

# IgorKa - Информационный ресурс

Немного обо всем и все о немногом, или практический опыт системного администратора.



- [Главная](#)
- [О сайте](#)
- [О себе](#)
- [Резюме](#)
- [Видео-уроки](#)
- [КУРСЫ LINUX](#)
- [Карта сайта](#)
- [Обратная связь](#)



Июль 2010

Пн Вт Ср Чт Пт Сб Вс

|                    |                   |                    |                    |                    |                    |                   |  |
|--------------------|-------------------|--------------------|--------------------|--------------------|--------------------|-------------------|--|
|                    |                   |                    | 1                  | <a href="#">2</a>  | 3                  | <a href="#">4</a> |  |
| <a href="#">5</a>  | <a href="#">6</a> | 7                  | <a href="#">8</a>  | <a href="#">9</a>  | <a href="#">10</a> | 11                |  |
| <a href="#">12</a> | 13                | 14                 | 15                 | 16                 | 17                 | 18                |  |
| 19                 | 20                | 21                 | <a href="#">22</a> | <a href="#">23</a> | 24                 | 25                |  |
| 26                 | 27                | <a href="#">28</a> | <a href="#">29</a> | <a href="#">30</a> | 31                 |                   |  |

[« Июнь](#)      [Авг »](#)

**Купить рекламу**

Купить рекламу



« [Установка и привязка профиля qutim к Dropbox в Ubuntu](#)  
[День системного администратора](#) »

## Временные параметры файла - atime mtime ctime



29 июля 2010, 19:34

Решил я сегодня уточнить значения параметров **mtime**, **atime** и **ctime**, которые присутствуют у каждого файла в **Linux**. На первый взгляд все вроде понятно:

**mtime - modification time** - время последней модификации (изменения) файла

**atime - access time** - время последнего доступа к файлу

**ctime - change time** - время последнего изменения атрибутов файла (данных которые хранятся в **inode-области**)

Но когда начинаешь спрашивать себя когда меняются эти параметры, в частности какие команды меняют их, а какие нет, и проверять на практике ответы на свои же вопросы - все не так очевидно. Особенно для каталогов, которые тоже являются разновидностью файлов в **Linux**. Вот решил поделиться некоторыми экспериментами в этом направлении. Эксперименты проводил конечно же в своей **Ubuntu 9.10** с файловой системой **ext4**. У **ext4**, кстати появилось два дополнительных временных параметра - **время создания файла linux** и **время удаления файла linux**, но о них в самом конце.

До начала своих экспериментов я предполагал следующее.

Параметр **mtime** - изменяется после того как изменяется содержимое файла. Например, открыли файл командой **nano**, дописали что-то, сохранили, закрыли, и время **mtime** поменялось. А как в случае с каталогами? Предполагал, что время модификации для каталога изменяется когда в каталоге создаются/удаляются файлы и подкаталоги.

Параметр **atime** - изменяется тогда, когда мы получаем доступ к файлу, например, той же командой **nano** мы получаем доступ к файлу. Значит **atime** должен измениться. Команды **cat**, **less**, **tail** выводят содержимое файла, значит мы получаем доступ к нему, но не меняем его поэтому **mtime** меняться не должен. А как быть с каталогами? Когда меняется **atime** для каталога? Тут я даже не знал, что себе ответить.

Параметр **ctime** - самый простой для моего понимания. Изменяется тогда когда изменяются права доступа к файлу (командой **chmod**), изменяется владелец файла (команда **chown**), создаются

[жесткие ссылки на файл](#) (команда **ln**). В этом плане различий с каталогом нет, с той лишь разницей, что на каталоги нельзя создавать жесткие ссылки.

Вот примерно так я понимал значение временных параметров файлов. Практические эксперименты несколько расширили и изменили мои познания. Во время экспериментов проверку временных параметров проверял командой **stat**, которая показывает сразу все три временных атрибута. Если использовать команду **ls**, то по умолчанию она выводит время **mtime** - **ls -l**, **ls -lu** (или **ls -time=atime|access|use**) - выводит время **atime** - время последнего доступа к файлу, **ls -lc** (или **ls -time=ctime|status**) - выводит время **ctime** - время последнего изменения атрибутов.

