Suricata - Bug #2918
Unable to mmap, error Resource temporarily unavailable - err seems OS specific
04/03/2019 10:48 PM - Peter Manev

Status: New
Priority: Normal
Assignee: Community Ticket
Category: TBD
Target version: TBD
Affected Versions: TBD
Difficulty: TBD
Effort: TBD
Label: TBD

Description
Based on some previous discussion with Victor.

If we run Suricata with some non root user and specifically enable and use the mmap-locked option - Suricata does not start.
If use the same command but without the user (aka not run it as user "--user=mychemicalromance" but simply as "root") there is no err.
I can reproduce this on Stretch/Buster. But it seems the same is not an issue on Ubuntu.

```
/usr/bin/suricata -c /etc/suricata/suricata.yaml --af-packet --v --user=mychemicalromance --set="af-packet.1.mmap-locked = yes"
```

```
[2540] 3/4/2019 -- 23:53:39 - (suricata.c:1058) <Notice> (LogVersion) -- This is Suricata version 5.0.0-dev (rev 69d0d484e)
```

```
###BELOW IS IMPORTANT PART
```

Then I looked at the limits configuration

---

History

#1 - 04/04/2019 11:36 AM - Alexander Gozman
Probably this is due to missing CAP_IPC_LOCK.

#2 - 04/04/2019 07:07 PM - Joel Samaroo
Thanks for the pointer Alex!

Good news.

I stumbled on this article after researching a bit about IPC_MEM_LOCK … shouldn’t affect us since we’re on a newer kernel.

I checked the documentation of mmap and it says the following
[https://www.systutorials.com/docs/linux/man/2-mmap/](https://www.systutorials.com/docs/linux/man/2-mmap/)

" ===
Limits and permissions
In Linux 2.6.8 and earlier, a process must be privileged (CAP_IPC_LOCK) in order to lock memory and the RLIMIT_MEMLOCK soft resource limit defines a limit on how much memory the process may lock.
###BELOW IS IMPORTANT PART
Since Linux 2.6.9, no limits are placed on the amount of memory that a privileged process can lock and the RLIMIT_MEMLOCK soft resource limit instead defines a limit on how much memory an unprivileged process may lock.
Then I looked at the limits configuration

---
By default the limit for memlock is 64Kb… no good.

I changed the ulimit for suricata user to unlimited and it fixed the issue. Now I just need to figure out what the appropriate limit should be but looks like that was the issue.

Thank you for the help and the pointer.

#3 - 04/04/2019 07:53 PM - Peter Manev
In your case - what kernel version did you try this on - that is working ?

#4 - 04/04/2019 07:54 PM - Joel Samaroo
Joel Samaroo wrote:

Thanks for the pointer Alex!

Good news.

I stumbled on this article after researching a bit about CAP_IPC_LOCK … shouldn’t affect us since we’re on a newer kernel.

I checked the documentation of mmap and it says the following
https://www.systutorials.com/docs/linux/man/2-mmap/

" ===
Limits and permissions
In Linux 2.6.8 and earlier, a process must be privileged (CAP_IPC_LOCK) in order to lock memory and the RLIMIT_MEMLOCK soft resource limit defines a limit on how much memory the process may lock.
###BELOW IS IMPORTANT PART
Since Linux 2.6.9, no limits are placed on the amount of memory that a privileged process can lock and the RLIMIT_MEMLOCK soft resource limit instead defines a limit on how much memory an unprivileged process may ===

Then I looked at the limits configuration

By default the limit for memlock is 64Kb… no good.

I changed the ulimit for suricata user to unlimited and it fixed the issue. Now I just need to figure out what the appropriate limit should be but looks like that was the issue.

Thank you for the help and the pointer.

#5 - 04/04/2019 07:57 PM - Joel Samaroo
Hi Peter,

Kernel version 3.10.0

#6 - 04/05/2019 06:19 AM - Peter Manev
Ok, thanks.
It did not do the trick for me on 4.9.x (changing it to unlimited). Do you mind sharing exactly what you did that it works for you ?

