



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления»  
КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

## Отчёт по лабораторной работе №6

Название «Системный вызов open()»

---

Дисциплина «Операционные системы»

---

Студент ИУ7-65Б

---

(подпись, дата)

Клименко А.К.  
(Фамилия И.О.)

Преподаватель

(подпись, дата)

Рязанова Н.Ю.  
(Фамилия И.О.)

Москва, 2022

# 1 Описание структур

```
1 struct audit_names;
2 struct filename {
3     const char      *name; /* pointer to actual string */
4     const __user char *uptr; /* original userland pointer */
5     int             refcnt;
6     struct audit_names *aname;
7     const char      iname[];
8 };
9 static_assert(offsetof(struct filename, iname) % sizeof(long) == 0);
```

```
1 /* When fs/namei.c:getname() is called, we store the pointer in name and bump
2  * the refcnt in the associated filename struct.
3  *
4  * Further, in fs/namei.c:path_lookup() we store the inode and device.
5  */
6 struct audit_names {
7     struct list_head    list; /* audit_context->names_list */
8
9     struct filename     *name;
10    int                  name_len; /* number of chars to log */
11    bool                 hidden; /* don't log this record */
12
13    unsigned long        ino;
14    dev_t                dev;
15    umode_t              mode;
16    kuid_t               uid;
17    kgid_t               gid;
18    dev_t                rdev;
19    u32                  osid;
20    struct audit_cap_data fcap;
21    unsigned int          fcap_ver;
22    unsigned char         type; /* record type */
23    /*
24     * This was an allocated audit_names and not from the array of
25     * names allocated in the task audit context. Thus this name
26     * should be freed on syscall exit.
27     */
28    bool                 should_free;
29 };
```

```

1 struct open_flags {
2     int      open_flag;
3     umode_t   mode;
4     int      acc_mode;
5     int      intent;
6     int      lookup_flags;
7 };

```

```

1 #define EMBEDDED_LEVELS 2
2 struct nameidata {
3     struct path    path;
4     struct qstr     last;
5     struct path     root;
6     struct inode    *inode; /* path.dentry.d_inode */
7     unsigned int    flags, state;
8     unsigned        seq, m_seq, r_seq;
9     int             last_type;
10    unsigned         depth;
11    int              total_link_count;
12    struct saved {
13        struct path link;
14        struct delayed_call done;
15        const char *name;
16        unsigned seq;
17    } *stack, internal[EMBEDDED_LEVELS];
18    struct filename *name;
19    struct nameidata *saved;
20    unsigned         root_seq;
21    int              dfd;
22    kuid_t           dir_uid;
23    umode_t          dir_mode;
24 } __randomize_layout;

```

## 2 Схема алгоритма функции open





















