



2^η Εργαστηριακή Άσκηση Λειτουργικών Συστημάτων

☼ Μάθημα : Λειτουργικά Συστήματα(MYY601)

☼ Διδάσκων : Αναστασιάδης Στέργιος

☼ Υπεύθυνος Εργαστηρίου : Κυριαζής Ιωάννης

☼ Βοηθοί : Καζαντζίδης Παναγιώτης | | Παπακώστας
Ιωάννης

☼ Μέλη ομάδας/ΑΜ : Καζακίδης Κωνσταντίνος/4065

Καζακίδης Θεοχάρης/4679

Μητρόπουλος Γεώργιος/4733

☼ Ακαδημαϊκό Έτος: 2022-2023



Περιεχόμενα

Κεφάλαιο 1.

Εισαγωγή.....	4
1.1 Σκοπός της αναφοράς.....	4
1.2 Περιγραφή των ζητούμενων της άσκησης.....	4

Κεφάλαιο 2.

Κατανόηση λειτουργίας του συστήματος αρχείων FAT.....	5
2.1 Προετοιμασία.....	5
2.2 Το struct super_operations fat_sops.....	5
2.3 Το struct address_space_operations fat_aops.....	11
2.4 Τα struct fatent_operations fat12/16/32_ops.....	11
2.5 Το struct file_operations fat_file_operations.....	13
2.6 Το struct inode_operations fat_file_inode_operations	13
2.7 Το struct inode_operations msdos_dir_inode_operations	14

Κεφάλαιο 3.

Αρχείο καταγραφής.....	15
3.1 Εισαγωγή.....	15
3.2 Παρουσίαση στοιχείων που θα περιέχει.....	17
3.3 Υλοποίηση.....	20

Κεφάλαιο 4

Έλεγχος ορθότητας	27
4.1 Compile.....	27
4.2 Tests.....	30
4.3 Εφαρμογές	33

Εισαγωγή

1.1 Σκοπός της αναφοράς

Η παρακάτω αναφορά έχει ως στόχο να επεξηγήσει τον κώδικα που υλοποιήσαμε στα πλαίσια της άσκησης. Συγκεκριμένα θα αναφέρουμε τόσο τις λειτουργίες που υλοποιήσαμε όσο και τον κώδικα που προσθέσαμε στα αρχεία που περιέχονται στην εικονική μηχανή. Ακόμη ,παρακάτω επισημαίνονται αναλυτικά οι αλλαγές στα αρχεία και παρουσιάζονται οι εκτελέσεις δοκιμαστικών προγραμμάτων (tests) και ενσωματωμένων εφαρμογών της LKL.

1.2 Περιγραφή των ζητούμενων της άσκησης

Μας δίνεται η βιβλιοθήκη Linux Kernel Library η οποία αποτελεί μια υλοποίηση του πυρήνα του Linux που εκτελείται σε επίπεδο χρήστη, δηλαδή τρέχει μέσα σε μια διεργασία. Η LKL υλοποιεί το σύστημα αρχείων FAT. Το πρώτο ζητούμενο είναι η κατανόηση της λειτουργίας του συστήματος αρχείων FAT βρίσκοντας τις συναρτήσεις που εμπλέκονται στην προσπέλαση αρχείων και κατόπιν τυπώνοντας στην κονσόλα μηνύματα όταν η ροή του προγράμματος φτάνει σε αυτές τις συναρτήσεις. Το δεύτερο ζητούμενο αφορά την υλοποίηση μηχανισμού καταγραφής δηλαδή αρχείου που περιέχει χρήσιμα δεδομένα/μεταδεδομένα για τα αρχεία του συστήματος αρχείου. Τέλος με την χρήση προγραμμάτων ελέγχου που προσφέρει η LKL πρέπει να δείξουμε την ορθότητα της υλοποίησής μας.

Κατανόηση λειτουργίας του συστήματος αρχείων FAT

2.1 Προετοιμασία

Αφού εγκαταστήσαμε την εικονική μηχανή αρχικά ακολουθήσαμε τις οδηγίες που δίνονται στην εκφώνηση για την μεταγλώττιση της βιβλιοθήκης LKL. Αφού ολοκληρώσαμε την μεταγλώττιση πειραματιστήκαμε με την εκτέλεση των tests στον κατάλογο lkl tools.

Τρέξαμε τα tests:

- `./cptofs -i /tmp/vfatfile -p -t vfat lklfuse.c/` → Δοκιμάζουμε το copy προς το file system
- `./cpfromfs -i /tmp/vfatfile -p -t vfat lklfuse.c/` → Δοκιμάζουμε το copy προς από file system
- `./fs2tar /tmp/vfatfile -p -t vfat lklfuse.c/` → Μετατρέπει μια εικόνα του fs αρχείο σε αρχείο tar
- `./boot -t vfat -d /tmp/vfatfile -p -P 0` → Έλεγχοι εκκίνησης του συστήματος FAT

Και έτσι κατανοήσαμε και εξοικειωθήκαμε με τις εντολές αυτές. Στις ενότητες 4.2/4.3 μετά την τροποποίηση των αρχείων φαίνεται ότι τα παραπάνω tests εξακολουθούν να επιτυγχάνουν.

2.2 Το struct struct super_operations fat_sops

Η πλοήγηση στον κώδικα ξεκίνησε από τα αρχεία που περιέχουν τα structs που αναφέρονται στην εκφώνηση δηλαδή τα αρχεία `inode.c`(λειτουργίες superblock και λειτουργίες μνήμης) `fatent.c`(λειτουργίες εγγραφών FAT) `file.c`(λειτουργίες εγγραφών αρχείου και μερικές λειτουργίες inode) `namei_msdos.c`(λειτουργίες καταλόγων για συστήματα αρχείων FAT) `namei_vsfat.c` (λειτουργίες καταλόγων για συστήματα αρχείων VFAT).Μελετώντας τα struct παρατηρήσαμε ότι τα πεδία τους αρχικοποιούνται από κάποια συνάρτηση

Στόχος είναι να βρούμε πού είναι υλοποιημένες αυτές οι συναρτήσεις, προκειμένου να κατανοήσουμε την ροή του προγράμματος. Η πλειοψηφία των συναρτήσεων αυτών ήταν υλοποιημένες στο ίδιο αρχείο με το struct που τις περιλάμβανε. Στις περιπτώσεις που αυτές οι συναρτήσεις δεν ορίζονται στο ίδιο

αρχείο με το struct τις ψάχνουμε με την εντολή **grep -R “όνομα συνάρτησης”** στο τερματικό. Για κάθε συνάρτηση που βρήκαμε προσθέσαμε ένα `printk(KERN_NFO “entering ‘function name’ at ‘file name’”)` στην αρχή για να δείξουμε ότι μπαίνει και ένα `printk(KERN_NFO “exiting ‘function name’ at ‘file name’”)` στο τέλος για να δείξουμε ότι βγαίνει λαμβάνοντας υπόψη όλες τις διαφορετικές περιπτώσεις που επιστρέφει η συνάρτηση.

Συγκεκριμένα

Οι λειτουργίες του **superblock** ορίζονται στην δομή **struct super_operations fat_sops** του αρχείου **fs/fat/inode.c**

Όπου όλες οι συναρτήσεις ορίζονται βρίσκονται στο αρχείο `lkl-source/fs/fat/inode.c`

```
static int fat_show_options(struct seq_file *m, struct dentry *root);
static const struct super_operations fat_sops = {
    .alloc_inode      = fat_alloc_inode,
    .destroy_inode    = fat_destroy_inode,
    .write_inode      = fat_write_inode,
    .evict_inode      = fat_evict_inode,
    .put_super        = fat_put_super,
    .statfs           = fat_statfs,
    .remount_fs       = fat_remount,

    .show_options     = fat_show_options,
};
```

Το πεδίο **.alloc_inode** αρχικοποιείται από την συνάρτηση **fat_alloc_inode** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static struct inode *fat_alloc_inode(struct super_block *sb)
{
    printk(KERN_INFO "entering fat_alloc_inode in inode.c");
    struct msdos_inode_info *ei;
    ei = kmem_cache_alloc(fat_inode_cache, GFP_NOFS);
    if (!ei)
    {
        printk(KERN_INFO "exiting fat_alloc_inode in inode.c");

        return NULL;
    }
    init_rwsem(&ei->truncate_lock);
    printk(KERN_INFO "exiting fat_alloc_inode in inode.c");
    return &ei->vfs_inode;
}
```

Το πεδίο **.destroy_inode** αρχικοποιείται από την συνάρτηση **fat_destroy_inode** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static void fat_destroy_inode(struct inode *inode)
{
    printk(KERN_INFO "entering fat_destroy_inode in inode.c");
    call_rcu(&inode->i_rcu, fat_i_callback);
    printk(KERN_INFO "exiting fat_destroy_inode in inode.c");
}
```

Το πεδίο **.write_inode** αρχικοποιείται από την συνάρτηση **fat_write_inode** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```

static int fat_write_inode(struct inode *inode, struct writeback_control *wbc)
{
    printk(KERN_INFO "entering fat_write_inode in inode.c");
    int err;

    if (inode->i_ino == MSDOS_FSINFO_INO) {
        struct super_block *sb = inode->i_sb;

        mutex_lock(&MSDOS_SB(sb)->s_lock);
        err = fat_clusters_flush(sb);
        mutex_unlock(&MSDOS_SB(sb)->s_lock);
    } else
        err = __fat_write_inode(inode, wbc->sync_mode == WB_SYNC_ALL);

    printk(KERN_INFO "exiting fat_write_inode in inode.c");
    return err;
}

```

Το πεδίο `.evict_inode` αρχικοποιείται από την συνάρτηση `fat_evict_inode` που βρίσκεται στο αρχείο `fs/fat/inode.c`

```

static void fat_evict_inode(struct inode *inode)
{
    printk(KERN_INFO "entering fat_evict_inode in inode.c");
    truncate_inode_pages_final(&inode->i_data);
    if (!inode->i_nlink) {
        inode->i_size = 0;
        fat_truncate_blocks(inode, 0);
    } else
        fat_free_eofblocks(inode);

    invalidate_inode_buffers(inode);
    clear_inode(inode);
    fat_cache_inval_inode(inode);
    fat_detach(inode);
    printk(KERN_INFO "exiting fat_evict_inode in inode.c");
}

```


Το πεδίο **.put_super** αρχικοποιείται από την συνάρτηση **fat_put_super** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static void fat_put_super(struct super_block *sb)
{
    printk(KERN_INFO "entering fat_put_super in inode.c");
    struct msdos_sb_info *sbi = MSDOS_SB(sb);

    fat_set_state(sb, 0, 0);

    iput(sbi->fsinfo_inode);
    iput(sbi->fat_inode);

    call_rcu(&sbi->rcu, delayed_free);
    printk(KERN_INFO "exiting fat_put_super in inode.c");
}
```

Το πεδίο **.statfs** αρχικοποιείται από την συνάρτηση **fat_statfs** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static int fat_statfs(struct dentry *dentry, struct kstatfs *buf)
{
    printk(KERN_INFO "entering fat_statfs in inode.c");
    struct super_block *sb = dentry->d_sb;
    struct msdos_sb_info *sbi = MSDOS_SB(sb);
    u64 id = huge_encode_dev(sb->s_bdev->bd_dev);

