Subject: Re: [PATCH] Send quota messages via netlink
Posted by Jan Kara on Mon, 03 Sep 2007 10:18:54 GMT
View Forum Message <> Reply to Message

On Fri 31-08-07 12:29:53, Balbir Singh wrote:
> Jan Kara wrote:
> >>>> + }
> >>>> + ret = nla_put_u32(skb, QUOTA_NL_A_QTYPE, dquot->dq_type);
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + ret = nla_put_u64(skb, QUOTA_NL_A_EXCESS_ID, dquot->dq_id);
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + ret = nla_put_u32(skb, QUOTA_NL_A_WARNING, warntype);
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + ret = nla_put_u32(skb, QUOTA_NL_A_DEV_MAJOR,
> >>>> + MAJOR(dquot->dq_sb->s_dev));
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + ret = nla_put_u32(skb, QUOTA_NL_A_DEV_MINOR,
> >>>> + MINOR(dquot->dq_sb->s_dev));
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + ret = nla_put_u64(skb, QUOTA_NL_A_CAUSED_ID, current->user->uid);
> >>>> + if (ret)
> >>>> + goto attr_err_out;
> >>>> + genlmsg_end(skb, msg_head);
> >>>> +
> >> Have you looked at ensuring that the data structure works across 32 bit
> >> and 64 bit systems (in terms of binary compatibility)? That's usually
> >> a nice to have feature.
> > Generic netlink should take care of this - arguments are typed so it
> > knows how much bits numbers have. So this should be no issue. Are there any
> > other problems that you have in mind?
> >
> Yes, but apart from that, if I remember Jamal Hadi's initial comments
> on taskstats, he recommended that we align everything to 64 bit so
> that the data is well aligned for 64 bit systems. You could also consider
  But each attribute is just one number (either 32 or 64 bit) so there's
not much to align. Also each attribute has its netlink header so alignment
is anyway hard to predict. Finally, this is by no means performance
critical - average system using quotas may get say 1 notification per user
per month?

> creating a data structure, document it's members, align them and use
> that to send out the data.
  I don't like sending one structure - by doing that you loose the

flexibility of netlink attributes...

> >>>> + ret = genlmsg_multicast(skb, 0, quota_genl_family.id, GFP_NOFS);
> >>>> + if (ret < 0 && ret != -ESRCH)
> >>>> + printk(KERN_ERR
> >>>> +   "VFS: Failed to send notification message: %d\n", ret);
> >>>> + return;
> >>>> +attr_err_out:
> >>>> + printk(KERN_ERR "VFS: Failed to compose quota message: %d\n", ret);
> >>>> +err_out:
> >>>> + kfree_skb(skb);
> >>>> +}
> >>>> +#endif
> >>> This is it. Normally netlink payloads are represented as a struct. How
> >>> come this one is built-by-hand?
> >>>
> >>> It doesn't appear to be versioned. Should it be?
> >>>
> >> Yes, versioning is always nice and genetlink supports it.
> >>
> It would nice for you to use the versioning feature.
  How does generic netlink support versioning? I have not found this
feature. Looking into Documentation/accounting/taskstats.txt it seems that
taskstats are versioning only the structure taskstats itself but not the
buch of attributes as a whole...

> >> The memory controller or VM would also be interested in notifications
> >> of OOM. At OLS this year interest was shown in getting OOM notifications
> >> and allow the user space a chance to handle the notification and take
> >> action (especially for containers). We already have containerstats for
> >> containers (which I was planning to reuse), but I was told that we would
> >> be interested in user space OOM notifications in general.
> >
> > Generic netlink can be used to pass this information (although in OOM
> > situation, it may be a bit hairy to get the network stack working...). But
> > I guess it's not related to my patch.
> >
> We could have a pre-allocated buffer stored at startup and use that for
> OOM notification. In the case of container OOM, we are likely to have
> free global memory. Working towards an infrastructure so that anybody can
> build on top of it and sending notifications on interesting events becomes
> easier would be nice. We can reuse code that way and add fewer bugs :-)
  Yes, but generic netlink itself is such an infrastructure, isn't it? It
is about 70 lines of code to implement notification for quota subsystem so
it's really simple...

        Honza
--

Jan Kara <jack@suse.cz>
SuSE CR Labs

_____

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