

Linux Cross Reference

Free Electrons

Embedded Linux Experts

• *source navigation* • [diff markup](#) • [identifier search](#) • [freetext search](#) •

Version:

[2.0.40](#) [2.2.26](#) [2.4.37](#) [3.1](#) [3.2](#) [3.3](#) [3.4](#) [3.5](#) [3.6](#) [3.7](#) [3.8](#) [3.9](#) [3.10](#) [3.11](#) [3.12](#) [3.13](#) [3.14](#) [3.15](#) [3.16](#) [3.17](#)

Linux/net/ipv4/tcp_scalable.c

```

1  /* Tom Kelly's Scalable TCP
2  *
3  * See http://www.deneholme.net/tom/scalable/
4  *
5  * John Heffner <jheffner@sc.edu>
6  */
7
8  #include <linux/module.h>
9  #include <net/tcp.h>
10
11 /* These factors derived from the recommended values in the aer:
12 * .01 and 7/8. We use 50 instead of 100 to account for
13 * delayed ack.
14 */
15 #define TCP_SCALABLE_AI_CNT      50U
16 #define TCP_SCALABLE_MD_SCALE    3
17
18 static void tcp_scalable_cong_avoid(struct sock *sk, u32 ack, u32 acked)
19 {
20     struct tcp_sock *tp = tcp_sk(sk);
21
22     if (!tcp_is_cwnd_limited(sk))
23         return;
24
25     if (tp->snd_cwnd <= tp->snd_ssthresh)
26         tcp_slow_start(tp, acked);
27     else
28         tcp_cong_avoid_ai(tp, min(tp->snd_cwnd, TCP_SCALABLE_AI_CNT));
29 }
30
31 static u32 tcp_scalable_ssthresh(struct sock *sk)
32 {
33     const struct tcp_sock *tp = tcp_sk(sk);
34     return max(tp->snd_cwnd - (tp->snd_cwnd >> TCP_SCALABLE_MD_SCALE), 2U);
35 }
36
37
38 static struct tcp_congestion_ops tcp_scalable_read_mostly = {
39     .sssthresh      = tcp_scalable_ssthresh,
40     .cong_avoid      = tcp_scalable_cong_avoid,
41
42     .owner           = THIS_MODULE,
43     .name            = "scalable",

```

```
44 };
45
46 static int __init tcp_scalable_register(void)
47 {
48     return tcp_register_congestion_control(&tcp_scalable);
49 }
50
51 static void __exit tcp_scalable_unregister(void)
52 {
53     tcp_unregister_congestion_control(&tcp_scalable);
54 }
55
56 module_init(tcp_scalable_register);
57 module_exit(tcp_scalable_unregister);
58
59 MODULE_AUTHOR("John Heffner");
60 MODULE_LICENSE("GPL");
61 MODULE_DESCRIPTION("Scalable TCP");
62
```

This page was automatically generated by [LXR](#) 0.3.1 ([source](#)). • Linux is a registered trademark of Linus Torvalds • [Contact us](#)

- [Home](#)
- [Development](#)
- [Services](#)
- [Training](#)
- [Docs](#)
- [Community](#)
- [Company](#)
- [Blog](#)