Начал с параметра **atime**. Создаю в домашнем каталоге, каталог **timetest** и проверяю его временные метки командой **stat** (вывод результата сокращаю для экономии места):

```
1$ mkdir timetest
2$ stat timetest
3Access: 2010-07-29 16:08:53.330974403 +0300
4Modify: 2010-07-29 16:08:53.330974403 +0300
5Change: 2010-07-29 16:08:53.330974403 +0300
```

Сейчас все временные метку равны друг другу так и должно быть. Далее захожу в каталог **timetest** и создаю пустой файл **test**, после чего проверяю временные метки каталога:

```
1$ touch ./timetest/test
2$ stat timetest
3Access: 2010-07-29 16:08:53.330974403 +0300
4Modify: 2010-07-29 16:34:24.442971906 +0300
5Change: 2010-07-29 16:34:24.442971906 +0300
```

и файла:

```
1$ stat timetest/test
2Access: 2010-07-29 16:34:24.442971906 +0300
3Modify: 2010-07-29 16:34:24.442971906 +0300
4Change: 2010-07-29 16:34:24.442971906 +0300
```

После создания файла в каталоге, ожидаемо изменился параметр каталога **atime** и неожиданно параметр **ctime**. Что ж возьму себе на заметку: **при создании файла в каталоге, у каталога изменяются временные параметры atime и ctime**. При создании подкаталога ситуация аналогичная. Попробую теперь удалить файл **test** и посмотреть как это повлияет на параметры каталога:

```
1$ stat timetest/
2Access: 2010-07-29 16:35:14.554977285 +0300
3Modify: 2010-07-29 16:54:09.082973370 +0300
4Change: 2010-07-29 16:54:09.082973370 +0300
5$ rm timetest/test
6$ stat timetest/
7Access: 2010-07-29 16:56:07.634977694 +0300
8Modify: 2010-07-29 16:56:13.386972802 +0300
9Change: 2010-07-29 16:56:13.386972802 +0300
```

Интересный результат... Время доступа к каталогу изменилось, но на 6 секунд раньше, чем время **mtime** и **ctime**. Значит команда удаления файла не могла изменить это время. А что тогда? Проведа

еще пару экспериментов догадался - это **двойной tab** который показывает содержимое каталога. То есть я набрал в консоли **rm ti**, затем нажал **tab**, чтобы дополнить до **timetest**, и затем нажал **двойной tab**, чтобы отобразилось содержимое каталога, а это получается и есть операция доступа к каталогу, которая и меняет время **atime**. В процессе выяснения этого обстоятельства определил еще один интересный момент. Оказывается **время доступа atime меняется только в том случае если оно меньше или равно времени mtime или ctime**. Если время доступа меньше, то сколько бы я не нажимал на **двойной tab** - время доступа не менялось. Проверить этот момент очень просто. Командой **touch -m** меняю время модификации каталога на текущее и проверяю изменения:

```
1$ touch -m timetest/
2$ stat timetest/
3Access: 2010-07-29 17:15:34.762968549 +0300
4Modify: 2010-07-29 17:30:51.857519567 +0300
5Change: 2010-07-29 17:30:51.854977549 +0300
```

Отмечаю, что снова изменился параметр **ctime**. Поэтому записываю себе еще одно предположение, **что изменение времени модификации файла, автоматически изменяет и время изменения атрибутов файла**.

Теперь если снова выполнить **двойной tab**, то время доступа изменится:

```
1$ stat timetest/
2dir1/ file3 file4
3$ stat timetest/
4Access: 2010-07-29 17:31:17.410974316 +0300
5Modify: 2010-07-29 17:30:51.857519567 +0300
6Change: 2010-07-29 17:30:51.854977549 +0300
```

Также проверил, что команды **ls**, **du** меняют время доступа, а команда **cd** и **pwd** нет. Что впрочем логично.

С временем доступа **atime** для файлов ситуация аналогичная. При выводе содержимого файла командами **cat**, **less**, **tail** изменяется время доступа, но опять таки только в том случае если оно меньше или равно времени **mtime** или **ctime**.