    /* If the count of free cluster is still unknown, counts it here. */
    if (sbi->free_clusters == -1 || !sbi->free_clus_valid) {
        int err = fat_count_free_clusters(dentry->d_sb);
        if (err)
        {
            printk(KERN_INFO "exiting fat_statfs in inode.c");
            return err;
        }
    }
}
```

Το πεδίο **.remount_fs** αρχικοποιείται από την συνάρτηση **fat_remount** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static int fat_remount(struct super_block *sb, int *flags, char *data)
{
    printk(KERN_INFO "entering fat_remount in inode.c");
    int new_rdonly;
    struct msdos_sb_info *sbi = MSDOS_SB(sb);
    *flags |= MS_NODIRATIME | (sbi->options.isvfat ? 0 : MS_NOATIME);

    sync_filesystem(sb);

    /* make sure we update state on remount. */
    new_rdonly = *flags & MS_RDONLY;
    if (new_rdonly != (sb->s_flags & MS_RDONLY)) {
        if (new_rdonly)
            fat_set_state(sb, 0, 0);
        else
            fat_set_state(sb, 1, 1);
    }
    printk(KERN_INFO "exiting fat_remount in inode.c");
    return 0;
}
```

Το πεδίο **.show_options** αρχικοποιείται από την συνάρτηση **fat_show_options** που βρίσκεται στο αρχείο **fs/fat/inode.c**

```
static int fat_show_options(struct seq_file *m, struct dentry *root)
{
    printk(KERN_INFO "entering fat_show_options in inode.c");
    struct msdos_sb_info *sbi = MSDOS_SB(root->d_sb);
    struct fat_mount_options *opts = &sbi->options;
    int isvfat = opts->isvfat;

    .
    .
    .
    lines(961-1044)
    .
    .
    .
    printk(KERN_INFO "exiting fat_show_options in inode.c");
    return 0;
}
```

Παρόμοια διαδικασία υλοποιήσαμε και για τα πεδία των παρακάτω struct όμως το αναφέρουμε συνοπτικά λόγω έκτασης της αναφοράς

2.3 Το struct address_space_operations fat_aops

Οι λειτουργίες της **μνήμης** ορίζονται στην δομή **struct address_space_operations fat_aops** του αρχείου **fs/fat/inode.c**

Οπου όλες οι συναρτήσεις ορίζονται βρίσκονται στο αρχείο **lkl-source/fs/fat/inode.c**

```
static const struct address_space_operations fat_aops = {  
    .readpage      = fat_readpage,  
    .readpages     = fat_readpages,  
    .writepage     = fat_writepage,  
    .writepages    = fat_writepages,  
    .write_begin   = fat_write_begin,  
    .write_end     = fat_write_end,  
    .direct_IO     = fat_direct_IO,  
    .bmap          = _fat_bmap  
};
```

2.4 Τα struct fatent_operations fat12/16/32_ops

Οι λειτουργίες των **εγγραφών FAT** ορίζονται στις δομές **struct fatent_operations fat12_ops** | **struct fatent_operations fat16_ops** | **struct fatent_operations fat32_ops**. Και οι τρεις δομές βρίσκονται μέσα στο αρχείο **fs/fat/fatent.c**

Οπου όλες οι συναρτήσεις ορίζονται βρίσκονται στο αρχείο **lkl-source/fs/fat/fatent.c**

```
static const struct fatent_operations fat12_ops = {  
    .ent_blocknr    = fat12_ent_blocknr,  
    .ent_set_ptr    = fat12_ent_set_ptr,  
    .ent_bread     = fat12_ent_bread,  
    .ent_get       = fat12_ent_get,  
    .ent_put       = fat12_ent_put,  
    .ent_next      = fat12_ent_next,  
};
```

```
static const struct fatent_operations fat16_ops = {  
    .ent_blocknr    = fat_ent_blocknr,  
    .ent_set_ptr    = fat16_ent_set_ptr,  
    .ent_bread      = fat_ent_bread,  
    .ent_get= fat16_ent_get,  
    .ent_put= fat16_ent_put,  
    .ent_next       = fat16_ent_next,  
};
```

```
static const struct fatent_operations fat32_ops = {  
    .ent_blocknr    = fat_ent_blocknr,  
    .ent_set_ptr    = fat32_ent_set_ptr,  
    .ent_bread      = fat_ent_bread,  
    .ent_get= fat32_ent_get,  
    .ent_put= fat32_ent_put,  
    .ent_next       = fat32_ent_next,  
};
```

2.5 Το struct file_operations fat_file_operations

Οι λειτουργίες του **αρχείου** ορίζονται στην δομή **struct file_operations** **fat_file_operations** του αρχείου **fs/fat/file.c**

Όπου οι συναρτήσεις ορίζονται :

- generic_file_llseek → (βρίσκεται στο αρχείο lkl-source/fs/read_write.c)
- generic_file_read_iter || generic_file_write_iter || generic_file_mmap → (βρίσκονται στο αρχείο lkl-source/mm/filemap.c)
- fat_file_release || fat_generic_ioctl || fat_file_fsync || fat_generic_compat_ioctl || fat_fallocate → (βρίσκονται στο αρχείο lkl-source/fs/fat/file.c)
- generic_file_splice_read → (βρίσκεται στο αρχείο lkl-source/fs/splice.c)

```
const struct file_operations fat_file_operations = {
    .llseek      = generic_file_llseek,
    .read_iter   = generic_file_read_iter,
    .write_iter  = generic_file_write_iter,
    .mmap        = generic_file_mmap,
    .release     = fat_file_release,
    .unlocked_ioctl = fat_generic_ioctl,
#ifdef CONFIG_COMPAT
    .compat_ioctl = fat_generic_compat_ioctl,
#endif
    .fsync       = fat_file_fsync,
    .splice_read = generic_file_splice_read,
    .fallocate   = fat_fallocate,
};
```

2.6 Το struct inode_operations fat_file_inode_operations

Μερικές λειτουργίες inode ορίζονται στην δομή **struct inode_operations** **fat_file_inode_operations** του αρχείου **fs/fat/file.c**

Όπου όλες οι συναρτήσεις ορίζονται βρίσκονται στο αρχείο lkl-source/fs/fat/file.c

```
const struct inode_operations fat_file_inode_operations = {
    .setattr = fat_setattr,
    .getattr = fat_getattr,
};
```

2.7 Το struct inode_operations msdos_dir_inode_operations

Οι λειτουργίες των **καταλόγων για το σύστημα αρχείων FAT** ορίζονται στην δομή **struct inode_operations msdos_dir_inode_operations** του αρχείου **fs/fat/namei_msdos.c**

Όπου οι συναρτήσεις ορίζονται :

- `msdos_create` | `msdos_lookup` | `msdos_unlink` | `msdos_mkdir` | `msdos_rmdir` | `msdos_rename` → (βρίσκονται στο αρχείο `lkl-source/fs/fat/namei_msdos.c`)
- `fat_setattr` | `fat_getattr` → (βρίσκονται στο αρχείο `lkl-source/fs/fat/file.c`)

```
static const struct inode_operations msdos_dir_inode_operations = {
    .create      = msdos_create,
    .lookup      = msdos_lookup,
    .unlink      = msdos_unlink,
    .mkdir       = msdos_mkdir,
    .rmdir       = msdos_rmdir,
    .rename      = msdos_rename,
    .setattr     = fat_setattr,
    .getattr     = fat_getattr,
};
```

Αρχείο καταγραφής

3.1 Εισαγωγή

Σκοπός μας είναι να αποθηκεύσουμε τα μεταδεδομένα που χρειαζόμαστε στο αρχείο καταγραφής. Τα μεταδεδομένα αυτά μετά από αναζήτηση τα ανακαλύψαμε στα εξής struct:

- **struct msdos_sb_info** { → βρίσκεται στο αρχείο `lkl-source/fs/fat/fat.h`)
- **struct fat_boot_sector** { → βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)
- **struct fat_boot_fsinfo** { → βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)
- **struct msdos_dir_entry** { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)
- **struct msdos_dir_slot** { → βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)
- **struct inode** { → βρίσκεται στο αρχείο `lkl-source/include/linux/msdos_fs.h`)

Μελετώντας τα παραπάνω αποφασίσαμε ότι το αρχείο καταγραφής πρέπει να έχει την εξής πληροφορία:

- Δεδομένα

Τα περιεχόμενα των αρχείων που ενδιαφέρουν τους χρήστες

- Μεταδεδομένα
- Χαρακτηριστικά που αφορούν κάθε αρχείο, π.χ.:
 - Δείκτες στα block του δίσκου που βρίσκονται τα δεδομένα
 - Μέγεθος
 - Δικαιώματα πρόσβασης
 - Ιδιοκτήτης
 - Χρόνος δημιουργίας, τελευταίας προσπέλασης/τροποποίησης
 - Χαρακτηριστικά που αφορούν συνολικά το σύστημα αρχείων, π.χ.:
 - Τύπος συστήματος αρχείων
 - Αριθμός block/ inode
 - Μέγεθος block

Αρχικά για να ορίσουμε το αρχείο καταγραφής στην **γγραμμή 88** του **lkl-source/fs/fat/fat.h** μέσα στο **struct msdos_sb_info** ορίζουμε το **int journal** που θα είναι ο file descriptor για το αρχείο καταγραφής.

Μετά στην **γγραμμή 1774** του **lkl-source/fs/fat/inode.c** εντός της συνάρτησης **fat_fill_super** η οποία αρχικοποιεί το superblock ανοίγω το αρχείο καταγραφής.

Σε κάθε αρχείο που περιέχει εντολές σχετικά με το αρχείο καταγραφής κάνουμε **#include** το header file **<linux/syscalls.h>** έτσι ώστε να μπορούμε να χρησιμοποιήσουμε τις συναρτήσεις **sys_write**, **sys_open** και **sys_fsync**.

Σε κάθε συνάρτηση μέσα στην οποία έχουμε προσθέσει κώδικα σχετικό με το αρχείο καταγραφής δηλώνουμε ένα **char str[100]** έτσι ώστε να κρατάμε τα στοιχεία που θέλουμε να αποθηκευτούν στο αρχείο.

Για να προσθέσουμε ένα στοιχείο στο αρχείο καταγραφής κάνουμε την εξής διαδικασία:

- 1) `sprintf(str, "τι προσθέτω", 'μεταβλητή που περιέχει αυτό που προσθέτω')`
- 2) `printk(KERN_INFO "%s", str)`
- 3) `sys_write(sbi->journal, str, sizeof(str));`
- 4) `sys_fsync(sbi->journal)` (αν κάνω πολλά `sys_write` συνεχόμενα το βάζω μια φορά στο τέλος)

Σημειώνουμε ότι η συνάρτηση `sys_fsync` χρησιμοποιείται διότι τα περιεχόμενα του αρχείου καταγραφής βρίσκονται στην μνήμη έως ότου να κλείσει το αρχείο καταγραφής και να μεταφερθούν στο δίσκο. Επειδή το σύστημα αρχείων FAT μπορεί να υποπέσει σε σφάλμα απροσδόκιτα και να διακοπεί η λειτουργία του δεν θέλουμε να χάσουμε τα στοιχεία του αρχείου καταγραφής που έχουμε αποθηκεύσει μέχρι εκείνη την στιγμή. Άρα με την εντολή `sys_fsync` τα στοιχεία του αρχείου καταγραφής θα γίνονται flush από την μνήμη στον δίσκο χωρίς να πρέπει να έχει κλείσει το αρχείο καταγραφής.

