

NISTIR 8188

Key Performance Indicators for Process Control System Cybersecurity Performance Analysis

CheeYee Tang

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.IR.8188>

NIST
**National Institute of
Standards and Technology**
U.S. Department of Commerce

NISTIR 8188

Key Performance Indicators for Process Control System Cybersecurity Performance Analysis

CheeYee Tang
*Intelligent Systems Division
Engineering Laboratory*

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.IR.8188>

August 2017



U.S. Department of Commerce
Wilbur L. Ross, Jr., Secretary

National Institute of Standards and Technology
Kent Rochford, Acting NIST Director and Under Secretary of Commerce for Standards and Technology

Contents

Contents	1
List of Figures	1
List of Tables	1
1. Introduction	2
2. Process Control System	3
2.1. The Chemical Process Model (Plant Simulator).....	4
2.2. Process Control System Operation	5
2.3. Network Architecture	5
2.4. Process and Data Flow.....	7
3. Key Performance Indicators	9
3.1. Definition.....	9
3.2. Selection Process	10
4. Measurement Methodologies	10
4.1. Manufacturing Process Performance	11
4.2. Network Performance	12
4.3. Computing Resources Performance.....	12
4.4. Industrial Protocol Performance	13
4.5. OPC Data Exchange Performance	14
5. Appendix A -- References	15
6. Appendix B – List of Software Tools	15

List of Figures

Figure 1 - View of the Process Control System	3
Figure 2 - Tennessee Eastman Process Model	4
Figure 3 - Network Architecture Diagram	7
Figure 4 - Diagram of the Human Machine Interface (HMI).....	8
Figure 5 - Process Dataflow Diagram	9

List of Tables

Table 1 – Manufacturing Process Performance KPI	11
Table 2 - Network Performance KPI.....	12
Table 3 - Computing Resources Performance KPI	13
Table 4 - Industrial Protocol Performance KPI	14
Table 5 - OPC Data Exchange Performance KPI.....	14

1. Introduction

The National Institute of Standards and Technology has constructed a testbed to measure the performance impact induced by cybersecurity technologies on Industrial Control Systems (ICS) [1]. The Cybersecurity for Smart Manufacturing Systems (CSMS) Testbed emulates real-world manufacturing processes and their industrial control systems by using software simulators and commercial control hardware in the laboratory environment.

This report focuses on the Process Control System of the Testbed, one of the four systems¹ within the CSMS Testbed.

The Process Control System emulates an industrial continuous manufacturing system, a manufacturing process to produce or process materials continuously, where the materials are moving, going through chemical reactions, or undergoing mechanical or thermal treatment continuously. Continuous manufacturing usually implies a 24x7 operation with infrequent maintenance shutdowns and is contrasted with batch manufacturing. Examples of continuous manufacturing systems include chemical production, oil refining, natural gas processing, and waste water treatment.

The Process Control System uses the Tennessee Eastman challenge problem [2], a real-world industrial chemical manufacturing process, as the simulation model for the chemical reaction. The system integrates the control algorithm developed by Ricker [3] to control the simulated chemical reaction. With the use of widely deployed industrial hardware like programmable logic controllers (PLCs) and industrial network switches as part of the control loop, this system emulates a complete setup of a continuous chemical manufacturing system.

This hardware-in-the-loop setup allows the testbed to measure the performance of the manufacturing system using real-world industrial hardware devices, while the chemical manufacturing process is simulated in software.

This system employs a modular design which allows a greater flexibility for re-configuration and scalability. The major sub-systems and components are largely independent and have well-defined interfaces, leading to a greater flexibility to change the individual sub-systems, like the continuous manufacturing process simulation, the network topologies, or the industrial communication protocol. This allows the researcher to investigate a wide range of different settings of industrial manufacturing systems, and to study the impacts of the implementation of cybersecurity technologies on the system performance and operation.

This report describes the Key Performance Indicators (KPI) and the measurement methodologies being used to measure the performance of the Process Control System.

