Secure Design principles (SD-1)

Table of contents

Revision History

Revision No	Date	Change description	Author	Reviewed by
001	2022-12-20	Draft document for Secure Design principles (SD-1)	Sai Ashrith	Dinesh Kumar
002	2022-12-23	Updated document after review	Sai Ashrith	Dinesh Kumar

---

Introduction

This document explains the secure design in CIP which counters misuse of the product by considering various scenarios for threats and their mitigations involved in the design regarding interfaces of CIP. Details about how the interfaces are implemented in CIP, involved users, allowed privileges and methods to use the CIP interface are documented.

Scope

CIP being a generic platform for product development on top of it, security context documented as per requirement is generic in nature. The threats mentioned in this document are only related to externally accessible interface.

SD-1 : Secure Design principles

- CIP being an Open Source project is externally accessible by any user and it provides the privilege to suggest changes in the CIP source code from a development perspective. But the acceptance of the change depends on the respective CIP repository maintainer. In run time all the CIP interfaces are same as in a standard Linux based system. Any changes in the internal interfaces should be documented by the end users.
- CIP strictly follows upstream first policy when it comes to make any changes in the system including CIP Kernel and CIP root file system.
- Considering only the external access to CIP interface, a generic security context document is documented in which **CIP_SEC_IEC_FUNC_REQ_1** and **CIP_SEC_IEC_FUNC_REQ_3** explain the security implications of CIP interface.
- There are no particular potential users to point out to because CIP is an Open source platform for any user to access its source code to build a product based on their requirement by accessing assets like **database files**, **configuration files**, **cryptographic key stores**, **audit logs**, **IPCs** etc. For development, the security environment in CIP is mentioned in this document.

- The external actors accessing the CIP-Core and CIP-Kernel is shown as a Data flow diagram in which red-dotted boundary is the trust boundary for the respective scenario and in In run time scenarios as a Networking switch and as a PLC.
- **6.2.1_191**, **6.2.2_197**, **6.2.2_200**, **6.2.2_202**, **6.2.2_204** threats mentioned in the threat modelling document are the interface related threats to CIP. Details about the actions taken by CIP to fulfill the requirements are mentioned in these documents Identification & Authentication control and User Control documents.
- In development phase of CIP, the source code merge privileges are given only to **CIP maintainers**. The CIP developers can only send the merger request.
- In run time there is an authorization enforcement for all its users based on their assigned privileges. **sudo** package was added to the IEC layer to make this possible. This role-based access is implemented on application level.
- Efibootguard is a third party component used in CIP to make the run time interface secure, as it provides Secure boot along with a secure fail safe update mechanism.
- External actors can use **https** and **SSH** protocols to download the CIP source code for their end product development.