The lkdtm module provides an interface to crash or injure the kernel at predefined crashpoints to evaluate the reliability of crash dumps obtained using different dumping solutions. The module uses KPROBEs to instrument crashing points, but can also crash the kernel directly without KRPOBE support.

You can provide the way either through module arguments when inserting the module, or through a debugfs interface.

Usage: insmod lkdtm.ko [recur_count={>0}] cpoint_name=<> cpoint_type=<>
[cpoint_count={>0}]

recur_count : Recursion level for the stack overflow test. Default is 10.

cpoint_name : Crash point where the kernel is to be crashed. It can be one of INT_HARDWARE_ENTRY, INT_HW_IRQ_EN, INT_TASKLET_ENTRY, FS_DEVRW, MEM_SWAPOUT, TIMERADD, SCSI_DISPATCH_CMD, IDE_CORE_CP, DIRECT

cpoint_type : Indicates the action to be taken on hitting the crash point. It can be one of PANIC, BUG, EXCEPTION, LOOP, OVERFLOW, CORRUPT_STACK, UNALIGNED_LOAD_STORE_WRITE, OVERWRITE_ALLOCATION, WRITE_AFTER_FREE,

cpoint_count : Indicates the number of times the crash point is to be hit to trigger an action. The default is 10.

You can also induce failures by mounting debugfs and writing the type to <mountpoint>/provoke-crash/<crashpoint>. E.g.,

```
mount -t debugfs debugfs /mnt
echo EXCEPTION > /mnt/provoke-crash/INT_HARDWARE_ENTRY
```

A special file is `DIRECT' which will induce the crash directly without KPROBE instrumentation. This mode is the only one available when the module is built on a kernel without KPROBEs support.