Далее решил проверить как себя ведет изменение времени модификации - **mtime**. Здесь для меня все обошлось без сюрпризов. Для каталога время модификации меняется всякий раз когда в каталоге создаются/удаляются/переименовываются подкаталоги и файлы.

```
1 $ stat timetest/
2 Access: 2010-07-29 17:41:02.706976846 +0300
3 Modify: 2010-07-29 17:40:13.765539027 +0300
4 Change: 2010-07-29 17:40:13.762977529 +0300
5
6 $ rm timetest/file3
7
8 $ stat timetest/
9 Access: 2010-07-29 17:41:02.706976846 +0300
10Modify: 2010-07-29 17:52:04.758977254 +0300
11Change: 2010-07-29 17:52:04.758977254 +0300
1 $ mkdir timetest/dir2
2
3 $ stat timetest/
```

```
4 Access: 2010-07-29 17:41:02.706976846 +0300
5 Modify: 2010-07-29 17:52:33.238971947 +0300
6 Change: 2010-07-29 17:52:33.238971947 +0300
7
8 igor@adm-ubuntu:~$ rm -r timetest/dir2
9
10 igor@adm-ubuntu:~$ stat timetest/
11 Access: 2010-07-29 17:41:02.706976846 +0300
12 Modify: 2010-07-29 17:53:05.874971500 +0300
13 Change: 2010-07-29 17:53:05.874971500 +0300
1 $ touch timetest/file5
2
3 $ stat timetest/
4 Access: 2010-07-29 17:55:26.978977669 +0300
5 Modify: 2010-07-29 17:58:53.078971720 +0300
6 Change: 2010-07-29 17:58:53.078971720 +0300
7
8 $ mv timetest/file5 timetest/file6
9
10 $ stat timetest/
11 Access: 2010-07-29 17:55:26.978977669 +0300
12 Modify: 2010-07-29 17:59:40.314988184 +0300
13 Change: 2010-07-29 17:59:40.314988184 +0300
```

Также из примеров видно, что действительно при изменении времени **mtime** автоматически изменяется и время **ctime**. В тоже время временной параметр **atime** каталога **timetest** не меняется от действия команд **rm**, **mkdir**, **mv**.

Для файла параметр **mtime** меняется при изменении содержимого файла. Открыв файл в редакторе nano, изменив содержимое и сохранив изменения увидел следующую картину:

до изменения файла:

```
1 $ stat timetest/file7
2 Access: 2010-07-29 17:58:53.078971720 +0300
3 Modify: 2010-07-29 18:00:39.058974330 +0300
4 Change: 2010-07-29 18:27:06.246971667 +0300
```

после изменения и сохранения изменений:

```
1 $ stat timetest/file7
2 Access: 2010-07-29 18:29:22.666968437 +0300
3 Modify: 2010-07-29 18:29:28.798977680 +0300
4 Change: 2010-07-29 18:29:28.798977680 +0300
```

Обратите внимание, что в данной ситуации изменяются все временные параметры: **atime** - в момент открытия файла, **mtime** и **ctime** в момент сохранения изменений в файле. Кстати команда переименования файла **mv** не должна изменять временной параметр файла **mtime**. Дело в том, что в этом случае модифицируется только имя файла, которое хранится в **inode-области**, но не в самом файле. По логике не должна команда **mv** и менять время доступа. А вот параметр снапе как раз и должен измениться. Но чего гадать - проверяю на практике:

```
1 $ stat timetest/file7
```

```
2 File: «timetest/file7»
3 Size: 8      Blocks: 8      IO Block: 4096  обычный файл
4 Device: 804h/2052d Inode: 659938  Links: 1
5 Access: (0644/-rw-r--r--) Uid: ( 1000/  igor)  Gid: ( 1000/  igor)
6 Access: 2010-07-29 18:34:20.438968510 +0300
7 Modify: 2010-07-29 18:34:24.438971636 +0300
8 Change: 2010-07-29 18:34:24.438971636 +0300
9
10$ mv timetest/file7 timetest/file8
11
12$ stat timetest/file8
13 File: «timetest/file8»
14 Size: 8      Blocks: 8      IO Block: 4096  обычный файл
15Device: 804h/2052d Inode: 659938  Links: 1
16Access: (0644/-rw-r--r--) Uid: ( 1000/  igor)  Gid: ( 1000/  igor)
17Access: 2010-07-29 18:34:20.438968510 +0300
18Modify: 2010-07-29 18:34:24.438971636 +0300
19Change: 2010-07-29 18:40:28.914973652 +0300
```

