

## Re: tbench regression in 2.6.25-rc1

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-02/msg10236.html>

---

- *From:* "Zhang, Yanmin" <[yanmin\\_zhang@xxxxxxxxxxxxxxxxx](mailto:yanmin_zhang@xxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 20 Feb 2008 16:41:04 +0800
- 

Comparing with kernel 2.6.24, tbench result has regression with 2.6.25-rc1.

- 1) On 2 quad-core processor stoakley: 4%.
- 2) On 4 quad-core processor tigerton: more than 30%.

bisect located below patch.

b4ce92775c2e7ff9cf79cca4e0a19c8c5fd6287b is first bad commit  
commit b4ce92775c2e7ff9cf79cca4e0a19c8c5fd6287b  
Author: Herbert Xu <[herbert@xxxxxxxxxxxxxxxxx](mailto:herbert@xxxxxxxxxxxxxxxxx)>  
Date: Tue Nov 13 21:33:32 2007 -0800

[IPV6]: Move nfheader\_len into rt6\_info

The dst member nfheader\_len is only used by IPv6. It's also currently creating a rather ugly alignment hole in struct dst. Therefore this patch moves it from there into struct rt6\_info.

Above patch changes the cache line alignment, especially member \_\_refcnt. I did a testing by adding 2 unsigned long padding before lastuse, so the 3 members, lastuse/\_\_refcnt/\_\_use, are moved to next cache line. The performance is recovered.

I created a patch to rearrange the members in struct dst\_entry.

With Eric and Valdis Kletnieks's suggestion, I made finer arrangement.

1) Move tclassid under ops in case CONFIG\_NET\_CLS\_ROUTE=y. So sizeof(dst\_entry)=200 no matter if CONFIG\_NET\_CLS\_ROUTE=y/n. I tested many patches on my 16-core tigerton by moving tclassid to different place. It looks like tclassid could also have impact on performance.

If moving tclassid before metrics, or just don't move tclassid, the performance isn't good. So I move it behind metrics.

2) Add comments before \_\_refcnt.

On 16-core tigerton:

If CONFIG\_NET\_CLS\_ROUTE=y, the result with below patch is about 18% better than the one without the patch;

If CONFIG\_NET\_CLS\_ROUTE=n, the result with below patch is about 30% better than the one without the patch.

Re: tbench regression in 2.6.25-rc1

With 32bit 2.6.25-rc1 on 8-core stoakley, the new patch doesn't introduce regression.

Thank Eric, Valdis, and David!

Signed-off-by: Zhang Yanmin <yanmin.zhang@xxxxxxxx>

Acked-by: Eric Dumazet <dada1@xxxxxxxxxxxx>

---

```
--- linux-2.6.25-rc1/include/net/dst.h 2008-02-21 14:33:43.000000000 +0800
+++ linux-2.6.25-rc1_work/include/net/dst.h 2008-02-22 12:52:19.000000000 +0800
@@ -52,15 +52,10 @@ struct dst_entry
unsigned short header_len; /* more space at head required */
unsigned short trailer_len; /* space to reserve at tail */

- u32 metrics[RTAX_MAX];
- struct dst_entry *path;
-
- unsigned long rate_last; /* rate limiting for ICMP */
unsigned int rate_tokens;
+ unsigned long rate_last; /* rate limiting for ICMP */

-#ifdef CONFIG_NET_CLS_ROUTE
- __u32 tclassid;
-#endif
+ struct dst_entry *path;

struct neighbour *neighbour;
struct hh_cache *hh;
@@ -70,10 +65,20 @@ struct dst_entry
int (*output)(struct sk_buff*);

struct dst_ops *ops;
-
- unsigned long lastuse;
+
+ u32 metrics[RTAX_MAX];
+
+#ifdef CONFIG_NET_CLS_ROUTE
+ __u32 tclassid;
+#endif
+
+ /*
+ * __refcnt wants to be on a different cache line from
+ * input/output/ops or performance tanks badly
+ */
atomic_t __refcnt; /* client references */
int __use;
+ unsigned long lastuse;
union {
struct dst_entry *next;
```

Re: tbench regression in 2.6.25-rc1

Re: tbench regression in 2.6.25-rc1

```
struct rtable *rt_next;
```

--

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>