Mount-v2 CRIU migration engine: status update

Pavel Tikhomirov
Software developer at Virtuozzo

Sep 2021
Agenda

- Recall mount propagation problems
- Sharing groups vs mount tree order inversion
- Kernel patch status/changes
- To do
Recall mount propagation problems
Recall mount propagation problems

- CRIU knows nothing about history
- Propagation creates vast amount of mounts
- Mount tree re-parenting
- “Mount trap”
- “Non-uniform” propagation
- “Cross-namespace” sharing groups

More detail [1].
Sharing groups vs mount tree order inversion
Sharing groups vs mount tree order inversion

- Imagine mount tree:

<table>
<thead>
<tr>
<th>mntid</th>
<th>parent</th>
<th>mountpoint</th>
<th>(shared_id, master_id)</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>1</td>
<td>/tmp</td>
<td>(0,3)</td>
</tr>
<tr>
<td>102</td>
<td>101</td>
<td>/tmp/sub</td>
<td>(3,2)</td>
</tr>
<tr>
<td>103</td>
<td>102</td>
<td>/tmp/sub/sub</td>
<td>(2,1)</td>
</tr>
<tr>
<td>104</td>
<td>103</td>
<td>/tmp/sub/sub/sub</td>
<td>(1,0)</td>
</tr>
</tbody>
</table>

- Order in mount tree is opposite to order in sharing groups tree
- See test.sh [5]
Sharing groups vs mount tree order inversion

• Mount order restrictions
  ○ At least one mount of each (1,0), (2,1), (3,2) should be mounted before first (0,3)
  ○ Children mounts created after parent mounts
• Without simplifying propagation restore we would need growing number of helper mounts and growing number of umounts to restore such an inverse order chain
Kernel patch status
Kernel patch status

- Original patch "mnt: allow to add a mount into an existing group" [2]
- Current patch (v5) "move_mount: allow to add a mount into an existing group" [3]
- Got to linux master recently, targeting v5.15 [4]
- Thanks to Andrei Vagin and Christian Brauner for a great help with it!!!
Kernel patch status: Interface

```c
syscall(SYS_move_mount, from_dirfd, from_pathname,
       to_dirfd, to_pathname, flags | MOVE_MOUNT_SET_GROUP)
```
Kernel patch status

• Changes from the origin:
  ○ Moved code from sys_mount to sys_move_mount
    ■ Don’t change old mount api (sys_mount)
    ■ Reuse cool path resolution features of sys_move_mount
      ● Lookup at fd, symlink nofollow
  ○ Security changes
    ■ Access by mountpoint
    ■ Copy from mount with narrow root prohibited
    ■ Copy from mount with locked children in place prohibited
  ○ Criu part proof of concept rework [6].
Kernel patch status

- Cross-namespace propagation group setting allowed
  - Let's save some time instead of setns(from_mntns) + open_tree(from, OPEN_TREE_CLONE) + setns(to_mntns) + move_mount(anon, to, MOVE_MOUNT_SET_GROUP)
  - Check that we are allowed to modify both mount namespaces
- Difference to regular move_mount
  - We don't need to check for MNT_LOCKED, d_is_dir matching, unbindable, nsfs loops and ancestor relation as we don't move mounts.
- namespace_lock & no “new” loops
To do
To do

- Fix zdtm uns flavour tests in mount-v2 poc (in Virtuozzo version with old kernel patch they pass)
- Copy sharing from external mounts resolve external mountpoint
- Use cool path resolution features of move_mount
- Fix copy from narrow root problem
- Merge mount-v2 to mainstream CRIU
- Document MOVE_MOUNT_SET_GROUP
Extra to do… / Ideas

- (kernel) need interface to get mount tag mntns (sysfs, nfs, mqueue, proc)
- (criu) unix socket bind-mounts
  - SIOCUNIXFILE gives “file” of unix socket
  - Unix socket should be created before bind-mounting it in topologically right mntns
  - We can use cross-namespace bindmounts to solve it
- (kernel) need interface to set mount MNT_LOCKED
- (criu) handle files on detached mounts (by s_dev)
- (criu) nested userns owned mountns-es
- (criu/kernel) get full mountinfo of mntns with overmounted “/” (by ebpf?)
- (criu) port opening overmounted files from VZ criu
1. Previous talk: https://www.linuxplumbersconf.org/event/7/contributions/640/
2. Original patch "mnt: allow to add a mount into an existing group":
   https://lore.kernel.org/linux-api/20170428051831.20084-1-avagin@openvz.org/
3. Current patch "move_mount: allow to add a mount into an existing group":
   https://lore.kernel.org/linux-api/20210715100714.120228-1-ptikhomirov@virtuozzo.com/
4. Commit to linux master:
   https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=1dd5915a5cbd
5. Order inversion problem:
   https://gist.github.com/Snorch/df0a31049057c8e189c169a9e3eefa75/
   https://lore.kernel.org/linux-api/aba1e14c-8af8-e171-dbf8-c9000ddcc070@virtuozzo.com/
6. Mounts v2 draft rework on MOVE_MOUNT_SET_GROUP:
   https://github.com/Snorch/criu/commits/mount-v2-poc
7. Mounts-v2 full algorithm description https://criu.org/Mounts-v2