bash-тест выдает "echo: команда не найдена"

Задан 12 лет, 7 месяцев назад Изменено 12 лет, 7 месяцев назад Просмотрено 54 тыс. раз



Ок, внезапно происходит кое-что забавное. Этот тест

19

```
for i in `cat /mnt/usb/liste.txt `; do [ -f /mnt/usb/lsdvd.xml/$i ] || echo $i;
done
```



раньше работало нормально (для каждой записи в файле liste.txt попробуйте найти соответствующий каталог, выведите имя, если не найдено) и использовалось для получения имен отсутствующих каталогов. Теперь эта команда выдает

```
echo: command not found echo: command not found echo: command not found (...)
```

Я пытался использовать [[...]] и "test" вместо "[..]", но безрезультатно.

Есть идеи, что могло случиться с моим любимым bash, пока я не искал:)?

Спасибо, Кристиан.

Редактировать (вывод set -x):

```
(\ldots)
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_GHOST_WRITER.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`
+ '[' -f /mnt/usb/lsdvd.xml/THE_IMAGINARIUM_OF_DOCTOR_PARNASSUS.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_INFORMANT.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_INTERNATIONAL.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_MEN_WHO_STARE_AT_GOATS.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_OTHER_MAN.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_QUEEN.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_READER.lsdvd.xml ']'
+ for i in '`cat /mnt/usb/liste.txt`'
+ '[' -f /mnt/usb/lsdvd.xml/THE_REBOUND.lsdvd.xml ']'
+ $'\302\240echo' THE_REBOUND
+ '[' -x /usr/lib/command-not-found ']'
+ /usr/bin/python /usr/lib/command-not-found -- $'\302\240echo'
echo: command not found
+ return 127
(\ldots)
```

bash тестирование команда

Поделиться

Улучшите этот вопрос

Подписаться

отредактировано 22 марта 2011 в спрошено 21 марта 2011 в 17:44 9:51

Христианин

вы пробовали заключать в двойные кавычки \$i? -Yannick Loiseau 21 марта 2011 в 17:47

Если заменить конец строки на echooooo \$1; done, выдает ли это ошибку echooooo: command not found? Эту ошибку "команда не найдена" я получаю, когда набираю junk в командной строке bash. Что, если заменить его на /bin/echo? -Дэвид Йоу 21 марта 2011 в 17:56 🥕

Добавьте set -x в начало вашего скрипта и следите за каждой командой, которую выполняет оболочка. -bobbogo 21 марта 2011 в 18:17

Мои выводы: двойное взятие в кавычки \$ і: безрезультатно, та же ошибка. echoooo: команда не найдена. - Кристиан 22 марта 2011 в 9:48

4 Ответа

Отсортировано по:

Highest score (default)

×

\$

Не нашли ответ? Задайте вопрос на Stack Overflow на русском.



Если вы присмотритесь, то увидите, что она печатается:

38

echo: command not found



с начальным пробелом, а не:



echo: command not found



set -x делает это намного понятнее при печати:

\$'\302\240echo' THE_REBOUND

0302 0240 используется восьмеричный UTF-8 для неразрывного пробела.

Попробуйте удалить пробел между | и echo и введите обычный пробел.

Скорее всего, файл редактировался на Мас, где неразрывный пробел может быть введен случайно, если вы нажмете Option Одновременно с Space.

Если вы используете Vim, вы можете сделать неразрывные пробелы более очевидными, используя опцию listchars. Например:

```
set listchars+=nbsp:% list
```

Смотрите, как заставить Vim выделять подозрительные символы, чтобы получить еще несколько идей.

В Linux неразрывный пробел равен или Compose Space Space ИЛИ Ctrl + Shift + u а 0, так что вы, вероятно, не собираетесь вводить это случайно.

Для всех, кому интересно, можно вручную преобразовать восьмеричные байты в UTF-8, используя <u>таблицу</u> UTF-8 из статьи UTF-8 в Википедии.

Но, безусловно, самым простым способом, который я нашел, было использование Python:

```
>>> import unicodedata
>>> unicodedata.name('\302\240'.decode('utf-8'))
'NO-BREAK SPACE'
```

Поделиться Улучшите этот ответ

Подписаться

отредактировано 22 марта 2011 в ответил 22 марта 2011 в 10:23 11:25

24,9тыс. 8 65 66

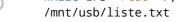


Редактировать: приведенный ниже код по-прежнему в порядке, но проблема, вероятно, в \$'\302\240есho': кажется, в вашем текстовом файле есть неизвестные **символы**. Попробуйте удалить пару символов перед echo, а затем ввести их снова.



Это вполне может быть проблема с *цитированием*. Кроме того, вы могли бы сэкономить некоторое время, избежав бесполезного использования cat:





while IFS= read -r; do [-f "/mnt/usb/lsdvd.xml/\$REPLY"] || echo "\$REPLY"; done <

Поделиться Улучшите этот ответ Подписаться

ответил 22 марта 2011 в 9:59





Что происходит при запуске simply:

for i in `cat /mnt/usb/liste.txt `; do echo \$i done



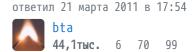


Во-первых, заметили ли вы что-нибудь ненормальное в именах файлов (пробелы, перевод строк, специальные символы и т.д.)? Если да, то вам нужно будет заключить имя файла в кавычки, чтобы оболочка не могла неправильно его интерпретировать.

Кроме того, попробуйте использовать обычный if оператор вместо логического OR:

```
for i in `cat /mnt/usb/liste.txt `; do
    if [ ! -f /mnt/usb/lsdvd.xml/$i ]; then
        echo $i
    fi
done
```

Поделиться Улучшите этот ответ Подписаться



Простой цикл for-echo-done работает нормально, ничего необычного в именах файлов. На самом деле, они довольно приятные: ни пробелов, ни четных чисел. – Кристиан 22 марта 2011 в 9:54

Цикл for-if-then-fi-done работает нормально, так что это должно быть что-то о || и странных строках без кавычек, но, похоже, я не могу найти ничего необычного в файлах и их названиях.

```
- Кристиан 22 марта 2011 в 9:55
```



Работает у меня:



```
for f in $(< d2.1st); do [ -f ./d2/$f ] || echo $f ; done
```



Только мои собственные имена файлов, каталогов и предложенная конструкция \$ (...) вместо обратных меток и с < вместо cat.



Возможно, в вашем списке есть перевод строки dos / Windows? Просмотрите его с помощью less.

Поделиться Улучшите этот ответ Подписаться

ответил 22 марта 2011 в 10:09 пользователь

неизвестен

35,6тыс. 11 75 121