#7 - 04/05/2019 06:46 AM - Joel Samaroo
Yes after updating limits.conf the effect will not take place until you either
1) log out and log back in as new user
2) ssh into the host again

In /etc/security/limits.conf
Add the following limit
suricata  hard  memlock   unlimited

You can confirm if it’s been set by running the following to verify
su -s /bin/bash -c 'ulimit -H -a' suricata | grep -i lock

That should let you know if the settings have been reloaded.

Note if you do not ssh back in the setting will not likely change. When testing on a box configured as run-level 5 (UI with X Windows) you must log out
then log back in as user. This is my understanding of how to trigger limits being updated.

#8 - 04/05/2019 06:51 AM - Joel Samaroo
Correction
updating limits.conf the effect will not take place until you either
1) log out and log back in as a user
2) ssh into the host again

#9 - 04/05/2019 07:59 AM - Peter Manev
I did that - I have actually rebooted.

#10 - 04/05/2019 08:04 AM - Victor Julien
You may need to set it in the systemd service file? https://github.com/elastic/elasticsearch/issues/22734

#11 - 04/05/2019 01:44 PM - Joel Samaroo
Ahh, yes I also added that line to my systemd suricata.service file. Looked like Peter was starting it from command line, so I didn't think to mention it!

#12 - 04/05/2019 02:56 PM - Joel Samaroo
When i try from command line i'm also not able to start suricata as unprivileged user, but via systemd (with LimitMEMLOCK=infinity) it works.
I checked the following stackexchange article
https://unix.stackexchange.com/questions/345595/how-to-set-ulimits-on-service-with-systemd
and it looks like LimitMEMLOCK maps to ulimit -l (memlock) so it's confusing why that would not work for command line.

#13 - 04/05/2019 04:32 PM - Joel Samaroo
- File limits comparison.jpg added

According to attachment there's definitely some issue that I'm having in my tests that's causing my normal user to cap out at 65536kb of locked mem limit.
Would be interested in what that looks like on your end Peter.

The way i'm getting this info is the following:
cat /proc/$(/ps -u suricata | tail -n1 | cut -f1 -d\) /limits

#14 - 04/05/2019 04:43 PM - Joel Samaroo
So this is interesting
if you set root ulimit for memory locking to unlimited (by default its 64kb or 65536 bytes) it works fine!

1. ulimit -H -l unlimited
2. ulimit -S -l unlimited
3. if [ -e /var/run/suricata.pid ];then rm /var/run/suricata.pid;fi ;/sbin/suricata -c /etc/suricata/suricata.yaml --pidfile /var/run/suricata.pid --af-packet


so if you're manually starting the command and using run-user (not via systemd) then you need to make sure the ulimit for root is also high evenough for memlock.

#15 - 04/05/2019 05:06 PM - Joel Samaroo
Please note -- setting the ulimit the way I did it above makes sure it's not persistent -- since i did not define it in any file. To make it persistent make the change in your file.

Moral of the store I suppose it that Systemd seems to override root's ulimit (not necessarily the user's ulimits here). It's an interesting lesson to learn, but it makes sense since all of this is set up before the privs are dropped down.
The other thing I'm wondering now, is why is it that if you are not dropping privs you are not given that error. The limit constraints are still there, so that is still a for mystery for me.

#16 - 04/05/2019 06:05 PM - Victor Julien
I think it makes sense. When Suricata starts it runs as root. Only after opening the AF_PACKET stuff the privs are dropped. So the limits that apply should be those for the root user.

#17 - 05/23/2019 10:02 PM - Andreas Herz
- Assignee set to Community Ticket
- Target version set to TBD
#18 - 07/27/2019 09:19 PM - Andreas Herz
Is this something we should document or where we can change something on suricata side?

#19 - 11/22/2019 05:25 PM - Joel Samaroo
I think it makes sense to document it as an OS Specific requirement (as noted above Ubuntu didn't seem to have any issues) when using run-user: with mmap_locked: together. Not sure where exactly this should be though. We did document internally in our organization, however I think that other's would benefit from noting the change in some sort of other documentation.

<table>
<thead>
<tr>
<th>Files</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>limits comparison.jpg</td>
<td>128 KB</td>
<td>04/05/2019</td>
<td>Joel Samaroo</td>
</tr>
</tbody>
</table>