

unable to find new object create with kmem_cache_create() in /proc/slabinfo

 stackoverflow.com/questions/24858424/unable-to-find-new-object-create-with-kmem-cache-create-in-proc-slabinfo

Ask Question

I have written a simple kernel module which allocates objects using the slab layer. This module uses kernel API's (`kmem_cache_{create,alloc,free}`). The version of the kernel I am working on 3.15.4-200.

Though, my code works as expected with no issues, I am unable to see the new object/slab "my_cache" created using `kmem_cache_create()` when I `grep /proc/slabinfo`. The objects created are a simple list of objects, being inserted or removed from the list.

Note: My module works fine with no issues. I can view the slab creating under, `/sys/kernel/slab/my_cache`, but not in `/proc/slabinfo` or `vmstat -m` or `slabtop`

```
mycache = kmem_cache_create("my_cache",
                            sizeof(struct mystruct),
                            0, SLAB_HWCACHE_ALIGN, NULL);
if (mycache == NULL)
    return -ENOMEM;
```

,

```
$ sudo cat /sys/kernel/slab/my_cache/objects
49108 N0=49108
$ sudo cat /sys/kernel/slab/my_cache/object_size
64
$ sudo cat /sys/kernel/slab/my_cache/order
0
$ sudo cat /sys/kernel/slab/my_cache/aliases
12
$ sudo cat /sys/kernel/slab/my_cache/cache_dma
0
$ sudo cat /sys/kernel/slab/my_cache/slab_size
64
$ sudo cat /sys/kernel/slab/my_cache/trace
0
$ sudo cat /sys/kernel/slab/my_cache/validate
$ sudo cat /sys/kernel/slab/my_cache/total_objects
49920 N0=49920
```

,

[memory-management linux-kernel](#)

2 Answers 2

[up vote 7](#) [down vote](#)

`kmem_cache_create` usually tries to merge this cache with other caches. So for example if it found another cache for the same size with the same properties (slab is going to be poisoned, etc) it will just use this cache instead of creating a new cache.

If you really want to make sure that it'll definitely create a new cache for you (= appear in `slabinfo`), you can always pass a valid constructor to `kmem_cache_create`. Something like this:

```
static void mystruct_constructor(void *addr)
{
    memset(addr, 0, sizeof(struct mystruct));
}

mycache = kmem_cache_create("my_cache",
    sizeof(struct mystruct),
    0, 0, mystruct_constructor);
```

A constructor is usually called for every object allocation from this cache.

answered Jul 21 '14 at 18:33



[KarimRaslan](#)

17316

[up vote 4](#) [down vote](#) [accepted](#)

passing the additional flag `SLAB_POISON` solves the issue.

[from link](#) `SLAB_POISON` - Poison the slab with a known test pattern (a5a5a5a5) to catch references to uninitialised memory.

answered Jul 21 '14 at 10:01



[askb](#)

3,9941431

Your Answer

Sign up or log in

Sign up using Google

Sign up using Facebook

Sign up using Email and Password

Post as a guest

[discard](#)

By clicking "Post Your Answer", you acknowledge that you have read our updated [terms of service](#), [privacy policy](#) and [cookie policy](#), and that your continued use of the website is subject to these policies.

Not the answer you're looking for? Browse other questions tagged [memory-management](#) [linux-kernel](#) or ask your own question.
