



Contribution ID: 432

Type: **not specified**

Exploring possibilities for integrating StrictDoc with ELISA's requirements template approach for the Linux kernel

Friday 12 December 2025 12:25 (15 minutes)

The ELISA project currently works on bringing the Linux kernel closer to safety compliance by proposing enhancements to the kernel documentation. This includes a model expressed as requirement templates inlined to source code. At the same time, comparable efforts with a similar goal are also ongoing in the wider open-source ecosystem. For example, the Zephyr OS is using the FLOSS StrictDoc model and tool to capture and process requirements. Another example is Linutronix who reached IEC 62443 4-2 security certification by using StrictDoc to trace requirements, design and tests for their Debian based IGLOS Secure Beacon embedded system.

This talk picks up the work of ELISA and compares it to a typical StrictDoc setup with the intention to show that both efforts could be joined. While ELISA focuses on the model and assumes tools will emerge from the community, StrictDoc both defines a similar model and provides mature tooling to validate and render requirements, design and tests. We'll see that the majority of the needs set by ELISA are already fulfilled by StrictDoc. Notably, ELISA's SPDX-REQ-* tags can be represented and parsed with StrictDoc's "Parsing SDoc nodes from source code comments" feature. The remaining gap is drift detection, i.e. to store the hash sum over project ID, source file, requirement text and proximal code with the intention to signal a human to check if requirement and code still align when some of the criteria changes. StrictDoc knows meta data, content and proximal code by function name, class name and line ranges, but has no hash generation built in yet. However, StrictDoc is advanced in defining language constructs as part of the model (functions, classes, test cases). It is also advanced in applicability, where for example OEM requirements and OSS project requirements can be linked together in one compatible model and in consequence can be processed and validated by the same tool in a single run. Various input/output format converters exist and customization of validation is next on the roadmap.

The talk concludes with the proposal that StrictDoc could close its gaps by implementing hash generation, optimizing ELISA requirement template parsing and by setting up conformity tests to maintain compatibility. ELISA could in turn list StrictDoc as one of their reference tools, and kernel developers will be invited to try it in their workflow.

Primary author: DEIMINGER, Tobias (Linutronix GmbH)

Presenter: DEIMINGER, Tobias (Linutronix GmbH)

Session Classification: Safe Systems with Linux MC

Track Classification: Safe Systems with Linux MC