3.2 Παρουσίαση στοιχείων που θα περιέχει

```
struct msdos_sb_info { →βρίσκεται στο αρχείο lkl-
source/fs/fat/fat.h )
    unsigned short sec_per_clus; /* sectors/cluster */
    unsigned short cluster_bits; /* log2(cluster_size) */
    unsigned int cluster_size; /* cluster size */ → μέγεθος του cluster
    unsigned char fats, fat_bits; /* number of FATs, FAT bits (12,16 or 32) */
    unsigned short fat_start;
    unsigned long fat_length; /* FAT start & length (sec.) */
    unsigned long dir_start;
    unsigned short dir_entries; /* root dir start & entries */
    unsigned long data_start; /* first data sector */
    unsigned long max_cluster; /* maximum cluster number */
    unsigned long root_cluster; /* first cluster of the root directory */
    unsigned long fsinfo_sector; /* sector number of FAT32 fsinfo */
    struct mutex fat_lock;
    struct mutex nfs_build_inode_lock;
    struct mutex s_lock;
    unsigned int prev_free; /* previously allocated cluster number */

    unsigned int free_clusters; /* -1 if undefined */
    unsigned int free_clus_valid; /* is free_clusters valid? */

    struct fat_mount_options options;
    struct nls_table *nls_disk; /* Codepage used on disk */
    struct nls_table *nls_io; /* Charset used for input and display */
    const void *dir_ops; /* Opaque; default directory operations */
    int dir_per_block; /* dir entries per block */
    int dir_per_block_bits; /* log2(dir_per_block) */
    unsigned int vol_id; /*volume ID*/

    int fatent_shift;
    const struct fatent_operations *fatent_ops;
    struct inode *fat_inode;
    struct inode *fsinfo_inode;

    struct ratelimit_state ratelimit;

    spinlock_t inode_hash_lock;
    struct hlist_head inode_hashtable[FAT_HASH_SIZE];

    spinlock_t dir_hash_lock;
    struct hlist_head dir_hashtable[FAT_HASH_SIZE];

    unsigned int dirty; /* fs state before mount */
    struct rcu_head rcu;
};
```

struct fat_boot_sector { → (βρίσκεται στο αρχείο lkl-source/include/uapi/linux/msdos_fs.h)

```
__u8   ignored[3];      /* Boot strap short or near jump */
__u8   system_id[8];    /* Name - can be used to special case
                        partition manager volumes */
__u8   sector_size[2];  /* bytes per logical sector */
__u8   sec_per_clus;    /* sectors/cluster */
__le16 reserved; /* reserved sectors */
__u8   fats;            /* number of FATs */
__u8   dir_entries[2];  /* root directory entries */
__u8   sectors[2];     /* number of sectors */
__u8   media;          /* media code */
__le16 fat_length;     /* sectors/FAT */
__le16 secs_track;     /* sectors per track */
__le16 heads;          /* number of heads */
__le32 hidden;         /* hidden sectors (unused) */
__le32 total_sect;     /* number of sectors (if sectors == 0) */

union {
    struct {
        /* Extended BPB Fields for FAT16 */
        __u8   drive_number; /* Physical drive number */
        10 ok
        __u8 state; /* undocumented, but used
                    for mount state. */
        __u8   signature; /* extended boot signature */
        __u8   vol_id[4]; /* volume ID */
        __u8   vol_label[11]; /* volume label */
        __u8   fs_type[8]; /* file system type */
        /* other fields are not added here */
    } fat16;

    struct {
        /* only used by FAT32 */
        __le32 length; /* sectors/FAT */
        __le16 flags; /* bit 8: fat mirroring,
                    low 4: active fat */
        __u8   version[2]; /* major, minor filesystem
                    version */
        __le32 root_cluster; /* first cluster in
                    root directory */
        __le16 info_sector; /* filesystem info sector */
        __le16 backup_boot; /* backup boot sector */
        __le16 reserved2[6]; /* Unused */
        /* Extended BPB Fields for FAT32 */
        __u8   drive_number; /* Physical drive number */
        __u8 state; /* undocumented, but used
                    for mount state. */
        __u8   signature; /* extended boot signature */
        __u8   vol_id[4]; /* volume ID */
        __u8   vol_label[11]; /* volume label */
        __u8   fs_type[8]; /* file system type */
        /* other fields are not added here */
    } fat32;
};
};
```

struct fat_boot_finfo { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

```
__le32 signature1; /* 0x41615252L */
__le32 reserved1[120]; /* Nothing as far as I can tell */
__le32 signature2; /* 0x61417272L */
__le32 free_clusters; /* Free cluster count. -1 if unknown */ ελεύθερων cluster
__le32 next_cluster; /* Most recently allocated cluster */
__le32 reserved2[4];
};
```

struct msdos_dir_entry { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

```
__u8 name[MSDOS_NAME]; /* name and extension */
__u8 attr; /* attribute bits */
__u8 lcase; /* Case for base and extension */
__u8 ctime_cs; /* Creation time, centiseconds (0-199) */
__le16 ctime; /* Creation time */
__le16 cdate; /* Creation date */
__le16 adate; /* Last access date */
__le16 starthi; /* High 16 bits of cluster in FAT32 */
__le16 time,date,start; /* time, date and first cluster */
__le32 size; /* file size (in bytes) */
};
```

struct msdos_dir_slot { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

```
__u8 id; /* sequence number for slot */
__u8 name0_4[10]; /* first 5 characters in name */
__u8 attr; /* attribute byte */
__u8 reserved; /* always 0 */
__u8 alias_checksum; /* checksum for 8.3 alias */
__u8 name5_10[12]; /* 6 more characters in name */
__le16 start; /* starting cluster number, 0 in long slots */
__u8 name11_12[4]; /* last 2 characters in name */
};
```

struct fat_boot_fsinfo { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

```
umode_t          i_mode;
unsigned short    i_opflags;
kuid_t           i_uid;
kgid_t           i_gid;
unsigned int      i_flags;
.....
.....
}; έχει και άλλα πολλά που δεν μας νοιάζουν
```

Τα στοιχεία που είναι **χρωματισμένα** είναι τα στοιχεία που θα περιέχονται στο αρχείο καταγραφής

3.3 Υλοποίηση

- **struct msdos_sb_info {** → (βρίσκεται στο αρχείο `lkl-source/fs/fat/fat.h`)

1)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει τον υπολογισμό των bits που καταλαμβάνει το cluster size

Αυτό αποθηκεύεται στο **unsigned short cluster_bits; /* log2(cluster_size) */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο `lkl-source/fs/fat/fat.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1779-1781** `lkl-source/fs/fat/inode.c`

2)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει τον υπολογισμό του πλήθους cluster

Αυτό αποθηκεύεται στο **unsigned int cluster_size; /* cluster size */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο `lkl-source/fs/fat/fat.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1776-1778** `lkl-source/fs/fat/inode.c`

3)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση τύπου FAT που είναι (12/16/32)

Αυτό αποθηκεύεται στο **unsigned char fats, fat_bits; /* number of FATs, FAT bits (12,16 or 32) */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο `lkl-source/fs/fat/fat.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1782-1784 & line 1834-1836** `lkl-source/fs/fat/inode.c`

4)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση offset σε sectors από την αρχή του δίσκου

Αυτό αποθηκεύεται στο **unsigned short fat_start;** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1785-1787 lkl-source/fs/fat/inode.c**

5)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση του μέγεθος συστήματος σε sectors

Αυτό αποθηκεύεται στο **unsigned long fat_length;** **/* FAT start & length (sec.)** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1788-1790 lkl-source/fs/fat/inode.c**

6)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση του 1ος cluster του root directory

Αυτό αποθηκεύεται στο **unsigned long root_cluster;** **/* first cluster of the root directory */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1837-1839 lkl-source/fs/fat/inode.c**

7)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση του τελευταίου δεσμευμένου cluster

Αυτό αποθηκεύεται στο **unsigned int prev_free;** **/* previously allocated cluster number */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1843-1845 lkl-source/fs/fat/inode.c**

8)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την εύρεση του αν υπάρχουν ελεύθερα clusters

Αυτό αποθηκεύεται στο **unsigned int free_clusters;** **/* -1 if undefined */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1840-1842 lkl-source/fs/fat/inode.c**

9)Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει τον έλεγχο για το αν ισχύει ότι υπάρχουν ελεύθερα clusters

Αυτό αποθηκεύεται στο **unsigned int free_clus_valid;** **/* is free_clusters valid? */** στο **struct msdos_sb_info** που βρίσκεται στο αρχείο **lkl-source/fs/fat/fat.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 1791-1794 lkl-source/fs/fat/inode.c**

- **struct fat_boot_sector** { → (βρίσκεται στο αρχείο lkl-source/include/uapi/linux/msdos_fs.h)

10) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για κατάσταση του FAT16

Αυτό αποθηκεύεται στο **__u8 state; /* undocumented, but used for mount state. */**

στο **struct fat_boot_sector** που βρίσκεται στο αρχείο **lkl-source/include/uapi/linux/msdos_fs.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 719-722 lkl-source/fs/fat/inode.c**

11) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για κατάσταση του FAT32

Αυτό αποθηκεύεται στο **__u8 state; /* undocumented, but used for mount state. */**

στο **struct fat_boot_sector** που βρίσκεται στο αρχείο **lkl-source/include/uapi/linux/msdos_fs.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 709-712 lkl-source/fs/fat/inode.c**

- **struct fat_boot_fsinfo** { → (βρίσκεται στο αρχείο lkl-source/include/uapi/linux/msdos_fs.h)

12) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη του αριθμού ελεύθερων cluster

Αυτό αποθηκεύεται στο **__le32 free_clusters; /* Free cluster count. -1 if unknown */** στο **struct fat_boot_fsinfo** που βρίσκεται στο αρχείο **lkl-source/include/uapi/linux/msdos_fs.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 88-91 lkl-source/fs/fat/misc.c**

13) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη του τελευταίο δεσμευμένο cluster

Αυτό αποθηκεύεται στο **__le32 next_cluster; /* Most recently allocated cluster */** στο **struct fat_boot_fsinfo** που βρίσκεται στο αρχείο **lkl-source/include/uapi/linux/msdos_fs.h**

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 96-99 lkl-source/fs/fat/misc.c**

- **struct msdos_dir_entry** { → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

14) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη του όνομα και τύπος αρχείου

Αυτό αποθηκεύεται στο `__u8 name[MSDOS_NAME]; /* name and extension */` στο **struct msdos_dir_entry** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 683-685** `lkl-source/fs/fat/namei_vfat.c`

15) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των μεταδεδομένα για την συμπεριφορά του αρχείου

Αυτό αποθηκεύεται στο `__u8 attr; /* attribute bits */` στο **struct msdos_dir_entry** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 686-688** `lkl-source/fs/fat/namei_vfat.c` && **line 935-937** `lkl-source/fs/fat/inode.c`

16) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για το case sensitivity των FAT

Αυτό αποθηκεύεται στο `__u8 lcase; /* Case for base and extension */` στο **struct msdos_dir_entry** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 689-692** `lkl-source/fs/fat/namei_vfat.c`

17) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για το χρόνο δημιουργίας

Αυτό αποθηκεύεται στο `__le16 ctime; /* Creation time */` στο **struct msdos_dir_entry** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 944-946** `lkl-source/fs/fat/inode.c`

18) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για την ημερομηνία δημιουργίας