Действительно изменился только параметр **ctime**. Здесь я привел вывод команды **stat** полностью, чтобы обратить внимание на то, что при переименовании файла, действительно меняется имя файла, а **inode** - идентификатор файла остается неизменным - **Inode: 659938**.

Так я плавно подошел к параметру **ctime** :) Для каталогов и для файлов этот параметр меняется после выполнения команд **chmod**, и **chown**.

```
1 $ stat timetest/dir1/
2 Access: 2010-07-29 18:49:00.602972037 +0300
3 Modify: 2010-07-29 18:49:00.602972037 +0300
4 Change: 2010-07-29 18:49:00.602972037 +0300
5
6 igor@adm-ubuntu:~$ chown igor:igor timetest/dir1
7
8 igor@adm-ubuntu:~$ stat timetest/dir1/
9 Access: 2010-07-29 18:49:00.602972037 +0300
10Modify: 2010-07-29 18:49:00.602972037 +0300
11Change: 2010-07-29 18:52:07.154977433 +0300
12
13igor@adm-ubuntu:~$ chmod o-r timetest/dir1
14
15igor@adm-ubuntu:~$ stat timetest/dir1/
16Access: 2010-07-29 18:49:00.602972037 +0300
17Modify: 2010-07-29 18:49:00.602972037 +0300
18Change: 2010-07-29 18:52:26.950977903 +0300
```

Для файла **ctime** меняется и после создания жесткой ссылки на файл:

```
1 $ stat timetest/file4
2 Access: 2010-07-29 18:54:14.007298839 +0300
3 Modify: 2010-07-29 17:11:32.442977462 +0300
4 Change: 2010-07-29 18:55:46.318977548 +0300
5
```

```
6 igor@adm-ubuntu:~$ ln timetest/file4 timetest/lnfile6
7
8 igor@adm-ubuntu:~$ stat timetest/file4
9 Access: 2010-07-29 18:54:14.007298839 +0300
10Modify: 2010-07-29 17:11:32.442977462 +0300
11Change: 2010-07-29 18:56:17.998971658 +0300
```

На этом буду заканчивать свой практический эксперимент, но в конце о двух новых временных атрибутах которые появились в файловой системе **ext4**. Это долгожданный многими **ctime** - **create time** - время создания файла, с которым очень часто путали параметр **ctime**. И параметр **dtime** - **delete time** - время удаления файла.

Как же посмотреть время создания файла - **ctime**? Так как функционал новый, то пока не нашел команды которая выдает эту информацию, но нашел [как выходят из положения другие](#). Для этого используют команду **debugfs**. У меня в **Ubuntu** она шла по умолчанию. Чтобы воспользоваться программой нужны права администратора и имя раздела на котором находится файл.

Смотрю куда у меня примонтирован домашний каталог в котором я работал:

```
1$ mount | grep home
2/dev/sda4 on /home type ext4 (rw)
```

Далее используем команду **debugfs** в таком виде:

```
1$ sudo debugfs -R 'stat /igor/timetest/' /dev/sda4
```

После нажатия Enter вижу следующую информацию:

```
1 Inode: 659937  Type: directory  Mode: 0755  Flags: 0x80000
2 Generation: 3428284077  Version: 0x00000000:00000018
3 User: 1000  Group: 1000  Size: 4096
4 File ACL: 0  Directory ACL: 0
5 Links: 3  Blockcount: 8
6 Fragment: Address: 0  Number: 0  Size: 0
7 ctime: 0x4c51a4a1:ee2c6428 -- Thu Jul 29 18:56:17 2010
8 atime: 0x4c51a398:029e0340 -- Thu Jul 29 18:51:52 2010
9 mtime: 0x4c51a4a1:ee2c6428 -- Thu Jul 29 18:56:17 2010
10ctime: 0x4c517d65:4ee9130c -- Thu Jul 29 16:08:53 2010
11Size of extra inode fields: 28
12EXTENTS:
13(0): 2650522
```

Чтобы вернуться в консоль нажимаю **q**. Видим параметр **ctime** равный Thu Jul 29 16:08:53 2010, что совпадает с временем создания (см. самый первый вывод команды **stat**).

