

Linux Kernel and Linux Foundation

By *Roy Schestowitz*

Created *19/11/2019 - 5:14pm*

Submitted by Roy Schestowitz on Tuesday 19th of November 2019 05:14:57 PM Filed under [Linux](#) [1]

- [Linux's Crypto API Is Adopting Some Aspects Of Zinc, Opening Door To Mainline WireGuard](#) [2]

Mainlining of the WireGuard secure VPN tunnel was being held up by its use of the new "Zinc" crypto API developed in conjunction with this network tech. But with obstacles in getting Zinc merged, WireGuard was going to be resorting to targeting the existing kernel crypto interfaces. Instead, however, it turns out the upstream Linux crypto developers were interested and willing to incorporate some elements of Zinc into the existing kernel crypto implementation.

Back in September is when Jason Donenfeld decided porting WireGuard to the existing Linux crypto API was the best path forward for getting this secure networking functionality into the mainline kernel in a timely manner. But since then other upstream kernel developers working on the crypto subsystem ended up with patches incorporating some elements of Zinc's design.

- [zswap: use B-tree for search](#) [3]

The current zswap implementation uses red-black trees to store entries and to perform lookups. Although this algorithm obviously has complexity of $O(\log N)$ it still takes a while to complete lookup (or, even more for replacement) of an entry, when the amount of entries is huge (100K+).

B-trees are known to handle such cases more efficiently (i. e. also with $O(\log N)$ complexity but with way lower coefficient) so trying zswap with B-trees was worth a shot.

The implementation of B-trees that is currently present in Linux kernel isn't really doing things in the best possible way (i. e. it

has recursion) but the testing I've run still shows a very significant performance increase. The usage pattern of B-tree here is not exactly following the guidelines but it is due to the fact that pgoff_t may be both 32 and 64 bits long.

- [Zswap Could See Better Performance Thanks To A B-Tree Search Implementation \[4\]](#)

For those using Zswap as a compressed RAM cache for swapping on Linux systems, the performance could soon see a measurable improvement.

Developer Vitaly Wool has posted a patch that switches the Zswap code from using red-black trees to a B-tree for searching. Particularly for when having to search a large number of entries, the B-trees implementation should do so much more efficiently.

- [AT&T Finally Opens Up dNOS "DANOS" Network Operating System Code \[5\]](#)

One and a half years late, the "DANOS" (known formerly as "dNOS") network operating system is now open-source under the Linux Foundation.

AT&T and the Linux Foundation originally announced their plan in early 2018 with pushing for this network operating system to be used on more mobile infrastructure. At the time they expected it to happen in H2'2018, but finally on 15 November 2019 the goal came to fruition.

[Linux](#)

Source URL: <http://www.tuxmachines.org/node/130684>

Links:

[1] <http://www.tuxmachines.org/taxonomy/term/63>

[2] https://www.phoronix.com/scan.php?page=news_item&px=Crypto-API-Doing-Some-Zinc

[3] <https://lkml.org/lkml/2019/11/17/149>

[4] https://www.phoronix.com/scan.php?page=news_item&px=Zswap-Linux-B-Tree-Search

[5] https://www.phoronix.com/scan.php?page=news_item&px=ATT-DANOS-Now-Posted