодно.

Фигурные скобки `{}` в этом примере означают «текущий аргумент» (т.е. текущее имя

файла).

Изменение прав для папок и файлов

С помощью `xargs` можно также ускорить процесс смены прав на файлы и папки для определенного пользователя или группы. Предположим, нам нужно найти все папки пользователя `root` и заменить их владельца на `temp`. Эта операция осуществляется при помощи команды:

```
$ find . -group root -print | xargs chown temp
```

Чтобы найти все папки группы `root` и заменить группу на `temp`, используется команда:

```
$ find . -group root -print | xargs chgrp temp
```

Xargs и find: сложные операции

С помощью команд `find` и `xargs` можно выполнять и более сложные операции. Вот так, например, можно удалить временные файлы, созданные более 7 дней назад:

```
$ find /tmp -type f -name '*' -mtime +7 -print0 | xargs -0 rm -f
```

А вот так – принудительно остановить процессы, которые уже работают больше 7 дней:

```
$ find /proc -user myuser -maxdepth 1 -type d -mtime +7 -exec basename {} \; | xarg
```



Xargs и cut

Xargs довольно часто используется в сочетании с командой `cut`, позволяющей вырезать строки из текстовых файлов. Рассмотрим некоторые практические примеры. С помощью приведённой ниже команды на консоль будет выведен список всех пользователей системы:

```
$ cut -d: -f1 < /etc/passwd | sort | xargs echo
```

А команда вида

```
file * | grep ASCII | cut -d":" -f1 | xargs -p vim
```

будет последовательно открывать файлы для редактирования в `vim`.

Обратим внимание на опцию `-p`. Благодаря ей команда будет выполняться в интерактивном режиме: перед открытием каждого файла будет запрашиваться подтверждение (`y/n`).

В заключение приведём ещё один сложный и интересный пример – рекурсивный поиск файлов самого большого размера в некоторой директории:

```
$ find . -type f -printf '%20s %p\n' | sort -n | cut -b22- | tr '\n' '\000' | xargs
```

Параллельный запуск процессов

Xargs часто используется для параллельного запуска нескольких процессов. Вот так, например, можно одновременно сжать несколько директорий в `tar.gz`:

```
$ echo dir1 dir2 dir3 | xargs -P 3 -I NAME tar czf NAME.tar.gz NAME
```

В приведенном примере используется ключ `-P`. Он указывает максимальное количество процессов, которые будут выполняться одновременно. Предположим, что у нас на входе имеется 10 аргументов. Если мы введём команду `xargs` с ключом `-P 3`, то будет запущено 3 экземпляра команды, следующей после `xargs`, с каждым из этих аргументов.

С помощью xargs можно также параллельно загружать из Интернета множество файлов:

```
$ wget -nv <ссылка> | egrep -o "http://[^[:space:]]*.jpg" | xargs -P 10 -n 1 wget -
```

В приведенном примере с указанного адреса будут скачаны все графические файлы с расширением jpg; ключ -P указывает, что требуется скачивать по 10 файлов одновременно.

Предварительные итоги

Подведём предварительные итоги и сформулируем несколько правил работы с xargs.

1. Xargs не работает с файлами, в имени которых присутствует пробел. Для решения этой проблемы с командой xargs используется опция -0. Пробел в имени файла можно обойти также следующим образом:

```
$ xargs -I FILE my_command "FILE"
```

2. Команда xargs принимает команды из со стандартного ввода, разделенные пробелом или переводом строки. Чтобы группировать эти команды, можно использовать двойные или одинарные кавычки. Можно также указать разделитель с помощью опции -d;
3. Если команде xargs не передать вообще никаких аргументов, то по умолчанию будет выполнена команда /bin/echo;
4. Во многих случаях команду xargs можно заменить циклом for. Например, команда

```
$ find . -type f -and -iname "*.deb" | xargs -n 1 dpkg -I
```

полностью эквивалента циклу

```
$ for file in `find . -type f -and -iname "*.deb"`; do dpkg -I "$file"; done
```

Нетривиальные примеры

Основы мы вспомнили, типичные варианты использования рассмотрели... Перейдем теперь к более сложным и нетривиальным примерам. До некоторых из них мы додумались самостоятельно, работая над повседневными задачами, а некоторые – почерпнули с сайта <http://www.commandlinefu.com> (всем желающим научиться тонкостям работы с командной строкой очень рекомендуем время от времени его посещать – там порой можно найти очень полезные советы).