Αυτό αποθηκεύεται στο `__le16 cdate; /* Creation date */` στο **struct msdos_dir_entry** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 947-949** `lkl-source/fs/fat/inode.c`

19) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για την ημερομηνία τελευταίας προσπέλασης

Αυτό αποθηκεύεται στο `__le16 adate; /* Last access date */` στο `struct`

`msdos_dir_entry` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής `line 950-953 lkl-source/fs/fat/inode.c`

20) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για την ημερομηνία τελευταίας τροποποίησης

Αυτό αποθηκεύεται στο `__le16 time,date,start; /* time, date and first cluster */` στο `struct`

`msdos_dir_entry` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής `line 938-940 & line 941-943 lkl-source/fs/fat/inode.c`

21) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη για το μέγεθος αρχείου σε bytes

Αυτό αποθηκεύεται στο `__le32 size; /* file size (in bytes) */` στο `struct`

`msdos_dir_entry` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής `line 932-934 lkl-source/fs/fat/inode.c`

- `struct msdos_dir_slot {` → (βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`)

22) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη του id για τους πίνακες με το όνομα

Αυτό αποθηκεύεται στο `__u8 id; /* sequence number for slot */` στο `struct`

`msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής `line 647-649 lkl-source/fs/fat/namei_vfat.c`

23) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη τους πρώτοι 5 χαρακτήρες του ονόματος

Αυτό αποθηκεύεται στο `__u8 name0_4[10]; /* first 5 characters in name */` στο `struct`

`msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής `line 662-665 lkl-source/fs/fat/namei_vfat.c`

24) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των μεταδεδομένα για την συμπεριφορά του αρχείου
Αυτό αποθηκεύεται στο `__u8 attr; /* attribute byte */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 650-652 lkl-source/fs/fat/namei_vfat.c**

25) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των άχρηστων bits, (πρέπει να τα βάλουμε για λόγους συνέχειας)
Αυτό αποθηκεύεται στο `__u8 reserved; /* always 0 */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 653-655 lkl-source/fs/fat/namei_vfat.c**

26) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την αντιστοίχιση με το ψευδώνυμο του αρχείου
Αυτό αποθηκεύεται στο `__u8 alias_checksum; /* checksum for 8.3 alias */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 656-658 lkl-source/fs/fat/namei_vfat.c**

27) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των επόμενων 6 χαρακτήρων του ονόματος Αυτό αποθηκεύεται στο `__u8 name5_10[12]; /* 6 more characters in name */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 662-665 lkl-source/fs/fat/namei_vfat.c**

28) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των επόμενων 6 χαρακτήρων του ονόματος Αυτό αποθηκεύεται στο `__le16 start; /* starting cluster number, 0 in long slots */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 659-661 lkl-source/fs/fat/namei_vfat.c**

29) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των τελευταίων 2 χαρακτήρων του ονόματος
Αυτό αποθηκεύεται στο `__u8 name11_12[4]; /* last 2 characters in name */` στο `struct msdos_dir_slot` που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`
Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 662-665 lkl-source/fs/fat/namei_vfat.c**

- **struct inode** { → (βρίσκεται στο αρχείο `lkl-source/include/linux/msdos_fs.h`)

29) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των δικαιώματων πρόσβασης

Αυτό αποθηκεύεται στο **umode_t i_mode**; στο **struct inode** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 662-665 lkl-source/fs/fat/namei_vfat.c**

30) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των δικαιώματων πρόσβασης

Αυτό αποθηκεύεται στο **umode_t i_mode**; στο **struct inode** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 890-892 lkl-source/fs/fat/inode.c**

31) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη των ιδιοκτητών του αρχείου

Αυτό αποθηκεύεται στο **kuid_t i_uid**; στο **struct inode** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 893-895 lkl-source/fs/fat/inode.c**

32) Βρήκαμε ότι το FAT σύστημα αρχείων υποστηρίζει την ένδειξη της ομάδας ιδιοκτήτη του αρχείου

Αυτό αποθηκεύεται στο **kgid_t i_gid**; στο **struct inode** που βρίσκεται στο αρχείο `lkl-source/include/uapi/linux/msdos_fs.h`

Το πεδίο αυτό αρχικοποιείται και κατόπιν το προσθέτουμε στο αρχείο καταγραφής **line 896-899 lkl-source/fs/fat/inode.c**

Έλεγχος ορθότητας

4.1 Compile

- `make -C tools/lkl clean` → Κάνει clean

```
myy601@myy601lab2:~/lkl/lkl-source$ make -C tools/lkl clean
make: Entering directory '/home/myy601/lkl/lkl-source/tools/lkl'
  CLEAN      objects
  CLEAN      headers
  CLEAN      host libraries
  CLEAN      hijack library
  CLEAN      programs
  CLEAN      tests
make: Leaving directory '/home/myy601/lkl/lkl-source/tools/lkl'
myy601@myy601lab2:~/lkl/lkl-source$
```

- `make -j4 -C tools/lkl/` → Κάνει compile

```

my601@my601lab2:~/lkl/lkl-source$ make -j4 -C tools/lkl
make: Entering directory '/home/my601/lkl/lkl-source/tools/lkl'
scripts/kconfig/conf --defconfig Kconfig
kernel/time/Kconfig:155:warning: range is invalid
warning: (LKL) selects TCP_CONG_BBR which has unmet direct dependencies (NET && INET && TCP_CONG_ADVANCED)
warning: (LKL) selects TCP_CONG_BBR which has unmet direct dependencies (NET && INET && TCP_CONG_ADVANCED)
#
# configuration written to .config
#
scripts/kconfig/conf --silentoldconfig Kconfig
kernel/time/Kconfig:155:warning: range is invalid
warning: (LKL) selects TCP_CONG_BBR which has unmet direct dependencies (NET && INET && TCP_CONG_ADVANCED)
warning: (LKL) selects TCP_CONG_BBR which has unmet direct dependencies (NET && INET && TCP_CONG_ADVANCED)
CHK include/config/kernel.release
CHK arch/lkl/include/generated/uapi/asm/syscall_defs.h
CHK include/generated/uapi/linux/version.h
CHK include/generated/utsrelease.h
CHK include/generated/timeconst.h
CHK include/generated/asm-offsets.h
CALL scripts/checksyscalls.sh
CHK include/generated/compile.h
objcopy -j -syscall_defs -O binary --set-section-flags=syscall_defs-alloc vmlinux arch/lkl/include/generated/uapi/asm/syscall_defs.h
sed -i 's/\x0//g' arch/lkl/include/generated/uapi/asm/syscall_defs.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/lib/lkl.o
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/shmbuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/virtio_ids.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/swab.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/ipcbuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/bitsperlong.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/siginfo.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/autoconf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/stddef.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/times.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/virtio_types.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/adv.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/hdrc/ioctl.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/utime.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/in6.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/signal-defs.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/shmbuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/msgbuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/ipcbuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/libc-compat.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/virtio_net.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/swab.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/posix_types.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/futex.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/shm.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/byteorder.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/syscalls.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/socket.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/sen.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/syscall_defs.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/errno.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/virtio_ring.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/ipc.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/ioctl.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/host_ops.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/rtnetlink.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/types.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/config.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/limits.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/fcntl.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/if_addr.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/socket.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/poll.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/errno.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm/stat.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/byteorder/little_endian.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/linux/fs.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/sembuf.h
INSTALL /home/my601/lkl/lkl-source/tools/lkl/include/lkl/asm-generic/posix_types.h

```

```

INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/poll.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/types.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/signal.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/poll.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/unistd.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/stat.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/bpf_common.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/stat.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/msg.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/fcntl.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/resource.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/netlink.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/resource.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/eventpoll.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/socket.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/signal.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/pkt_sched.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/kernel.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/if_ether.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/swab.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/types.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/in.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/statfs.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/sysinfo.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/byteorder/big_endian.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/sockios.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/times.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/socket.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/capability.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/posix_types.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/sockios.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/route.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/ioctl.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/msgbuf.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/if_link.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/irq.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/ipv6.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/statfs.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/senbuf.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/resource.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/unistd.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/if.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/int-ll64.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/bpf.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/ioctl.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/time.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/neighbor.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/nqueue.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/fcntl.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn/siginfo.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/asn-generic/errno-base.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/aio_abi.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/perf_event.h
INSTALL /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/uio.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/in6.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/swab.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/host_ops.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/irq.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/perf_event.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/sockios.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/ioctl.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/senbuf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/sockios.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/socket.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/bpf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/limits.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/stat.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/if_addr.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/autoconf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/pkt_sched.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/msgbuf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/swab.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/futex.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/ipcbuf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/shmbuf.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/sockios.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/poll.h
REPLACE /home/myy601/lkl/lkl-source/tools/lkl/include/lkl/linux/siginfo.h

```



```

REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/siginfo.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/statfs.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/utime.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/poll.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/msg.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/sembuf.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/posix_types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/syinfo.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/aio_abi.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/posix_types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/if.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/bitsperlong.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/kernel.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/shmbuf.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/route.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/fcntl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/if_link.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/neighbor.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/if_ether.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/libc_compat.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/resource.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/fs.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/signal.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/sem.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_ring.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/signal.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/time.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/unistd.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ipv6_route.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/swab.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/stddef.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/fcntl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/signal-defs.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ipc.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_ids.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ioctl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/unistd.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ipv6.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/bpf_common.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/socket.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/in.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/errno-base.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_net.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ioctl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/shm.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/fcntl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/uio.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/statfs.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/rtnetlink.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/resource.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/times.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/socket.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/syscalls.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/posix_types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/little_endian.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_config.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ipcbuf.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_blk.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/stat.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/resource.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/ioctl.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/virtio_types.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/times.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/big_endian.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/syscall_defs.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/hdow_t.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/poll.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/errno.h
REPLACE /home/myy601/kl/kl-source/tools/kl/include/kl/linux/byteorder.h
CC /home/myy601/kl/kl-source/tools/kl/klfuse.o
LD /home/myy601/kl/kl-source/tools/kl/klfuse-in.o
CC /home/myy601/kl/kl-source/tools/kl/lib/fs.o
CC /home/myy601/kl/kl-source/tools/kl/lib/ionen.o
CC /home/myy601/kl/kl-source/tools/kl/lib/net.o
CC /home/myy601/kl/kl-source/tools/kl/lib/tcp_buf.o
CC /home/myy601/kl/kl-source/tools/kl/lib/posix-host.o

```

```

CC /home/myy601/kl/kl-source/tools/kl/lib/posix-host.o
CC /home/myy601/kl/kl-source/tools/kl/lib/utls.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_blk.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio.o
CC /home/myy601/kl/kl-source/tools/kl/lib/dbg.o
CC /home/myy601/kl/kl-source/tools/kl/lib/dbg_handler.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_fd.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_tap.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_raw.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_nacvtap.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_dpdk.o
CC /home/myy601/kl/kl-source/tools/kl/lib/virtio_net_vde.o
LD /home/myy601/kl/kl-source/tools/kl/lib/kl-in.o
AR /home/myy601/kl/kl-source/tools/kl/lib/kl.a
LINK /home/myy601/kl/kl-source/tools/kl/klfuse
CC /home/myy601/kl/kl-source/tools/kl/cptofs.o
LD /home/myy601/kl/kl-source/tools/kl/cptofs-in.o
LINK /home/myy601/kl/kl-source/tools/kl/cptofs
CC /home/myy601/kl/kl-source/tools/kl/fs2tar.o
LD /home/myy601/kl/kl-source/tools/kl/fs2tar-in.o
LINK /home/myy601/kl/kl-source/tools/kl/fs2tar
LINK /home/myy601/kl/kl-source/tools/kl/lib/kl.so
CC /home/myy601/kl/kl-source/tools/kl/lib/hijack/hijack.o
CC /home/myy601/kl/kl-source/tools/kl/lib/hijack/init.o
CC /home/myy601/kl/kl-source/tools/kl/lib/hijack/xlate.o
LD /home/myy601/kl/kl-source/tools/kl/lib/hijack/hijack-in.o
LINK /home/myy601/kl/kl-source/tools/kl/lib/kl-hijack.so
make: Leaving directory '/home/myy601/kl/kl-source/tools/kl'
myy601@myy601lab2:~/kl/kl-sources$ sss

