Subject: Re: [PATCH 2.6.19-rc3] VFS: per-sb dentry lru list
Posted by Eric Dumazet on Mon, 30 Oct 2006 15:08:25 GMT
View Forum Message <> Reply to Message

On Monday 30 October 2006 15:24, Vasily Averin wrote:
> Andrew Morton wrote:
> > On Fri, 27 Oct 2006 18:05:50 +0400
> >
> > Vasily Averin <vvs@sw.ru> wrote:
> >> Virtuozzo/OpenVZ linux kernel team has discovered that umount/remount
> >> can last for hours looping in shrink_dcache_sb() without much successes.
> >> Since during shrinking s_umount semaphore is taken lots of other
> >> unrelated operations like sync can stop working until shrink finished.
> >
> > Did you consider altering shrink_dcache_sb() so that it holds onto
> > dcache_lock and moves all the to-be-pruned dentries onto a private list
> > in a single pass, then prunes them all outside the lock?
>
> At the first glance it is wrong because of 2 reasons:
> 1) it continues to check the whole global LRU list (we propose to use
> per-sb LRU, it will provide very quick search)

Quick search maybe, but your patch adds 2 pointers to each dentry in the
system... That's pretty expensive, as dentries are already using a *lot* of
ram.

Maybe an alternative would be to not have anymore a global dentry_unused, but
only per-sb unused dentries lists ?

> 2) we have not any guarantee that someone will add new unused dentries to
> the list when we prune it outside the lock. And to the contrary, some of
> unused dentries can be used again. As far as I understand we should hold
> dcache_lock beginning at the removing dentry from unused_list until
> dentry_iput() call.
>
> David did it inside shrink_dcache_for_umount() just because it have
> guarantee that all the filesystem operations are finished and new ones
> cannot be started.