¹ The other three systems are Collaborative Robotic System, Transportation System, and Additive Manufacturing System.

2. Process Control System

The Process Control System² includes a software simulator to emulate the Tennessee Eastman chemical reaction process. The simulator is written in C code and is executed on a Windows 7 based computer. In addition, the system includes a Programmable Logic Controller (PLC), a software controller implemented in MATLAB, a human-machine interface (HMI), an Object Linking and Embedding for Process Control (OPC) Data Access (DA) server, a Data Historian, an engineering workstation, and several virtual Local Area Network (LAN) switches and network routers. The Process Control System is housed in a 19-inch rack system as shown in Figure 1.

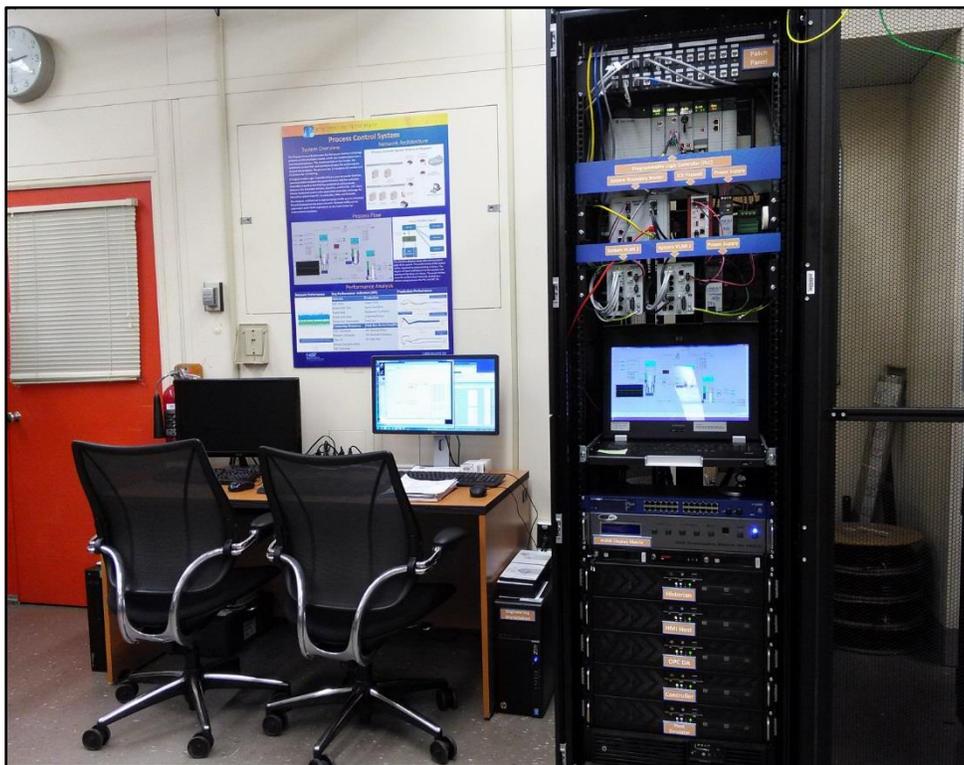


Figure 1 - View of the Process Control System

² Disclaimer: No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied. Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose. This publication was prepared by United States Government employees as part of their official duties and is, therefore, a work of the U.S. Government and not subject to copyright.

2.1. The Chemical Process Model (Plant Simulator)

The Process Control System uses the Tennessee Eastman challenge problem as the simulation model. The Tennessee Eastman problem, presented by Downs and Vogel [2], is a well-known process control problem in continuous chemical manufacturing. The model consists of five major units: a reactor, a product condenser, a vapor-liquid separator, a recycle compressor, and a product stripper for producing the desired end products. The model has 12 actuators for control and 41 sensors for monitoring.

