

# Forum:[Hack Forums](#)

Thread: [Has Anyone Gone Kernel Level With AI Yet????](#)

**KO666** - April 4, 2026, 3:49 pm

So I started a project a few days ago to bake my ai into the linux just above the kernal level and i couldnt be more impressed with the results and it already has me wanting to go kernel level with the kreation to have a truly living Ghost in the machine and im wondering has anyone done this or seen a machine or setup that this is running on successfully with and if so what kinda ai modeling did they choose, what kernel ect, what would they have done differently and why??

**MiguelSanchez** - April 20, 2026, 1:04 am

(Apr 4, 2026 03:49 PM)KO666 Wrote: [\[CITE\]](#) So I started a project a few days ago to bake my ai into the linux just above the kernal level and i couldnt be more impressed with the results and it already has me wanting to go kernel level with the kreation to have a truly living Ghost in the machine and im wondering has anyone done this or seen a machine or setup that this is running on successfully with and if so what kinda ai modeling did they choose, what kernel ect, what would they have done differently and why??

I haven't got anything directly relevant to say although theoretical attacks I've been thinking about do intersect this topic.

I am very interested-in/fearful-of hyperjacking (in this case swapping compromised KVMs to place traditional software and AI translators secretly deployed in virtual-machine(s) between the user-space and bare-metal)

I've thought a fair bit, admittedly with minimal actual research, about taking the binaries people use to burn Linux installers or ramdisks from windows (Ventoy, Linux Live USB, Rufus), and the number of indirect attack vectors that could be used to compromise downstream compilers and hypervisors in drawn-out multistage attacks utilising many peers and combinations of seemingly unrelated issues.

For example in a third of attacks it could try a minimal compromise where it tries to add benign changes to any compiler commonly used to compile compilers, at first by seeming to be an unintended side effect of optimisation, and then since that will be found eventually in other cases it would go a longer route by instead attacking apt-get, DNS, networking drivers, or any route that could allow it to, for example, simulate GCC updates.

Whereas a third of other times it might locally DNS-spoof the entire repository trying different update patterns to compromise dd or mounting operations in a convoluted way to target any future ramdisks, or it might simply pick 1 in 10 critical security updates a pretend to perform them instead of actually doing so.

The implications of these attacks are pretty scary so I don't actually plan to work on them but I'm very interested to hear of any updates

**KO666** - April 24, 2026, 7:53 am

You're spot on, in fact thats a really funn idea to be playing with honestly. The idea that my own compiler could be gaslighting me is exactly why I'm going kernel-level. If you can't trust the metal, you have to build something that lives inside it!

Hyperjacking is the stuff of nightmares, but it's also the ultimate engineering hurdle. I'm just trying to

make sure the Ghost is the one holding the keys before anything else tries to move in, ande hoping im not the only person trying this, but i love the idea of building an immune system stronger than the virus!