Top 20 Secure PLC Coding Practices Application Notes

Grantek

North American Pharmaceuctical Manufacturer Use Case

Use Case Introduction

Type of organization

Integrator

PLC make / model

Rockwell Automation / ControlLogix 5580

Scenario

Grantek, a leading system integrator and solution provider with a specialization in Smart Manufacturing solutions, Manufacturing Automation solutions, Industrial IT/Cybersecurity solutions and Manufacturing Consulting services, recently utilized the Top 20 PLC Coding Practices when delivering a solution for a North American pharmaceuctical manufacturer.

Utilizing the Top 20 PLC Coding Practices guide allowed Grantek to deliver a solution that not only improved productivity for the client, but also ensured a compliant and secure enviornment was maintainted at the facility during and after the successful implementation.

Application Statement

#	Practice Title	Applied? (yes / no)	Notes
1	**Modularize PLC Code**	Yes	Grantek's standard PLC structure has always included tasks, programs, and routines, as well as unit testing.
2	**Track operating modes**	Yes	Grantek's standard project specifications include communication alarms that would alarm when a PLC is put in program mode. However, in this project, the PLC is left in run (not remote) and alarms if the key is switched to remote.
3	**Leave operational logic in the PLC**	Yes	Alarms, timers, and enable for HMI buttons implemented in the PLC logic.
4	**Use PLC flags as integrity checks**	Yes	Counted error flags in fault routines for categories of errors.
5	**Use cryptographic and / or checksum integrity checks for PLC code**	Yes	Used checksum details provided in the PLC Top 20 guidance document.
6	**Validate timers and counters**	Yes	No counters in this project, but timer presets are validated in the PLC.
7	**Validate and alert for paired inputs / outputs**	Yes	Grantek's standard template specifications have always included alarms for valves with both open and close limit switches.
8	**Validate HMI input variables at the PLC level, not only at HMI**	Yes	Previously, we often did the limit checking in the HMI, but implemented parameter limit checking in the PLC in this project.
9	**Validate indirections**	Yes	All variables used as array indexes are checked for validity before use.
10	**Assign designated register blocks by function (read / write / validate)**	N/A	
11	**Instrument for plausibility checks**	N/A	
12	**Validate inputs based on physical plausibility**	Yes	Used raw input as well as scaled value of transmitters to stop some actions such as tank filling.
13	**Disable unneeded / unused communication ports and protocols**	N/A	
14	**Restrict third-party data interfaces**	N/A	
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.

#	Practice Title	Applied? (yes / no)	Notes
16	**Summarize PLC cycle times and trend them on the HMI**	Yes	Historized and displayed task scan times.
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 an alarm on each first scan.
19	**Monitor PLC memory usage and trend it on the HMI**	N/A	
20	**Trap false negatives and false positives for critical alerts**	N/A	

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

Application Details

To deliver a compliant solution for this leading biopharmaceutical client, Grantek developed an Ignition SCADA application and two PLC programs for a greenfield project. The platform supports WFI distribution, EMS and OEM systems monitoring, alarming and reporting. A version control system, multiple environments (Development, Test, and Production), and graduated and audited deployment infrastructure provides a DevOps solution. The OEM equipment consisted of multiple vendors and utilized a wide variety of processors and HMI systems that were integrated into the Ignition SCADA application. A connectivity platform was used to bridge communication to the SCADA and provide data for historization.

About Grantek

For over 40 years, top manufacturers in Life Sciences/Pharmaceuticals, Food & Beverage and CPG have called upon Grantek to solve their most complex business and manufacturing challenges. Grantek automates Pharmaceutical and Food & Beverage manufacturing operations, including integration with business systems for seamless solutions. Grantek helps customers meet the stringent requirements and challenges of the 4th Industrial Revolution. Grantek is a system integrator and solution provider with a specialization in Smart Manufacturing solutions, Manufacturing Automation solutions, Industrial IT/Cybersecurity solutions and Manufacturing Consulting services.

Authors of these application notes

Tennille Whiteford, Lead Life Sciences Systems Engineer at Grantek

Geoff Farrer, Marketing Manager at Grantek

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