1. IEC 62304, paragraph 3.29 - From Medical Domain, Linux is considered as a SOUP
2. IEC 62304, paragraph 5.1.1 - software development plan must address “software configuration and change management, including SOUP configuration items and software used to support development”.
3. IEC 62304, paragraph 5.1.7 - With Linux, risk analysis would need to be produced by the development organization, and it is not clear how it could be produced without a failure rate figure being available.

All fault reports in the file system, communications stacks, drivers, etc. would also have to be analyzed.

1. IEC 62304, paragraph 5.3.3 - The functional safety requirements would have to be defined. Compliance team would then have to carry out the necessary validation to ensure that the Linux OS complies with these requirements.
2. IEC 62304, paragraph 6.1 - Publish anomaly SOUP lists

TIR80002, paragraph 3.4.1 - If SOUP (Linux) is used, then actively monitoring and evaluating publicly available anomaly lists and information about the SOUP field performance should be planned.

1. IEC 62304, paragraph B.1.2 - MDM’s Responsibility with SOUP - If Linux is used for a medical device or medical software, the device manufacturer must assume responsibility for the full risk management of the Linux variant used.
2. TIR80002, paragraph 6.2.3 - Failure modes of SOUP - The higher the potential risks of the medical device, the more closely potential failure modes of SOUP should be analyzed and risk control measures identified.
3. TIR80002, section 9 - Inside the SOUP OS, identify the SOUP components and remove the unnecessary SOUP components.
4. ISO 14971, paragraph 4.3 (called up in TIR80002, table B) - Failure modes and frequencies of the SOUP components - Warns the manufacturer against omitting the failure of SOUP in the fault tree (or FMECA) for the medical device. Again, this warning stresses the need for a fault tree.
5. TIR80002, table C.1 - Failure to identify safety-related aspects of the Linux architecture, results in unknown safety risks when these architectural elements are subsequently changed or eliminated.
6. For demanding applications (blood diagnostics, ultrasound imaging, infusion delivery, heart monitoring and resuscitation, and robotic surgery), the standard Linux kernel can’t deliver the safety-critical capabilities that many medical embedded devices require.