Баним IP-адреса из списка

Чтобы забанить IP-адреса из списка, нужно их добавить в IP tables с правилом DROP. Эта операция осуществляется при помощи команды:

```
$ cat bad_ip_list | xargs -I IP iptables -A INPUT -s IP -j DROP
```

Можно проделать и более сложную операцию и забанить все адреса по AS:

```
$ /usr/bin/whois -H -h whois.ripe.net -T route -i origin AS<номер>|egrep "^route"|a
```

Изменяем формат URL

Преобразовать URL вида «`http%3A%2F%2Fwww.google.com`» в «`www,google.com`» можно при помощи команды:

```
echo "http%3A%2F%2Fwww.google.com" | sed -e 's/%\([0-9A-F][0-9A-F]\)/\\\\\\x\\1/g' | x
```

Генерируем пароль из 10 символов

Сгенерировать надежный пароль можно при помощи команды вида:


```
$ tr -dc A-Za-z0-9_ < /dev/urandom | head -c 10 | xargs
```

Генерировать пароли можно и без помощи `xargs`: для этого существует специализированная утилита `pwgen`. Некоторые другие способы генерации паролей описаны также [здесь](#).

Ищем бинарные файлы, установленные без использования `dpkg`

Такая операция может потребоваться в случае, если, например, машина стала жертвой хакерской атаки и на ней было установлено вредоносное программное обеспечение. Выявить, что за программы поставили злоумышленники, поможет следующая команда (она ищет запущенные «бинарники», установленные без использования менеджера пакетов `dpkg`):

```
$ cat /var/lib/dpkg/info/*.list > /tmp/listin ; ls /proc/*/exe |xargs -l readlink |
```

Удаляем устаревшие пакеты ядра

```
$ dpkg -l linux-* | awk '/^ii/{ print $2}' | grep -v -e `uname -r` | cut -f1,2 -d" -"
```

Проблема удаления старых ядер уже обсуждалась на Хабре – см. [здесь](#) (по этой же ссылке можно найти любопытные примеры команд).

Преобразуем скрипт в строку

Иногда возникает необходимость преобразовать большой скрипт в одну строку. Сделать это можно так:

```
$ (sed 's/#.*//g'|sed '/^ *$/d'|tr '\n' ';' |xargs echo) < script.sh
```

Заключение

Как видно из проделанного обзора, возможности xargs гораздо шире, чем это может показаться на первый взгляд. Надеемся, что приведённые в статье примеры окажутся для вас полезными. Если вам известны другие интересные варианты использования xargs – добро пожаловать в комментарии. Читателей, которые по тем или иным причинам не могут оставлять комментарии здесь, приглашаем в наш блог.

Теги: xargs, linux, системное администрирование, селектел, selectel

Хабы: Блог компании Selectel

Редакторский дайджест

Присылаем лучшие статьи раз в месяц

Электронная почта



Selectel

IT-инфраструктура для бизнеса

[ВКонтакте](#) [Telegram](#)



145



Карма

Рейтинг

Андрей Емельянов @AndreiYemelianov

Пользователь

 Комментарии 44

Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ



Holmogorov

23 часа назад

Советский UNIX. Как в СССР ковали собственную операционную систему

 Простой  9 мин  19K

Ретроспектива

 +83 92 71

ru_vds

19 часов назад

Итоги двадцати лет работы — технический долг и неподдерживаемый код

 Простой  8 мин  12K

Кейс

Перевод

 +50 51 45

mutatim

вчера в 15:28

Про FPV-квадрокоптеры. Введение

 Простой  23 мин  7.5K

Обзор

Из песочницы

 +34 92 53

DRoman0v

9 часов назад

Наиболее раздражающие причины поломок ноутбуков в моей практике: топ-5 проблем

 5 мин  9.7K +30 22 21

MaFrance351

вчера в 15:01

Самый необычный Windows-коммуникатор

 Простой  7 мин  2.9K

Ретроспектива

 +25 8 6

Показать еще

ВАКАНСИИ КОМПАНИИ «SELECTEL»

Python-разработчик в команду разработки Биллинговой платформы

Selectel • Можно удаленно

Frontend-разработчик (Angular) в команду разработки Облачного хранилища

Selectel • Можно удаленно

Python-разработчик в команду Compute

Selectel • Можно удаленно

Аудитор по информационной безопасности

Selectel • Санкт-Петербург

DevRel-специалист

Selectel • Санкт-Петербург

Больше вакансий на Хабр Карьере

Ваш аккаунт

Войти

Регистрация

Разделы

Статьи

Новости

Хабы

Компании

Авторы

Песочница

Информация

Устройство сайта

Для авторов

Для компаний

Документы

Соглашение

Конфиденциальность

Услуги

Корпоративный блог

Медийная реклама

Нативные проекты

Образовательные

программы

Стартапам

Спецпроекты



Настройка языка

[Вернуться на старую версию](#)

© 2006–2023, Habr

Дата регистрации	16 марта 2010
Дата основания	11 сентября 2007
Численность	501–1 000 человек
Местоположение	Россия
Представитель	Ульяна Малышева

ССЫЛКИ

Выделенный сервер от 26 рублей в день
[selectel.ru](#)

Сервер для 3D-моделирования и рендеринга
[selectel.ru](#)

Физический сервер от 800 рублей в месяц
[selectel.ru](#)

Облачные серверы от 280 рублей в месяц
[selectel.ru](#)

FAQ
[slc.tl](#)

