

---

Subject: Re: [patch 1/5][RFC - ipv4/udp checkpoint/restart] : add lookup for unhashed inode

Posted by [serue](#) on Wed, 06 Jun 2007 14:21:34 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Quoting dlezcano@fr.ibm.com (dlezcano@fr.ibm.com):

> The socket relies on the sockfs. In some cases, the socket are orphans and  
> it is not possible to access them via a file descriptor, this is the case for  
> example for timewait sockets. Hopefully, an inode is still usable to specify  
> a socket. This one can be retrieved from /proc/net/tcp for orphan sockets or  
> from a fstat.

>

> When a socket is created the socket inode is added to the sockfs.

> Unfortunately, this one is not stored into the hashed inode list, so

> I need a helper to browse the inode list contained in the superblock  
> of the sockfs.

>

> This is one solution, another solution is to stored the inode into  
> the hashed list when socket is created.

I assume that would be unacceptable overhead on a very busy server.  
Walking all the inodes NUM\_INODES(task\_set) for a checkpoint could  
be a real bottleneck, but at least it's only at checkpoint time.

Have you checked net-dev archives for discussions about not hashing  
these inodes? I suppose at some point you'll want to ask there what the  
preference is.

But certainly for now this seems the right approach.

> Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>

Acked-by: Serge E. Hallyn <serue@us.ibm.com>

(Or whatever tag they decide over on lkml that I should be using :)

thanks,

-serge

PS - I won't be acking other patches bc I just haven't looked at  
netlink enough - so don't read anything more into that :)

> ---

> fs/inode.c | 29 ++++++

> include/linux/fs.h | 1 +

> 2 files changed, 30 insertions(+)

>

> Index: 2.6.20-cr/fs/inode.c

> =====

```

> --- 2.6.20-cr.orig/fs/inode.c
> +++ 2.6.20-cr/fs/inode.c
> @@ -877,6 +877,35 @@
>
> EXPORT_SYMBOL(lookup);
>
> +
> +/**
> + * lookup_unhashed - search for an inode in the superblock
> + * @sb: super block of file system to search
> + * @ino: inode number to search for
> + *
> + * The lookup_unhashed browse the superblock inode list to find the inode.
> + *
> + * If the inode is found in the inode list stored in the superblock, the inode is
> + * with an incremented reference count.
> + *
> + * Otherwise NULL is returned.
> + */
> +struct inode *lookup_unhashed(struct super_block *sb, unsigned long ino)
> +{
> + struct inode *inode = NULL;
> +
> + spin_lock(&inode_lock);
> + list_for_each_entry(inode, &sb->s_inodes, i_sb_list)
> + if (inode->i_ino == ino) {
> + __iget(inode);
> + break;
> + }
> + spin_unlock(&inode_lock);
> + return inode;
> +
> +}
> +EXPORT_SYMBOL(lookup_unhashed);
> +
> +/**
> + * iget5_locked - obtain an inode from a mounted file system
> + * @sb: super block of file system
> Index: 2.6.20-cr/include/linux/fs.h
> =====
> --- 2.6.20-cr.orig/include/linux/fs.h
> +++ 2.6.20-cr/include/linux/fs.h
> @@ -1657,6 +1657,7 @@
> extern struct inode *lookup5(struct super_block *sb, unsigned long hashval,
> int (*test)(struct inode *, void *), void *data);
> extern struct inode *lookup(struct super_block *sb, unsigned long ino);
> +extern struct inode *lookup_unhashed(struct super_block *sb, unsigned long ino);
>

```

```
> extern struct inode * iget5_locked(struct super_block *, unsigned long, int (*test)(struct inode *,
void *), int (*set)(struct inode *, void *), void *);
> extern struct inode * iget_locked(struct super_block *, unsigned long);
>
> --
> _____
> Containers mailing list
> Containers@lists.linux-foundation.org
> https://lists.linux-foundation.org/mailman/listinfo/containers
```

---

Containers mailing list  
Containers@lists.linux-foundation.org  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---