Run Tests without kunit_tool

If we do not want to use kunit_tool (For example: we want to integrate with other systems, or run tests on real hardware), we can include KUnit in any kernel, read out results, and parse manually.

> **Note**
>
> KUnit is not designed for use in a production system. It is possible that tests may reduce the stability or security of the system.

Configure the Kernel

KUnit tests can run without kunit_tool. This can be useful, if:

- We have an existing kernel configuration to test.
- Need to run on real hardware (or using an emulator/VM kunit_tool does not support).
- Wish to integrate with some existing testing systems.

KUnit is configured with the `CONFIG_KUNIT` option, and individual tests can also be built by enabling their config options in our `.config`. KUnit tests usually (but don't always) have config options ending in `_KUNIT_TEST`. Most tests can either be built as a module, or be built into the kernel.

> **Note**
>
> We can enable the `KUNIT_ALL_TESTS` config option to automatically enable all tests with satisfied dependencies. This is a good way of quickly testing everything applicable to the current config.

Once we have built our kernel (and/or modules), it is simple to run the tests. If the tests are built-in, they will run automatically on the kernel boot. The results will be written to the kernel log (`dmesg`) in TAP format.

If the tests are built as modules, they will run when the module is loaded.

```
# modprobe example-test
```

The results will appear in TAP format in `dmesg`.

> **Note**
>
> If `CONFIG_KUNIT_DEBUGFS` is enabled, KUnit test results will be accessible from the `debugfs` filesystem (if mounted). They will be in `/sys/kernel/debug/kunit/<test_suite>/results`, in TAP format.