А где же параметр **dtime**? А он появляется только когда файл удален. А как тогда его посмотреть? А посмотреть можно по **inode**. На примере файла **file8**:

Смотрим сначала **inode** файла:

```
1$ ls -li timetest/file8
2659938 -rw-r--r-- 1 igor igor 8 2010-07-29 18:34 timetest/file8
```

**inode** равен 659938. Теперь я удаляю файл:

```
1$ rm timetest/file8
```

и выполняю команду **debugfs**, где указываю не имя файла, а **inode**:

```
1 $ sudo debugfs -R 'stat <659938>' /dev/sda4
2
3 Inode: 659938  Type: regular  Mode: 0644  Flags: 0x80000
4 Generation: 3428285866  Version: 0x00000000:00000001
5 User: 1000  Group: 1000  Size: 0
6 File ACL: 0  Directory ACL: 0
7 Links: 0  Blockcount: 0
8 Fragment: Address: 0  Number: 0  Size: 0
9  ctime: 0x4c51a95e:8afe201c -- Thu Jul 29 19:16:30 2010
10 atime: 0x4c519f7c:68a882f8 -- Thu Jul 29 18:34:20 2010
11 mtime: 0x4c51a95e:8afe201c -- Thu Jul 29 19:16:30 2010
12 ctime: 0x4c51972d:12d40d20 -- Thu Jul 29 17:58:53 2010
13 dtime: 0x4c51a95e -- Thu Jul 29 19:16:30 2010
14 Size of extra inode fields: 28
15 EXTENTS:
```

Вот и параметр **dtime** нашелся :)

На этом все. Надеюсь статья будет кому-то полезной.

Спасибо за то что поделились  
ссылкой!



Теги: [Linux](#), [Ubuntu](#)

Рубрика: [Разное](#) | [Отзывы \(RSS\)](#)

### Статьи и новости схожей тематики:

- [Ubuntu и ADSL модем в режиме бриджа](#)
- [Проверка поддержки процессором аппаратной виртуализации](#)
- [VBoxManage - VirtualBox из командной строки](#)
- [Your kernel probably supports journaled quota...](#)
- [Установка поддержки o3d для chrome в Ubuntu](#)

### Комментариев: 3

1. [Доступно приложение к PDF-журналу UserAndLINUX для системных администраторов:](#)

[...] Временные параметры файла – ATIME MTIME CTIME [...]

[18 января 2011, 13:44](#)

2. [Доступно приложение к PDF-журналу UserAndLINUX для системных администраторов | AllUNIX.ru – Всероссийский портал о UNIX-системах:](#)



[...] Временные параметры файла – ATIME MTIME CTIME [...]

[18 января 2011, 16:20](#)

### 3. Zenitur:

Привет. Я столкнулся с такой проблемой. Моя система стала выдавать просьбу почистить жёсткий диск. С помощью kldirstat я нашёл большие файлы, которые мне не нужны. Потом просьба повторилась снова. Я увеличил раздел на гигабайт. И вот снова повторилась. Но у меня нет ничего лишнего! Значит всё заполнилось большим количеством маленьких файлов. Но где они? Я хочу получить список файлов, которые создавались и изменялись последними. kldirstat позволяет сделать сортировку по дате только в пределах одного каталога. Поможешь?

