
Subject: [PATCH 2/3] extent compilation fixes
Posted by [Dmitry Monakhov](#) on Fri, 22 Jun 2007 12:17:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Fix compilation with EXT_DEBUG, also fix leXX_to_cpu conversions.

Signed-off-by: Dmitry Monakhov <dmonakhov@openvz.org>

```
---
fs/ext4/extents.c | 18 ++++++++-----
1 files changed, 10 insertions(+), 8 deletions(-)

diff --git a/fs/ext4/extents.c b/fs/ext4/extents.c
index 6f72dcb..12fe3d7 100644
--- a/fs/ext4/extents.c
+++ b/fs/ext4/extents.c
@@ -382,13 +382,14 @@ ext4_ext_binsearch_idx(struct inode *inode, struct ext4_ext_path
 *path, int bloc
     r = m - 1;
     else
         l = m + 1;
-    ext_debug("%p(%u):%p(%u):%p(%u) ", l, l->ei_block,
-    m, m->ei_block, r, r->ei_block);
+    ext_debug("%p(%u):%p(%u):%p(%u) ", l, le32_to_cpu(l->ei_block),
+    m, le32_to_cpu(m->ei_block),
+    r, le32_to_cpu(r->ei_block));
 }

     path->p_idx = l - 1;
     ext_debug(" -> %d->%lld ", le32_to_cpu(path->p_idx->ei_block),
-    idx_block(path->p_idx));
+    idx_pblock(path->p_idx));

#ifdef CHECK_BINSEARCH
{
@@ -447,8 +448,9 @@ ext4_ext_binsearch(struct inode *inode, struct ext4_ext_path *path, int
block)
    r = m - 1;
    else
        l = m + 1;
-    ext_debug("%p(%u):%p(%u):%p(%u) ", l, l->ee_block,
-    m, m->ee_block, r, r->ee_block);
+    ext_debug("%p(%u):%p(%u):%p(%u) ", l, le32_to_cpu(l->ee_block),
+    m, le32_to_cpu(m->ee_block),
+    r, le32_to_cpu(r->ee_block));
}

    path->p_ext = l - 1;
@@ -580,7 +582,7 @@ static int ext4_ext_insert_index(handle_t *handle, struct inode *inode,
```

```

if (curp->p_idx != EXT_LAST_INDEX(curp->p_hdr)) {
    len = (len - 1) * sizeof(struct ext4_extent_idx);
    len = len < 0 ? 0 : len;
-   ext_debug("insert new index %d after: %d. "
+   ext_debug("insert new index %d after: %llu. "
        "move %d from 0x%p to 0x%p\n",
        logical, ptr, len,
        (curp->p_idx + 1), (curp->p_idx + 2));
@@ -591,7 +593,7 @@ static int ext4_ext_insert_index(handle_t *handle, struct inode *inode,
/* insert before */
    len = len * sizeof(struct ext4_extent_idx);
    len = len < 0 ? 0 : len;
-   ext_debug("insert new index %d before: %d. "
+   ext_debug("insert new index %d before: %llu. "
        "move %d from 0x%p to 0x%p\n",
        logical, ptr, len,
        curp->p_idx, (curp->p_idx + 1));
@@ -791,7 +793,7 @@ static int ext4_ext_split(handle_t *handle, struct inode *inode,
BUG_ON(EXT_MAX_INDEX(path[i].p_hdr) !=
EXT_LAST_INDEX(path[i].p_hdr));
while (path[i].p_idx <= EXT_MAX_INDEX(path[i].p_hdr)) {
-   ext_debug("%d: move %d:%d in new index %llu\n", i,
+   ext_debug("%d: move %d:%llu in new index %llu\n", i,
        le32_to_cpu(path[i].p_idx->ei_block),
        idx_pblock(path[i].p_idx),
        newblock);

```

--

1.5.2
