Collaboration/unification around unit testing frameworks

From the initial reactions and interest I have seen wrt. KTF (http://heim.ifi.uio.no/~knuto/ktf/, https://github.com/oracle/ktf) and the discussions on LKML around KUnit (https://lkml.org/lkml/2018/11/29/82), it seems there’s a general belief that some form of unit test framework like these can be a good addition to the tools and infrastructure already available in the kernel.

It seems however that different people have different notions about what and how such a framework should ideally look, and what features belong there. I’d like to see if we can bring that discussion forward by focusing on some of these items, where people seem to have quite differing views depending on where they come from. Here is a non extensive list of some topics that seems to pop up when this gets discussed:

• "Purity" of unit testing - what constitutes a "unit" in the kernel?
• Testing kernel code - user space vs kernel space? (both useful)
• Immediate development/debugging requirements vs longer term needs
• Driver/hardware interaction testing?
• "Neat"-factor
• ease of use
• Network testing (more than 1 kernel involved)
• How to best integrate with existing test infrastructure in the kernel
• Unification and simplication options ...

I’d like to make a short intro into this, and hopefully we can have some good exchange based on that.

I agree to abide by the anti-harassment policy

Yes

I confirm that I am already registered for LPC 2019

Primary author: Dr OMANG, Knut (Oracle)
Presenter: Dr OMANG, Knut (Oracle)
Session Classification: Testing and Fuzzing MC