```

4.2 Tests

- `cd tools/lkl` → Πάμε στον κατάλογο `tools/lkl`
- `make test` → Τρέχουμε το test να δούμε ότι κάνει pass

```
myy601@myy601lab2:~/lkl/lkl-source$ cd tools/lkl
myy601@myy601lab2:~/lkl/lkl-source/tools/lkl$ make test
CC      /home/myy601/lkl/lkl-source/tools/lkl/tests/boot.o
LD       /home/myy601/lkl/lkl-source/tools/lkl/tests/boot-in.o
LINK     /home/myy601/lkl/lkl-source/tools/lkl/tests/boot
CC      /home/myy601/lkl/lkl-source/tools/lkl/tests/net-test.o
LD       /home/myy601/lkl/lkl-source/tools/lkl/tests/net-test-in.o
LINK     /home/myy601/lkl/lkl-source/tools/lkl/tests/net-test
make -C tests test
for fs in vfat; do ./boot.sh -t $fs || exit 1; done
102400+0 records in
102400+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 0.400432 s, 262 MB/s
mutex      passed [1]
semaphore  passed [1]
join       passed [joined 140449474832128]
disk_add   passed [3 0]
getpid     passed [1]
syscall_latency passed [avg/min/max lkl: 109/105/327 native: 468/413/27829]
umask      passed [22 777]
creat      passed [0]
close      passed [0]
failopen   passed [-2]
open       passed [0]
write      passed [5]
lseek      passed [0 ]
read       passed [5 test]
fstat      passed [0 100721 5]
mkdir      passed [0]
stat       passed [0 40721]
nanosleep  passed [98597328]
pipe2      passed [Hello world!]
epoll      passed [Hello world!]
mount_fs   passed [proc: 0]
chdir      passed [0]
opendir    passed [4]
getdents64 passed [4 . . . fs bus irq net sys tty kmsg maps misc stat iomem cry]
umount_fs  passed [proc: 0 0 0]
mount_dev  passed [0]
chdir      passed [0]
opendir    passed [5]
getdents64 passed [5 . . . ]
umount_dev passed [0 0 0]
lo_ifup    passed [0]
gettid     passed [14801]
syscall_thread passed [1]
many_syscall_threads passed [1]
myy601@myy601lab2:~/lkl/lkl-source/tools/lkl$
```

- `cd tests` → Πάμε στον κατάλογο `tools/lkl`
- `./boot -t vfat -d /tmp/vfatfile -p -P 0` → Έλεγχος εκκίνησης του συστήματος FAT

```

lkl@lkl-source/tools/lkl$ cd tests
lkl@lkl-source/tools/lkl$ ./boot -t vfat -d /tmp/vfatfile -p -P 0
mutex                passed [1]
semaphore            passed [1]
join                 passed [joined 140227702261504]
disk_add             passed [3 0]
[ 0.000000] Linux version 4.11.0 (lkl@lkl-source/tools/lkl) (gcc version 8.3.0 (Debian 8.3.0-6) ) #9 Mon May 22 15:39:25 EEST 2023
[ 0.000000] bootmem address range: 0x7f894ce18000 - 0x7f894de17000
[ 0.000000] On node 0 totalpages: 4095
[ 0.000000] free_area_init_node: node 0, pgdat 5638d27ad480, node_mem_map 7f894ce21540
[ 0.000000]   Normal zone: 56 pages used for memmap
[ 0.000000]   Normal zone: 0 pages reserved
[ 0.000000]   Normal zone: 4095 pages, LIFO batch:0
[ 0.000000] pcpu-alloc: s0 r0 d32768 u32768 alloc=1*32768
[ 0.000000] pcpu-alloc: [0] 0
[ 0.000000] Built 1 zonelists in Zone order, mobility grouping off. Total pages: 4039
[ 0.000000] Kernel command line: mem=16M loglevel=8 virtio-mmio.device=292@0x1000000:1
[ 0.000000] PID hash table entries: 64 (order: -3, 512 bytes)
[ 0.000000] Dentry cache hash table entries: 2048 (order: 2, 16384 bytes)
[ 0.000000] Inode-cache hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.000000] Memory available: 16028k/0k RAM
[ 0.000000] SLUB: HWalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
[ 0.000000] NR_IRQS:4096
[ 0.000000] lkl: Irqs initialized
[ 0.000000] clocksource: lkl: mask: 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000000] lkl: time and timers initialized (irq2)
[ 0.000000] pid_max: default: 4096 minimum: 301
[ 0.000026] Mount-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000027] Mountpoint-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.006112] console [lkl console0] enabled
[ 0.006129] clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 19112604462750000 ns
[ 0.006242] NET: Registered protocol family 16
[ 0.009115] clocksource: Switched to clocksource lkl
[ 0.009221] NET: Registered protocol family 2
[ 0.009337] TCP established hash table entries: 512 (order: 0, 4096 bytes)
[ 0.009356] TCP bind hash table entries: 512 (order: 0, 4096 bytes)
[ 0.009360] TCP: Hash tables configured (established 512 bind 512)
[ 0.009391] UDP hash table entries: 128 (order: 0, 4096 bytes)
[ 0.009394] UDP-Lite hash table entries: 128 (order: 0, 4096 bytes)
[ 0.009472] virtio-mmio: Registering device virtio-mmio.0 at 0x1000000-0x1000123, IRQ 1.
[ 0.009807] workingset: timestamp_bits=62 max_order=12 bucket_order=0
[ 0.017517] io scheduler noop registered
[ 0.017552] io scheduler deadline registered
[ 0.017617] io scheduler cfq registered (default)
[ 0.017635] io scheduler mq-deadline registered
[ 0.017645] virtio-mmio virtio-mmio.0: Failed to enable 64-bit or 32-bit DMA. Trying to continue, but this might not work.
[ 0.022826] vda:
[ 0.023091] NET: Registered protocol family 10
[ 0.023732] Segment Routing with IPv6
[ 0.023780] sit: IPv6, IPv4 and MPLS over IPv4 tunneling driver
[ 0.024146] Warning: unable to open an initial console.
[ 0.024179] This architecture does not have kernel memory protection.
[ 0.024209] entering generic_file_read_iter in filemap.c
getpid                passed [1]
syscall_latency       passed [avg/min/max lkl: 153/98/23901 native: 398/388/6873]
umask                 passed [22 777]
creat                 passed [0]
close                 passed [0]

```



```

failopen      passed [-2]
open          passed [0]
[ 0.024213] exiting generic_file_read_iter in filemap.c
[ 0.025113] entering generic_file_write_iter in filemap.c
write         passed [5]
[ 0.025120] exiting generic_file_write_iter in filemap.c
[ 0.025125] entering generic_file_llseek in read_write.c
lseek        passed [0]
[ 0.025127] exiting generic_file_llseek in read_write.c
[ 0.025131] entering generic_file_read_iter in filemap.c
read         passed [5 test]
fstat        passed [0 100721 5]
mkdir        passed [0]
stat         passed [0 40721]
nanosleep    passed [87846044]
pipe2        passed [Hello world!]
epoll        passed [Hello world!]
mount_fs     passed [proc: 0]
chdir        passed [0]
opendir      passed [4]
getdents64   passed [4 . . . fs bus irq net sys tty kmsg maps misc stat iomem cry]
umount_fs    passed [proc: 0 0 0]
[ 0.025133] exiting generic_file_read_iter in filemap.c
[ 0.113588] cluster size = 2048
[ 0.113632] entering generic_file_write_iter in filemap.c
[ 0.113638] exiting generic_file_write_iter in filemap.c
[ 0.113641] cluster bits = 11
[ 0.113645] entering generic_file_write_iter in filemap.c
[ 0.113646] exiting generic_file_write_iter in filemap.c
[ 0.113648] fats =
[ 0.113652] entering generic_file_write_iter in filemap.c
[ 0.113653] exiting generic_file_write_iter in filemap.c
[ 0.113655] fat start = 4
[ 0.113659] entering generic_file_write_iter in filemap.c
[ 0.113660] exiting generic_file_write_iter in filemap.c
[ 0.113663] fat length = 200
[ 0.113666] entering generic_file_write_iter in filemap.c
[ 0.113667] exiting generic_file_write_iter in filemap.c
[ 0.113670] free clus valid = 0
[ 0.113673] entering generic_file_write_iter in filemap.c
[ 0.113679] exiting generic_file_write_iter in filemap.c
[ 0.113702] fat bits =
[ 0.113706] entering generic_file_write_iter in filemap.c
[ 0.113710] exiting generic_file_write_iter in filemap.c
[ 0.113712] root cluster = 0
[ 0.113736] entering generic_file_write_iter in filemap.c
[ 0.113737] exiting generic_file_write_iter in filemap.c
[ 0.113740] free clusters = -1
[ 0.113744] entering generic_file_write_iter in filemap.c
[ 0.113744] exiting generic_file_write_iter in filemap.c
[ 0.113747] prev free = 2
[ 0.113751] entering generic_file_write_iter in filemap.c
[ 0.113752] exiting generic_file_write_iter in filemap.c
[ 0.113758] entering fat_alloc_inode in inode.c
[ 0.114026] exiting fat_alloc_inode in inode.c
[ 0.114052] entering fat_alloc_inode in inode.c
[ 0.114056] exiting fat_alloc_inode in inode.c
[ 0.114058] entering fat_alloc_inode in inode.c
[ 0.114060] exiting fat_alloc_inode in inode.c

```

```

[ 0.114060] exiting fat_alloc_inode in inode.c
[ 0.114188] fat16 state = 1
[ 0.114212] entering generic_file_write_iter in filemap.c
mount_dev     passed [0]
chdir         passed [0]
opendir       passed [5]
getdents64    passed [5 . . . lklfuse.c]
[ 0.114214] exiting generic_file_write_iter in filemap.c
[ 0.114997] entering fat_evict_inode in inode.c
[ 0.115025] exiting fat_evict_inode in inode.c
[ 0.115029] entering fat_destroy_inode in inode.c
[ 0.115032] exiting fat_destroy_inode in inode.c
[ 0.115034] entering fat_put_super in inode.c
[ 0.115038] fat16 state = 0
[ 0.115043] entering generic_file_write_iter in filemap.c
[ 0.115045] exiting generic_file_write_iter in filemap.c
[ 0.115067] entering fat_evict_inode in inode.c
[ 0.115070] exiting fat_evict_inode in inode.c
[ 0.115073] entering fat_destroy_inode in inode.c
[ 0.115075] exiting fat_destroy_inode in inode.c
[ 0.115096] entering fat_evict_inode in inode.c
[ 0.115099] exiting fat_evict_inode in inode.c
[ 0.115102] entering fat_destroy_inode in inode.c
[ 0.115104] exiting fat_destroy_inode in inode.c