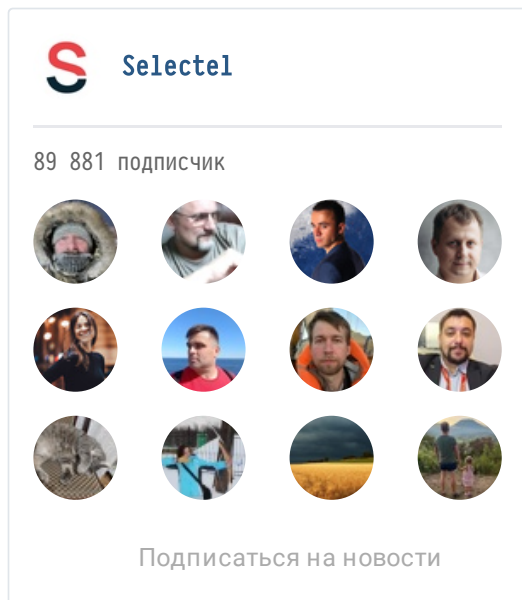
Реферальная программа
[slc.tl](#)

Telegram-канал о технологиях
[t.me](#)

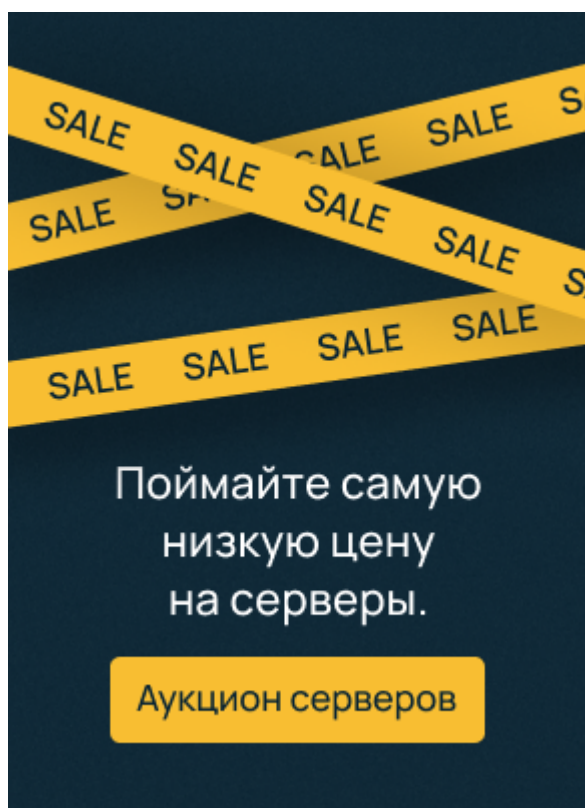
Telegram-канал про карьеру в IT
[t.me](#)

Вакансии
[slc.tl](#)

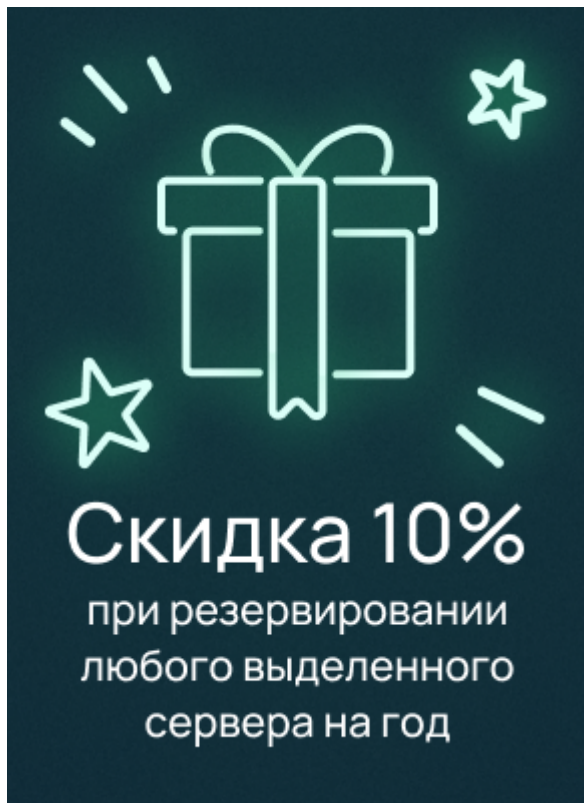
В КОНТАКТЕ



ВИДЖЕТ



ВИДЖЕТ



БЛОГ НА ХАБРЕ

9 часов назад

Наиболее раздражающие причины поломок ноутбуков в моей практике: топ-5 проблем

 9.7K  21

вчера в 03:40

Механические клавиатуры 2023 года: 5 моделей, на которые стоит обратить внимание

 13K  48

27 мая в 19:41

Дополнения и расширения для Raspberry Pi: варианты для DIY- и корпоративны проектов мая 2023 года

 5.1K  0

26 мая в 17:19

Новые инструменты для работы с ML-моделями и обзор MLOps от CERN

 1.9K  1

25 мая в 20:56

Обзор облачных IaaS-решений: как использовать преимущества частного облака в Public Cloud?



674



0