The Capability Hardware Enhanced RISC Instructions (CHERI) CPU architecture is motivated by the phenomenon of privilege patching. The main motivation for Cheri is to provide a rich, flexible, extensible, and efficient approach to software security. The architecture is designed to allow the co-existence of normal programs and security paths on the same processor. 

CHERI provides a framework for the software security structures and design principles are reinforced by Capability Hardware Enhanced RISC Instructions (CHERI) and Temporally Enhanced Security Logic Assertions (TESLA).

CHERI addresses those problems through a flexible and compiler-friendly approach to support object capability security model. It provides a protection space in a natural and lightweight manner, allowing users to manipulate it in a community or on its own.

CHERI is a flexible, extensible, and efficient approach to software security. The architecture is designed to allow the co-existence of normal programs and security paths on the same processor. It provides a framework for the software security structures and design principles are reinforced by Capability Hardware Enhanced RISC Instructions (CHERI) and Temporally Enhanced Security Logic Assertions (TESLA).