PLC Security TOP 20 LIST

Top 20 Secure PLC Coding Practices Application Notes

Siemens

Continuous Manufacturing Platforms - LCS

Secure PLC Coding Practices Application

Use Case Yellowfin Line Control System / Siemens

Use Case Introduction

Type of organization

Integrator

PLC make / model

Siemens CPU 410 Single AS

Scenario

Siemens will provide a monitoring system based on SIMATIC PCS 7, our flagship Distributed Control System for unit operation integration. The system provides centralized data acquisition, alarming, and historization, as well as holistic, flexible control of the entire line to coordinate quality production. We are proposing a flexible unit operation integration model to minimize initial deployment effort and pre-validated libraries to design the system with a future cGMP environment in mind.

This document describes which ones of the Top 20 Secure PLC Coding Practices were adopted while developing this project.

Secure PLC Coding Practices Application

Use Case Yellowfin Line Control System / Siemens

Application Statement

#	Practice Title	Applied? (yes / no)	Notes
1	**Modularize PLC Code**	Yes	PLC code is modularized using functions, function blocks and organization blocks. The Advanced Process Library (APL) which is a library developed by Siemens Headquarters has been used as the primary library for this project. A single block was custom developed for this project to get data from the Shakers. APL blocks were used to display the values in the HMI.
2	**Track operating modes**	Yes	An alarm is generated in the HMI when the PLC is not in RUN mode. A password has also been configured to prevent changes in the logic.
3	**Leave operational logic in the PLC**	Yes	Alarm setpoints, timers, and integrators are all based on Siemens APL library and apart of the PLC code.
4	**Use PLC flags as integrity checks**	Yes	Alarms have been added in case the APL blocks report an error during execution.
5	**Use cryptographic and / or checksum integrity checks for PLC code**	No	Not available for this type of controller.
6	**Validate timers and counters**	Yes	No counters/timers have been programmed in the PLC.
7	**Validate and alert for paired inputs / outputs**	Yes	There are no motors/valves connected to the Siemens controller for this project. The APL blocks have these alarms implemented, but they are currently not being used.
8	**Validate HMI input variables at the PLC level, not only at HMI**	Yes	APL blocks are preconfigured with limit checks in the internal logic.
9	**Validate indirections**	N/A	No arrays implemented in this project.
10	**Assign designated register blocks by function (read / write / validate)**	N/A	
11	**Instrument for plausibility checks**	N/A	Gradient alarms could be enabled for the APL Analog monitoring blocks.
12	**Validate inputs based on physical plausibility**	N/A	APL blocks have internal logic with timers to monitor plausibility of control devices (motors and valves)
13	**Disable unneeded / unused communication ports and protocols**	Yes	
14	**Restrict third-party data interfaces**	Yes	

#	Practice Title	Applied? (yes / no)	Notes
15	**Define a safe process state in case of a PLC restart**	Yes	Used first scan/initialization routines to ensure known state for latched bits and analog values.
16	**Summarize PLC cycle times and trend them on the HMI**	Yes	CPU used is being trended and historized. Also, alarms will occur if any limit is approached.
17	**Log PLC uptime and trend it on the HMI**	N/A	
18	**Log PLC hard stops and trend them on the HMI**	Yes	Generates and alarm in HMI.
19	**Monitor PLC memory usage and trend it on the HMI**	Yes	
20	**Trap false negatives and false positives for critical alerts**	No	

Note: Certain Top 20 Secure PLC Coding Practices were not applicable due to limitations to the technology, demarcation of scope, and requests from the client.

Application Details

Siemens provided a Line Control System (LCS) based on SIMATIC PCS 7, our flagship Distributed Control System, for unit operation integration and overall line monitoring. The continuous manufacturing line pilot encompasses of several off-the-shelf standalone unit operations and several custom unit operations. Using single-use components, peristaltic pumps and valves, the various individual processing steps are connected into a continuous production line. A SIMATIC PCS 7 system will be implemented as a Proof of Concept (POC) monitoring system, intended to capture data, alarms and events from the custom mini unit operations. The control system will record, and archive process data required to support the process validation and quality assurance programs implemented at the customer plant. No control from the PCS 7 is required.

About Siemens

Siemens is more than an automation vendor - we are an enabler of the new Digital Industry. Through our disruptive technologies and industry expertise, our solutions have helped companies like yours to achieve manufacturing that is faster, safer, and more flexible.

Authors of these application notes

Judy Wu, Engineering Team at Siemens

Lisandro De La Oliva Rojas, Lead Engineer at Siemens

About Top 20 Application Notes

The Top 20 Secure PLC Coding Practices are a community effort with best practices gathered from a large crowd of engineers from all kinds of different organizations. Thus, each single practice has been used by someone in the community.

However, there are many different kinds of PLCs and environments out there, for which the Top 20 as they are may or may not apply. The Top 20 Application Notes are case studies for specific PLCs, specific organizations (vendors, integrators, operators) and their workflows. People who have tried to apply the Top 20 take notes on their experiences – how they applied the practices, what worked, and what did not work. The aim is to gather application examples to help others, one use case at a time, and to eventually improve the Top 20's real-world applicability. Application notes issued by vendors and integrators are especially important since operators can use them as guidance for the PLCs they have in operation or consider buying.

Sharing your own Top 20 Application Note is easy. Just complete this template (feel free to modify as needed), send to plc-security@admeritia.de so we can publish on the Secure PLC project's website and social media channels and share widely with your clients, colleagues, prospects, network and across social media.

About the Top 20 Secure PLC Programming project

For many years, Programmable Logic Controllers (PLCs) have been insecure by design. Several years into customizing and applying best practices from IT gave rise to secure protocols, encrypted communications, network segmentation etc. However, to date, there has not been a focus on using the characteristic features in PLCs (or SCADA/DCS) for security, or how to program PLCs with security in mind. The Secure PLC Programming project – inspired by the existing Secure Coding Practices for IT – fills that gap.

Written for engineers by engineers: The aim of this project is to provide guidelines to engineers that are creating software (ladder logic, function charts etc.) to help improve the security posture of Industrial Control Systems.

These practices leverage natively available functionality in the PLC/DCS. Little to no additional software tools or hardware is needed to implement these practices. They can all be fit into the normal PLC programming and operating workflow. More than security expertise, good knowledge of the PLCs to be protected, their logic, and the underlying process is needed for implementing these practices. To fit the scope of the Top 20 Secure PLC Coding practices list, practices need to involve changes made directly to a PLC.

For more information, visit: plc-security.com