```

4.3 Εφαρμογές

- `./cptofs -i /tmp/vfatfile -p -t vfat lklfuse.c/` → Δοκιμάζουμε το copy προς το file system

```
my@mylaptop:~/lkl-source/tools/rats$ ./cptofs -i /tmp/vfatfile -p -t vfat lklfuse.c /
[ 0.000000] Linux version 4.11.0 (my5010my501lab2) (gcc version 8.3.0 (Debian 8.3.0-6) ) #8 Sat May 13 22:31:19 EEST 2023
[ 0.000000] bootmem address range: 0x7f1505c00000 - 0x7f150bffff000
[ 0.000000] Built 1 zonelists in Zone order, mobility grouping on. Total pages: 25749
[ 0.000000] Kernel command line: mem=100M virtio mmio.device=292@0x1000000:1
[ 0.000000] PID hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000000] Dentry cache hash table entries: 16384 (order: 5, 131072 bytes)
[ 0.000000] Inode-cache hash table entries: 8192 (order: 4, 65536 bytes)
[ 0.000000] Memory available: 100752k/0k RAM
[ 0.000000] SLUB: HMAAlign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
[ 0.000000] NR_IRQS:4096
[ 0.000000] lkl: irq: initialized
[ 0.000000] clocksource: lkl: mask: 0xffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000001] lkl: time and timers initialized (irq2)
[ 0.000020] pid_max: default: 4096 minimum: 301
[ 0.000037] Mount-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000038] Mountpoint-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.004738] console [lkl console0] enabled
[ 0.004745] clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 1911260446750000 ns
[ 0.004832] NET: Registered protocol family 16
[ 0.007474] clocksource: Switched to clocksource lkl
[ 0.007542] NET: Registered protocol family 2
[ 0.007686] TCP established hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.007711] TCP bind hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.007716] TCP: Hash tables configured (established 1024 bind 1024)
[ 0.007989] UDP hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008013] UDP-Lite hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008101] virtio-mmio: Registering device virtio-mmio.0 at 0x1000000-0x1000123, IRQ 1.
[ 0.008654] workqueue: timestamp Bits=62 max_order=15 bucket_order=0
[ 0.015853] io scheduler noop registered
[ 0.015887] io scheduler deadline registered
[ 0.015937] io scheduler cfq registered (default)
[ 0.015953] io scheduler mq-deadline registered
[ 0.015964] virtio-mmio virtio-mmio.0: Failed to enable 64-bit or 32-bit DMA. Trying to continue, but this might not work.
[ 0.019367] vda:
[ 0.019588] NET: Registered protocol family 10
[ 0.020052] Segment Routing with IPv6
[ 0.020084] sit: IPv6, IPv4 and MPLS over IPv4 tunneling driver
[ 0.020375] Warning: unable to open an initial console.
[ 0.020408] This architecture does not have kernel memory protection.
[ 0.020416] entering generic file read_iter in filemap.c
[ 0.020419] exiting generic file_read_iter in filemap.c
[ 0.021151] cluster size = 2048
[ 0.021175] entering generic file write_iter in filemap.c
[ 0.021181] exiting generic file_write_iter in filemap.c
[ 0.021243] cluster bits = 11
[ 0.021247] entering generic file write_iter in filemap.c
[ 0.021248] exiting generic file_write_iter in filemap.c
[ 0.021250] fats =
[ 0.021253] entering generic file write_iter in filemap.c
[ 0.021254] exiting generic file_write_iter in filemap.c
[ 0.021256] fat start = 4
[ 0.021258] entering generic file write_iter in filemap.c
[ 0.021259] exiting generic file_write_iter in filemap.c
[ 0.021264] fat length = 200
[ 0.021281] entering generic file write_iter in filemap.c
[ 0.021282] exiting generic file_write_iter in filemap.c
[ 0.021284] free clus valid = 0
[ 0.021287] entering generic file write_iter in filemap.c
[ 0.021288] exiting generic file_write_iter in filemap.c
[ 0.021290] fat bits =
[ 0.021292] entering generic file write_iter in filemap.c
[ 0.021294] exiting generic file_write_iter in filemap.c
[ 0.021300] root_cluster = 0
[ 0.021317] entering generic file write_iter in filemap.c
[ 0.021318] exiting generic file_write_iter in filemap.c
[ 0.021320] free clusters = -1
[ 0.021323] entering generic file write_iter in filemap.c
[ 0.021323] exiting generic file_write_iter in filemap.c
[ 0.021325] prev free = 2
[ 0.021328] entering generic file write_iter in filemap.c
[ 0.021328] exiting generic file_write_iter in filemap.c
[ 0.021333] entering fat_alloc_inode in inode.c
[ 0.021339] exiting fat_alloc_inode in inode.c
[ 0.021341] entering fat_alloc_inode in inode.c
[ 0.021342] exiting fat_alloc_inode in inode.c
[ 0.021344] entering fat_alloc_inode in inode.c
[ 0.021345] exiting fat_alloc_inode in inode.c
[ 0.021778] fat16 state = 1
[ 0.021783] entering generic file write_iter in filemap.c
[ 0.021784] exiting generic file_write_iter in filemap.c
[ 0.021787] entering generic file write_iter in filemap.c
[ 0.021787] exiting generic file_write_iter in filemap.c
```

```

0.021905] exiting vfat_lookup in namei_vfat.c
0.021909] entering vfat_create in namei_vfat.c
0.021921] id = 1
0.021924] entering generic_file_write_iter in filemap.c
0.021925] exiting generic_file_write_iter in filemap.c
0.021927] attr = 15
0.021930] entering generic_file_write_iter in filemap.c
0.021931] exiting generic_file_write_iter in filemap.c
0.021933] reserved = 0
0.021936] entering generic_file_write_iter in filemap.c
0.021936] exiting generic_file_write_iter in filemap.c
0.021938] alias checksum = 209
0.021941] entering generic_file_write_iter in filemap.c
0.021942] exiting generic_file_write_iter in filemap.c
0.021944] start = 0
0.021962] entering generic_file_write_iter in filemap.c
0.021963] exiting generic_file_write_iter in filemap.c
0.021965] name = ls0000
0.021968] entering generic_file_write_iter in filemap.c
0.021968] exiting generic_file_write_iter in filemap.c
0.021971] name = LKLFUSE C
0.021974] entering generic_file_write_iter in filemap.c
0.021975] exiting generic_file_write_iter in filemap.c
0.021990] attr = 32
0.021993] entering generic_file_write_iter in filemap.c
0.021994] exiting generic_file_write_iter in filemap.c
0.021996] lcase = 0
0.022012] entering generic_file_write_iter in filemap.c
0.022013] exiting generic_file_write_iter in filemap.c
0.022018] entering fat_alloc_inode in inode.c
0.022032] exiting fat_alloc_inode in inode.c
0.022035] exiting vfat_create in namei_vfat.c
0.022042] entering generic_file_write_iter in filemap.c
0.022058] entering fat_write_begin in inode.c
0.022064] entering fat_ent_blocknr in fatent.c
0.022065] entering fat_ent_blocknr in fatent.c
0.022067] entering fat_ent_bread in fatent.c
0.022080] entering fat16_ent_set_ptr in fatent.c
0.022095] exiting fat16_ent_set_ptr in fatent.c
0.022097] exiting fat_ent_bread in fatent.c
0.022099] entering fat16_ent_get in fatent.c
0.022101] exiting fat16_ent_get in fatent.c
0.022103] entering fat16_ent_put in fatent.c
0.022120] exiting fat16_ent_put in fatent.c
0.022130] entering fat_ent_blocknr in fatent.c
0.022144] entering fat_ent_blocknr in fatent.c
0.022147] entering fat_ent_bread in fatent.c
0.022149] entering fat16_ent_set_ptr in fatent.c
0.022150] exiting fat16_ent_set_ptr in fatent.c
0.022152] exiting fat_ent_bread in fatent.c
0.022185] entering fat16_ent_get in fatent.c
0.022188] exiting fat16_ent_get in fatent.c
0.022189] entering fat16_ent_put in fatent.c
0.022191] exiting fat16_ent_put in fatent.c
0.022194] entering fat_ent_blocknr in fatent.c
0.022225] entering fat_ent_blocknr in fatent.c
0.022227] entering fat_ent_bread in fatent.c
0.022230] entering fat16_ent_set_ptr in fatent.c
0.022244] exiting fat16_ent_set_ptr in fatent.c
0.022246] exiting fat_ent_bread in fatent.c
0.022248] entering fat16_ent_get in fatent.c
0.022249] exiting fat16_ent_get in fatent.c
0.022277] entering fat_ent_blocknr in fatent.c
0.022279] entering fat_ent_blocknr in fatent.c
0.022280] entering fat_ent_bread in fatent.c
0.022282] entering fat16_ent_set_ptr in fatent.c
0.022296] exiting fat16_ent_set_ptr in fatent.c
0.022298] exiting fat_ent_bread in fatent.c
0.022300] entering fat16_ent_get in fatent.c
0.022301] exiting fat16_ent_get in fatent.c
0.022304] entering fat16_ent_put in fatent.c
0.022317] exiting fat16_ent_put in fatent.c
0.022320] entering fat_ent_blocknr in fatent.c
0.022322] entering fat_ent_blocknr in fatent.c
0.022324] entering fat_ent_bread in fatent.c
0.022338] entering fat16_ent_set_ptr in fatent.c
0.022340] exiting fat16_ent_set_ptr in fatent.c
0.022342] exiting fat_ent_bread in fatent.c
0.022343] entering fat16_ent_get in fatent.c
0.022345] exiting fat16_ent_get in fatent.c
0.022361] exiting fat_write_begin in inode.c
0.022366] entering fat_write_end in inode.c