[Ответить](#)

[4 февраля 2013, 13:32](#)

## Оставьте свой отзыв

Имя \*

Почта (скрыта) \*

Сайт

Добавить

•  Search  
Google Custom Search

#### • Метки

[ActiveSync](#) [ADSL](#) [Apache](#) [Apple](#) [apt](#) [audio](#) [awk](#) [Bada](#) [bash](#) [Chrome](#) [D-Link](#) [DNS](#) [Dropbox](#) [DynDNS](#) [English](#) [ETKA](#) [fdisk](#) [fstab](#)

[Google](#) [Grub2](#) [iOS](#) [IP-телефония](#) [iPad](#) [iPhone](#) [Linux](#) [Lotus Notes](#) [LVM](#) [mount](#) [MS Office](#) [MySQL](#) [nano](#) [NTFS](#) [on-line tv](#)

[OpenVPN](#) [Panasonic TDA 200](#) [Samsung](#) [SEO](#) [SMDR Tariff](#) [SVN](#) [Ubuntu](#) [Ubuntu 10.04](#) [Ubuntu 12.04](#) [upstart](#) [VirtualBox](#)

[Webmoney](#) [Wi-Fi](#) [Windows](#) [Windows 7](#) [Wordpress](#) [xCode](#) [Андроид](#) [БЛОГ](#) [Бытовая техника](#) [Бэкапы](#) [Виртуализация](#) [Графика](#)

[Здоровье](#) [Игры](#) [Интернет](#) [Квоты](#) [Киевстар](#) [Кино](#) [Команды](#) [Консоль](#) [МФУ](#) [Микроконтроллеры](#) [НЖМД](#) [ОГО!](#) [Обзор](#)

[Планшеты](#) [Подкасты](#) [Почта](#) [Проблемы](#) [Программы](#) [Производительность](#) [Разметка диска](#) [Реестр](#) [Ремонт](#) [Сети](#) [Статистика](#)

[Телевизоры](#) [Телефоны](#) [Торговля](#) [Удаленный доступ](#) [Управление проектами](#) [Финансы](#) [Футбол](#) [Хостинг](#) [Юмор](#) [Яндекс](#) [Лекция](#)

[процессы](#) [сигналы](#)

•

## • Рубрики

- [Досуг](#) (18)
- [Закладки](#) (6)
- [Моя работа](#) (285)
  - [Lotus Notes](#) (6)
  - [Ubuntu](#) (83)
  - [Администрирование](#) (9)
  - [Безопасность](#) (15)
  - [Курсы Администратор ПК с Linux](#) (31)
  - [Разное](#) (133)
  - [Телефония](#) (7)
- [Новости](#) (73)
- [Обзоры устройств](#) (2)
- [Рекламная пауза :\)](#)  (49)
- [Статьи](#) (5)
- [Юмор](#) (9)

## • Последние комментарии

- [Настройка почты ssmtp для PHP mail\(\)](#)  [1](#)  
[Дмитрий](#)
- [Что такое динамический ДНС \(DynDNS\) и какая нам от него польза?](#)  [52](#)  
[Александр \[...\]](#)
- [Вопросы пользователей по Lotus Notes](#)  [10](#)  
[Анна \[...\]](#)
- [Что такое серый и белый IP адрес](#)  [37](#)  
[Михаил \[...\]](#)
- [Разрешение экрана Ubuntu в VirtualBox](#)  [44](#)  
[Алексей \[...\]](#)
- [Лекция №23 - bash. Объединение команд](#)  [7](#)  
[IgorKa \[...\]](#)
- [USPS - Посылки из США в Украину](#)  [5](#)  
[Yaroslav \[...\]](#)
- [D-Link DSL-2640U, ОГО и прошивка](#)  [37](#)  
[andrei \[...\]](#)
- [Remmina - Remote Desktop Client для Ubuntu](#)  [17](#)  
[Вячеслав \[...\]](#)
- [Как настроить DynDNS для домашнего компьютера](#)  [149](#)  
[Игорь \[...\]](#)

## • Ссылки

- [proft.com.ua](#)
- [Видео-уроки](#)

## • i na

[Публикации \(RSS\)](#) и [Комментарии \(RSS\)](#).

Все права защищены © 2009-2018 IgorKa (Игорь Кузьменко)

При использовании любых материалов с данного ресурса ссылка (гиперссылка) на [igorka.com.ua](http://igorka.com.ua) обязательна.