



# ITS PARTNER

Full-cycle software development

[www.itspartner.net](http://www.itspartner.net)



# Рекурсивное наблюдение за файловой системой

на примере библиотеки **librnotify**

[https://github.com / zmushko / librnotify](https://github.com/zmushko/librnotify)

Змушко Андрей

[andrey.zmushko@itspartner.net](mailto:andrey.zmushko@itspartner.net)

# Файловые события

- Создание
- Удаление
- Открытие
- Заккрытие
- Чтение
- Запись
- Изменение метаданных
- Перемещение (переименование)

# Подсистемы ядра Linux для нотификаций

- **dnotify** - устаревший, заменен на inotify.
- **inotify** - используется начиная с ядра 2.6.13
- **fanotify** - **не поддерживаются нотификации удаления и перемещения**, однако возвращает PID процесса вызвавшего нотификацию.

## **inotify вызовы:**

```
Int inotify_init(int flags); // возвращает fd
```

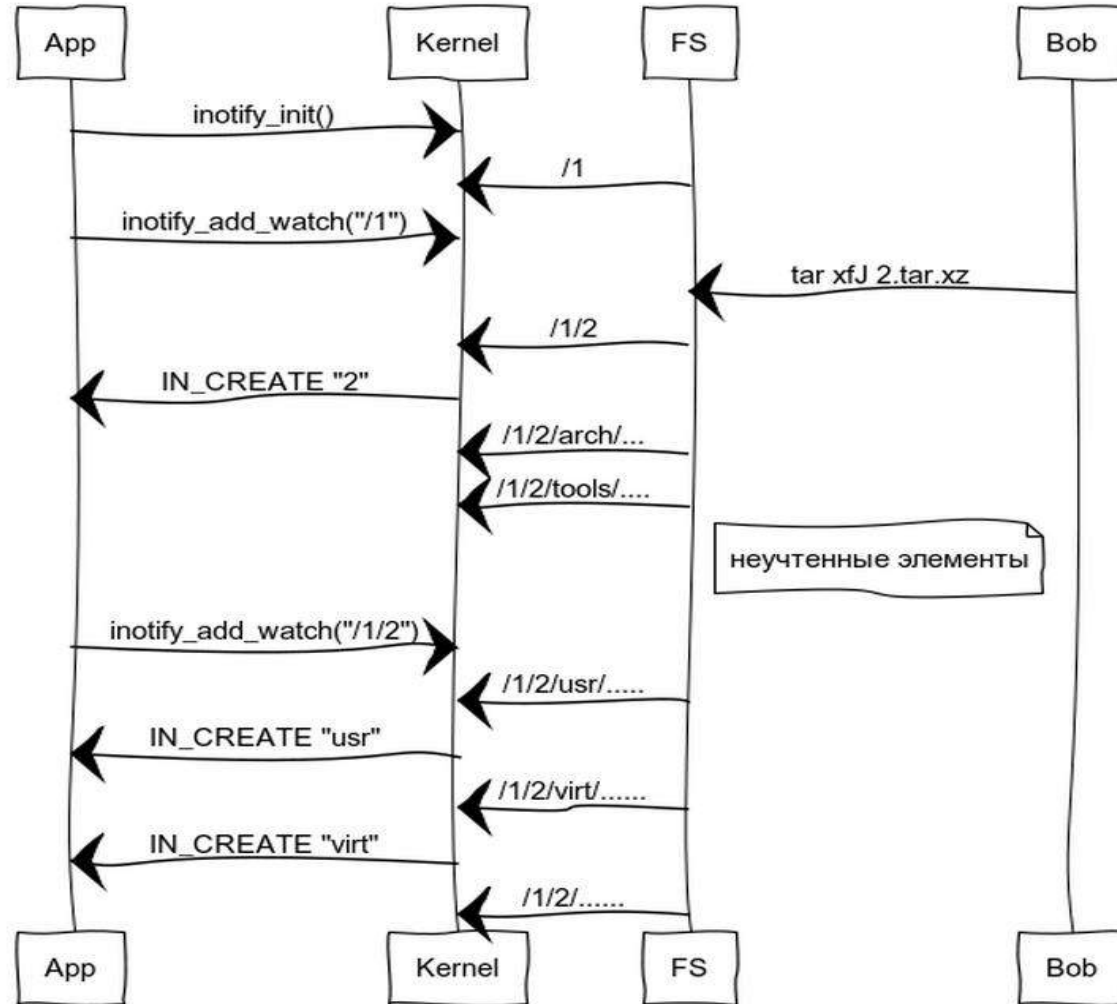
```
int inotify_add_watch(int fd, const char* path, uint32_t mask); // возвращает wd
```

```
Int inotify_rm_watch(int fd, uint32_t wd);
```

# Структура inotify\_event

```
struct inotify_event {  
    int wd; /* Watch descriptor */  
    uint32_t mask; /* Mask of events */  
    uint32_t cookie; /* Unique cookie associating related events */  
    uint32_t len; /* Size of name field */  
    char name[]; /* Optional null-terminated name */  
};
```

# Схема работы с inotify





www.itspartner.net

<https://github.com/zmushko/librnotify>

```
Notify* initNotify(const char* path, uint32_t mask, const char* exclude);
```

```
int waitNotify(Notify* ntf, char** const path, uint32_t* mask, const int  
timeout, uint32_t* cookie);
```

```
void freeNotify(Notify* ntf);
```

# Рекомендации

- Поток нотификаций.
- Перемещение папки за пределы нотификаций.
- FTP, SMB etc.
- Подавление эха.





www.itspartner.net

Вопросы

**Спасибо за внимание**

**Змушко Андрей**

[https://github.com / zmushko](https://github.com/zmushko)