```





```

0.022856| entering fat_ent_blocknr in fatent.c
0.022870| entering fat_ent_bread in fatent.c
0.022872| entering fat16_ent_set_ptr in fatent.c
0.022874| exiting fat16_ent_set_ptr in fatent.c
0.022875| exiting fat_ent_bread in fatent.c
0.022889| entering fat16_ent_get in fatent.c
0.022891| exiting fat16_ent_get in fatent.c
0.022893| entering fat16_ent_put in fatent.c
0.022894| exiting fat16_ent_put in fatent.c
0.022897| entering fat_ent_blocknr in fatent.c
0.022911| entering fat_ent_blocknr in fatent.c
0.022913| entering fat_ent_bread in fatent.c
0.022915| entering fat16_ent_set_ptr in fatent.c
0.022917| exiting fat16_ent_set_ptr in fatent.c
0.022931| exiting fat_ent_bread in fatent.c
0.022933| entering fat16_ent_get in fatent.c
0.022935| exiting fat16_ent_get in fatent.c
0.022936| entering fat_ent_blocknr in fatent.c
0.022938| entering fat_ent_blocknr in fatent.c
0.022952| entering fat_ent_bread in fatent.c
0.022954| entering fat16_ent_set_ptr in fatent.c
0.022956| exiting fat16_ent_set_ptr in fatent.c
0.022957| exiting fat_ent_bread in fatent.c
0.022972| entering fat16_ent_get in fatent.c
0.022974| exiting fat16_ent_get in fatent.c
0.022976| entering fat16_ent_put in fatent.c
0.022977| exiting fat16_ent_put in fatent.c
0.022980| entering fat_ent_blocknr in fatent.c
0.022993| entering fat_ent_blocknr in fatent.c
0.022996| entering fat_ent_bread in fatent.c
0.022997| entering fat16_ent_set_ptr in fatent.c
0.022999| exiting fat16_ent_set_ptr in fatent.c
0.023013| exiting fat_ent_bread in fatent.c
0.023015| entering fat16_ent_get in fatent.c
0.023017| exiting fat16_ent_get in fatent.c
0.023019| entering fat_ent_blocknr in fatent.c
0.023036| entering fat_ent_blocknr in fatent.c
0.023038| entering fat_ent_bread in fatent.c
0.023040| entering fat16_ent_set_ptr in fatent.c
0.023054| exiting fat16_ent_set_ptr in fatent.c
0.023056| exiting fat_ent_bread in fatent.c
0.023058| entering fat16_ent_get in fatent.c
0.023059| exiting fat16_ent_get in fatent.c
0.023061| entering fat16_ent_put in fatent.c
0.023075| exiting fat16_ent_put in fatent.c
0.023078| entering fat_ent_blocknr in fatent.c
0.023080| entering fat_ent_blocknr in fatent.c
0.023082| entering fat_ent_bread in fatent.c
0.023095| entering fat16_ent_set_ptr in fatent.c
0.023097| exiting fat16_ent_set_ptr in fatent.c
0.023099| exiting fat_ent_bread in fatent.c
0.023101| entering fat16_ent_get in fatent.c
0.023103| exiting fat16_ent_get in fatent.c
0.023116| entering fat_ent_blocknr in fatent.c
0.023119| entering fat_ent_blocknr in fatent.c
0.023120| entering fat_ent_bread in fatent.c
0.023122| entering fat16_ent_set_ptr in fatent.c
0.023136| exiting fat16_ent_set_ptr in fatent.c
0.023138| exiting fat_ent_bread in fatent.c
0.023140| entering fat16_ent_get in fatent.c
0.023141| exiting fat16_ent_get in fatent.c
0.023144| entering fat16_ent_put in fatent.c
0.023191| exiting fat16_ent_put in fatent.c
0.023194| entering fat_ent_blocknr in fatent.c
0.023196| entering fat_ent_blocknr in fatent.c
0.023198| entering fat_ent_bread in fatent.c
0.023212| entering fat16_ent_set_ptr in fatent.c
0.023214| exiting fat16_ent_set_ptr in fatent.c
0.023216| exiting fat_ent_bread in fatent.c
0.023218| entering fat16_ent_get in fatent.c
0.023220| exiting fat16_ent_get in fatent.c
0.023247| exiting fat_write_begin in inode.c
0.023252| entering fat_write_end in inode.c
0.023267| exiting fat_write_end in inode.c
0.023269| exiting generic_file_write_iter in filenap.c
0.023274| entering generic_file_write_iter in filenap.c
0.023288| entering fat_write_begin in inode.c
0.023292| entering fat_ent_blocknr in fatent.c
0.023294| entering fat_ent_blocknr in fatent.c
0.023307| entering fat_ent_bread in fatent.c
0.023310| entering fat16_ent_set_ptr in fatent.c
0.023311| exiting fat16_ent_set_ptr in fatent.c

```



```

[ 0.023327] exiting fat16_ent_bread in fatent.c
[ 0.023327] entering fat16_ent_get in fatent.c
[ 0.023329] exiting fat16_ent_get in fatent.c
[ 0.023331] entering fat16_ent_put in fatent.c
[ 0.023332] exiting fat16_ent_put in fatent.c
[ 0.023335] entering fat_ent_blocknr in fatent.c
[ 0.023340] entering fat_ent_blocknr in fatent.c
[ 0.023351] entering fat_ent_bread in fatent.c
[ 0.023353] entering fat16_ent_set_ptr in fatent.c
[ 0.023354] exiting fat16_ent_set_ptr in fatent.c
[ 0.023356] exiting fat_ent_bread in fatent.c
[ 0.023370] entering fat16_ent_get in fatent.c
[ 0.023372] exiting fat16_ent_get in fatent.c
[ 0.023374] entering fat_ent_blocknr in fatent.c
[ 0.023376] entering fat_ent_blocknr in fatent.c
[ 0.023378] entering fat_ent_bread in fatent.c
[ 0.023391] entering fat16_ent_set_ptr in fatent.c
[ 0.023393] exiting fat16_ent_set_ptr in fatent.c
[ 0.023395] exiting fat_ent_bread in fatent.c
[ 0.023396] entering fat16_ent_get in fatent.c
[ 0.023398] exiting fat16_ent_get in fatent.c
[ 0.023412] entering fat16_ent_put in fatent.c
[ 0.023414] exiting fat16_ent_put in fatent.c
[ 0.023417] entering fat_ent_blocknr in fatent.c
[ 0.023419] entering fat_ent_blocknr in fatent.c
[ 0.023432] entering fat_ent_bread in fatent.c
[ 0.023435] entering fat16_ent_set_ptr in fatent.c
[ 0.023439] exiting fat16_ent_set_ptr in fatent.c
[ 0.023452] exiting fat_ent_bread in fatent.c
[ 0.023455] entering fat16_ent_get in fatent.c
[ 0.023456] exiting fat16_ent_get in fatent.c
[ 0.023459] exiting fat_write_begin in inode.c
[ 0.023474] entering fat_write_end in inode.c
[ 0.023477] exiting fat_write_end in inode.c
[ 0.023478] exiting generic_file_write_iter in filemap.c
[ 0.023499] entering fat_file_release in file.c
[ 0.023514] exiting fat_file_release in file.c
[ 0.023887] entering fat_write_inode in inode.c
[ 0.023894] exiting fat_write_inode in inode.c
[ 0.023897] entering fat_writepages in inode.c
[ 0.023909] entering fat_write_inode in inode.c
[ 0.023911] i_node = 33261
[ 0.023915] i_uid = 0
[ 0.023916] i_gid = 0
[ 0.023919] raw entry size = 13958
[ 0.023920] raw entry attr = 32
[ 0.023922] raw entry time = 0
[ 0.023924] raw entry date = 33
[ 0.023925] raw entry ctime = 0
[ 0.023927] raw entry cdate = 33
[ 0.023928] raw entry adate = 33
[ 0.023930] exiting fat_write_inode in inode.c
[ 0.023998] entering fat_evict_inode in inode.c
[ 0.024001] exiting fat_evict_inode in inode.c
[ 0.024003] entering fat_destroy_inode in inode.c
[ 0.024005] exiting fat_destroy_inode in inode.c
[ 0.024007] entering fat_evict_inode in inode.c
[ 0.024014] exiting fat_evict_inode in inode.c
[ 0.024016] entering fat_destroy_inode in inode.c
[ 0.024017] exiting fat_destroy_inode in inode.c
[ 0.024019] entering fat_put_super in inode.c
[ 0.024022] fat16 state = 0
[ 0.024025] entering generic_file_write_iter in filemap.c
[ 0.024027] exiting generic_file_write_iter in filemap.c
[ 0.024035] entering fat_evict_inode in inode.c
[ 0.024037] exiting fat_evict_inode in inode.c
[ 0.024039] entering fat_destroy_inode in inode.c
[ 0.024040] exiting fat_destroy_inode in inode.c
[ 0.024042] entering fat_evict_inode in inode.c
[ 0.024044] exiting fat_evict_inode in inode.c
[ 0.024045] entering fat_destroy_inode in inode.c
[ 0.024047] exiting fat_destroy_inode in inode.c
[ 0.024049] exiting fat_put_super in inode.c
[ 0.027447] reboot: Restarting system

```


- `./cpfromfs -i /tmp/vfatfile -p -t vfat lklfuse.c/` → Δοκιμάζουμε το copy από το file system

```

my6010myy601lab2:~/lkl/lkl-source/tools/lkl$ ./cpfromfs -i /tmp/vfatfile -p -t vfat lklfuse.c /
[ 0.000000] Linux version 4.11.0 (my6010myy601lab2) (gcc version 8.3.0 (Debian 8.3.0-6) ) #8 Sat May 13 22:31:19 EEST 2023
[ 0.000000] bootmem address range: 0x7f30fcd00000 - 0x7f3103ffff000
[ 0.000000] Built 1 zonelists in Zone order, mobility grouping on. Total pages: 25249
[ 0.000000] Kernel command line: mem=100M virtio mmio.device=29200x1000000:1
[ 0.000000] PID hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000000] Dentry cache hash table entries: 16384 (order: 5, 131072 bytes)
[ 0.000000] Inode-cache hash table entries: 8192 (order: 4, 65536 bytes)
[ 0.000000] Memory available: 100752k/0k RAM
[ 0.000000] SLUB: Hwalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
[ 0.000000] NR_IRQS:4096
[ 0.000000] lkl: irq: initialized
[ 0.000000] clocksource: lkl: mask: 0xffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000000] lkl: time and timers initialized (irq2)
[ 0.000000] pid max: default: 4096 minimum: 301
[ 0.000025] Mount-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000026] Mountpoint-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.004645] console [lkl_console0] enabled
[ 0.004662] clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 19112604462750000 ns
[ 0.004831] NET: Registered protocol family 16
[ 0.007372] clocksource: Switched to clocksource lkl
[ 0.007520] NET: Registered protocol family 2
[ 0.007674] TCP established hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.007698] TCP bind hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.007703] TCP: Hash tables configured (established 1024 bind 1024)
[ 0.007980] UDP hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008005] UDP-Lite hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008088] virtio-mmio: Registering device virtio-mmio.0 at 0x10000000-0x10000123, IRQ 1.
[ 0.008522] workingset: timestamp_bits=62 max_order=15 bucket_order=0
[ 0.015739] io scheduler noop registered
[ 0.015759] io scheduler deadline registered
[ 0.015827] io scheduler cfq registered (default)
[ 0.015843] io scheduler mq-deadline registered
[ 0.015853] virtio-mmio virtio-mmio.0: Failed to enable 64-bit or 32-bit DMA. Trying to continue, but this might not work.
[ 0.020062] vda:
[ 0.020375] NET: Registered protocol family 10
[ 0.020882] Segment Routing with IPv6
[ 0.020929] sit: IPv6, IPv4 and MPLS over IPv4 tunneling driver
[ 0.021305] Warning: unable to open an initial console.
[ 0.021353] This architecture does not have kernel memory protection.
[ 0.021361] entering generic file read_iter in filemap.c
[ 0.021372] exiting generic file read_iter in filemap.c
[ 0.021414] cluster size = 3048
[ 0.022455] entering generic file write_iter in filemap.c
[ 0.022460] exiting generic file write_iter in filemap.c
[ 0.022463] cluster bits = 11
[ 0.022466] entering generic file write_iter in filemap.c
[ 0.022467] exiting generic file write_iter in filemap.c
[ 0.022469] fats =
[ 0.022472] entering generic file write_iter in filemap.c
[ 0.022472] exiting generic file write_iter in filemap.c
[ 0.022487] fat start = 4
[ 0.022505] entering generic file write_iter in filemap.c
[ 0.022506] exiting generic file write_iter in filemap.c
[ 0.022509] fat length = 200
[ 0.022512] entering generic file write_iter in filemap.c
[ 0.022512] exiting generic file write_iter in filemap.c
[ 0.022527] free clus valid = 0
[ 0.022531] entering generic file write_iter in filemap.c
[ 0.022532] exiting generic file write_iter in filemap.c
[ 0.022534] fat bits =
[ 0.022536] entering generic file write_iter in filemap.c
[ 0.022538] exiting generic file write_iter in filemap.c
[ 0.022543] root cluster = 0
[ 0.022569] entering generic file write_iter in filemap.c
[ 0.022570] exiting generic file write_iter in filemap.c
[ 0.022573] free clusters = -1
[ 0.022577] entering generic file write_iter in filemap.c
[ 0.022577] exiting generic file write_iter in filemap.c
[ 0.022591] prev free = 2
[ 0.022595] entering generic file write_iter in filemap.c
[ 0.022596] exiting generic file write_iter in filemap.c
[ 0.022613] entering fat_alloc_inode in inode.c
[ 0.022620] exiting fat_alloc_inode in inode.c
[ 0.022636] entering fat_alloc_inode in inode.c
[ 0.022638] exiting fat_alloc_inode in inode.c
[ 0.022640] entering fat_alloc_inode in inode.c
[ 0.022642] exiting fat_alloc_inode in inode.c
[ 0.022977] entering vfat_lookup in namei_vfat.c
[ 0.023015] entering fat_alloc_inode in inode.c
[ 0.023018] exiting fat_alloc_inode in inode.c
[ 0.023021] exiting vfat_lookup in namei_vfat.c
[ 0.023023] exiting vfat_lookup in namei_vfat.c

```

```

[ 0.023022] exiting vfat_lookup in namei_vfat.c
[ 0.023025] entering fat_getattr in file.c
unable to open file //lklfuse.c for writing: Permission denied
error processing entry /mnt/0000fe00/lklfuse.c, aborting
[ 0.023027] exiting fat_getattr in file.c
[ 1.114031] reboot: Restarting system

```

- `./fs2tar/tmp/vfatfile -p -t vfat lklfuse.c/` → Μετατρέπει μια εικόνα του fs αρχείο σε αρχείο tar

```

myy60lgmyy60llab2:~/lkl-source/tools/lkl$ ./fs2tar/tmp/vfatfile -p -t vfat lklfuse.c/+
[ 0.000000] Linux version 4.11.0 (myy60lgmyy60llab2) (gcc version 8.3.0 (Debian 8.3.0-6)) #8 Sat May 13 22:31:19 EEST 2023
[ 0.000000] bootmem address range: 0x7fa9f031f600 - 0x7fa9f0d1e000
[ 0.000000] Built 1 zonelists in Zone order, mobility grouping off. Total pages: 2524
[ 0.000000] Kernel command line: nom-i8m virtio-mmio.device=202px1000000:1
[ 0.000000] PID hash table entries: 64 (order: -3, 512 bytes)
[ 0.000000] Dentry cache hash table entries: 2048 (order: 2, 16384 bytes)
[ 0.000000] Inode-cache hash table entries: 1024 (order: 1, 8192 bytes)
[ 0.000000] Memory available: 9940k/0k RAM
[ 0.000000] SLUB: HWalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
[ 0.000000] NR_IRQS:4096
[ 0.000000] lkl: irq initialized
[ 0.000000] clocksource: lkl: mask: 0xffffffffffffffff max_cycles: 0x1cd42e4dffb, max_idle_ns: 881590591483 ns
[ 0.000000] lkl: time and timers initialized (irq2)
[ 0.000000] pid_max: default: 4096 minimum: 301
[ 0.000028] Mount-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000029] Mountpoint-cache hash table entries: 512 (order: 0, 4096 bytes)
[ 0.000076] console [lkl_console0] enabled
[ 0.000095] clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 1911260462750000 ns
[ 0.000312] NET: Registered protocol family 16
[ 0.007877] clocksource: Switched to clocksource lkl
[ 0.008016] NET: Registered protocol family 2
[ 0.008184] TCP established hash table entries: 512 (order: 0, 4096 bytes)
[ 0.008208] TCP bind hash table entries: 512 (order: 0, 4096 bytes)
[ 0.008213] TCP: Hash tables configured (established 512 bind 512)
[ 0.008272] UDP hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008280] UDP-Lite hash table entries: 128 (order: 0, 4096 bytes)
[ 0.008361] virtio-mmio: Registering device virtio-mmio.0 at 0x10000000-0x1000123, IRQ 1.
[ 0.008992] workingset: timestamp_bits=62 max_order=12 bucket_order=0
[ 0.015202] io scheduler noop registered
[ 0.015237] io scheduler deadline registered
[ 0.015304] io scheduler cfq registered (default)
[ 0.015333] io scheduler mq-deadline registered
[ 0.015344] virtio-mmio virtio-mmio.0: Failed to enable 64-bit or 32-bit DMA. Trying to continue, but this might not work.
[ 0.024154] vda:
[ 0.024401] NET: Registered protocol family 10
[ 0.024406] Segment routing with IPv6
[ 0.024838] sit: IPv6, IPv4 and MPLS over IPv4 tunneling driver
[ 0.025053] Warning: unable to open an initial console.
[ 0.025084] This architecture does not have kernel memory protection.
[ 0.025119] entering generic file read iter in filenap.c
[ 0.025151] exiting generic file_read_iter in filenap.c
[ 0.025856] cluster size = 2048
[ 0.025881] entering generic file write iter in filenap.c
[ 0.025885] exiting generic file_write_iter in filenap.c
[ 0.025888] cluster bits = 11
[ 0.025891] entering generic file write iter in filenap.c
[ 0.025892] exiting generic file_write_iter in filenap.c
[ 0.025894] fats =
[ 0.025897] entering generic file write iter in filenap.c
[ 0.025897] exiting generic file_write_iter in filenap.c
[ 0.025899] fat start = 4
[ 0.025902] entering generic file write iter in filenap.c
[ 0.025903] exiting generic file_write_iter in filenap.c
[ 0.025905] fat length = 200
[ 0.025908] entering generic file write iter in filenap.c
[ 0.025908] exiting generic file_write_iter in filenap.c
[ 0.025910] free clus valid = 0
[ 0.025913] entering generic file write iter in filenap.c
[ 0.025914] exiting generic file_write_iter in filenap.c
[ 0.025916] fat bits =
[ 0.025918] entering generic file write iter in filenap.c
[ 0.025920] exiting generic file_write_iter in filenap.c
[ 0.025925] root cluster = 0
[ 0.025928] entering generic file write iter in filenap.c
[ 0.025929] exiting generic file_write_iter in filenap.c
[ 0.025931] free clusters = -1
[ 0.025934] entering generic file write iter in filenap.c
[ 0.025934] exiting generic file_write_iter in filenap.c
[ 0.025964] prev free = 2
[ 0.025971] entering generic file write iter in filenap.c
[ 0.025972] exiting generic file_write_iter in filenap.c
[ 0.025977] entering fat alloc inode in inode.c
[ 0.025983] exiting fat alloc inode in inode.c
[ 0.025986] entering fat alloc inode in inode.c
[ 0.025988] exiting fat alloc inode in inode.c
[ 0.025990] entering fat alloc inode in inode.c
[ 0.025991] exiting fat alloc inode in inode.c
[ 0.026031] entering vfap lookup in namei vfap.c
[ 0.026055] entering fat alloc inode in inode.c
[ 0.026059] exiting fat alloc inode in inode.c
[ 0.026061] exiting vfap lookup in namei vfap.c
[ 0.026063] exiting vfap lookup in namei vfap.c

```



```

[ 0.026961] exiting vfat_lookup in names.vfat.c
[ 0.026963] exiting vfat_lookup in names.vfat.c
[ 0.026966] entering fat_getattr in file.c
[ 0.026968] exiting fat_getattr in file.c
[ 0.027380] entering generic_file_read_iter in filemap.c
[ 0.027403] exiting fat_readpages in inode.c
[ 0.027408] entering fat_ont_blocknr in fatent.c
[ 0.027409] entering fat_ont_blocknr in fatent.c
[ 0.027411] entering fat_ont_bread in fatent.c
[ 0.028158] entering fat16_ont_set_ptr in fatent.c
[ 0.028164] exiting fat16_ont_set_ptr in fatent.c
[ 0.028166] exiting fat_ont_bread in fatent.c
[ 0.028168] entering fat16_ont_get in fatent.c
[ 0.028170] exiting fat16_ont_get in fatent.c
[ 0.028176] entering fat_ont_blocknr in fatent.c
[ 0.028178] entering fat_ont_blocknr in fatent.c
[ 0.028179] entering fat_ont_bread in fatent.c
[ 0.028181] entering fat16_ont_set_ptr in fatent.c
[ 0.028183] exiting fat16_ont_set_ptr in fatent.c
[ 0.028184] exiting fat_ont_bread in fatent.c
[ 0.028186] entering fat16_ont_get in fatent.c
[ 0.028187] exiting fat16_ont_get in fatent.c
[ 0.028207] entering fat_ont_blocknr in fatent.c
[ 0.028209] entering fat_ont_blocknr in fatent.c
[ 0.028211] entering fat_ont_bread in fatent.c
[ 0.028212] entering fat16_ont_set_ptr in fatent.c
[ 0.028214] exiting fat16_ont_set_ptr in fatent.c
[ 0.028216] exiting fat_ont_bread in fatent.c
[ 0.028217] entering fat16_ont_get in fatent.c
[ 0.028232] exiting fat16_ont_get in fatent.c
[ 0.028234] entering fat_ont_blocknr in fatent.c
[ 0.028236] entering fat_ont_blocknr in fatent.c
[ 0.028237] entering fat_ont_bread in fatent.c
[ 0.028239] entering fat16_ont_set_ptr in fatent.c
[ 0.028254] exiting fat16_ont_set_ptr in fatent.c
[ 0.028256] exiting fat_ont_bread in fatent.c
[ 0.028257] entering fat16_ont_get in fatent.c
[ 0.028259] exiting fat16_ont_get in fatent.c
[ 0.028260] entering fat_ont_blocknr in fatent.c
[ 0.028262] entering fat_ont_blocknr in fatent.c
[ 0.028263] entering fat_ont_bread in fatent.c
[ 0.028266] entering fat16_ont_set_ptr in fatent.c
[ 0.028267] exiting fat16_ont_set_ptr in fatent.c
[ 0.028269] exiting fat_ont_bread in fatent.c
[ 0.028270] entering fat16_ont_get in fatent.c
[ 0.028272] exiting fat16_ont_get in fatent.c
[ 0.028274] entering fat_ont_blocknr in fatent.c
[ 0.028275] entering fat_ont_blocknr in fatent.c
[ 0.028277] entering fat_ont_bread in fatent.c
[ 0.028278] entering fat16_ont_set_ptr in fatent.c
[ 0.028281] exiting fat16_ont_set_ptr in fatent.c
[ 0.028282] exiting fat_ont_bread in fatent.c
[ 0.028284] entering fat16_ont_get in fatent.c
[ 0.028285] exiting fat16_ont_get in fatent.c
error writing file /k1fuse.c to archive: INTERNAL ERROR: Function 'archive_write_header' invoked with archive structure in state 'new', should be in state 'open'
[ 0.028318] exiting generic_file_read_iter in filemap.c
[ 0.028574] entering fat_file_release in file.c
error processing entry /k1fuse.c, aborting
[ 0.028607] exiting fat_file_release in file.c
[ 1.115041] reboot: Restarting system

```