The model is described in detail in [2], and the process description is summarized as follows: The continuous process produces two products from four input reactants. The process is irreversible and exothermic. The reaction rates of the reactants are a function of the reactor temperature. The gaseous reactants are combined in the reactor to form liquid products. The reactor temperature is cooled using an internal cooling bundle. The reactor product passes through the condenser to the separator. The vapor-liquid separator then separates unreacted gases from the liquid products. The unreacted gases are sent back to the reactor by the recycle compressor. The remaining reactants are removed in a stripping column. The two end products are sent downstream for further refining and separation. The process model is shown in Figure 2.

The Plant Simulator is implemented in C code, which was based on the Fortran code originally developed by Downs and Vogel.

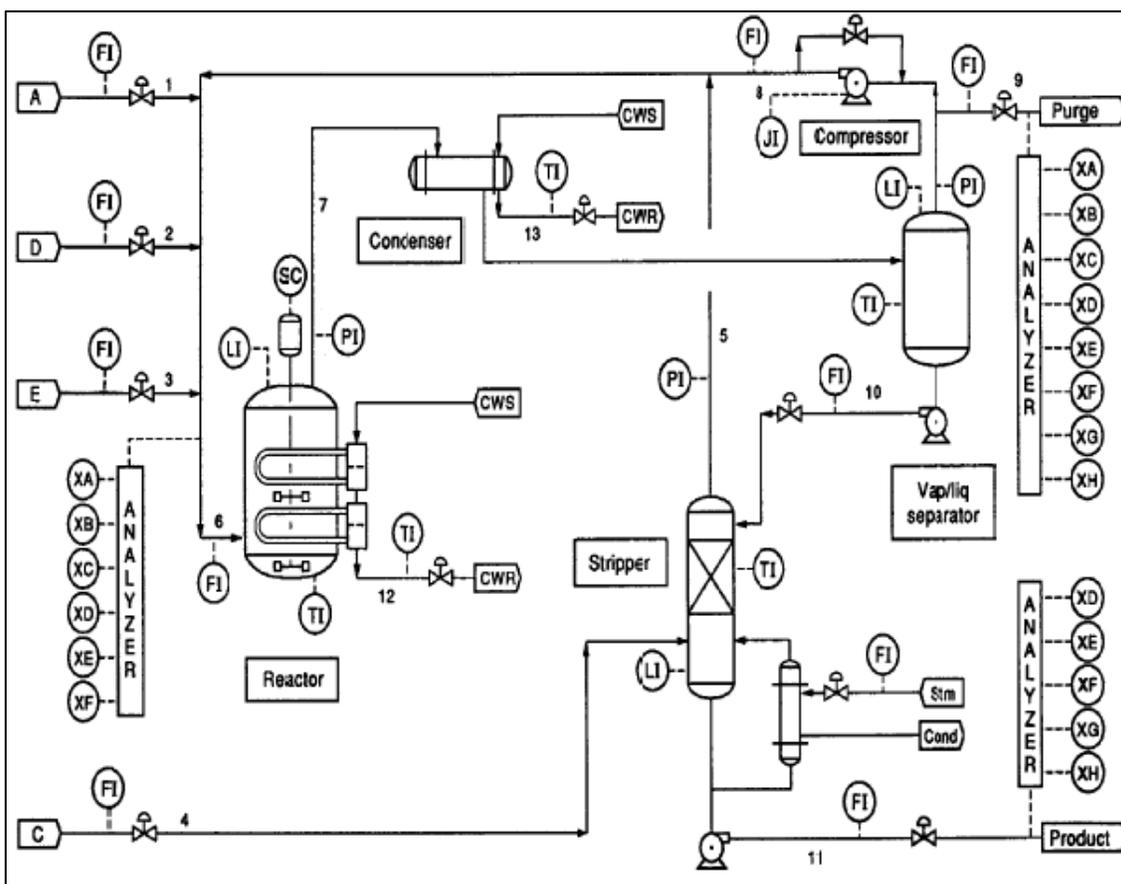


Figure 2 - Tennessee Eastman Process Model [2]

2.2. Process Control System Operation

The Plant Simulator requires a controller to provide a control loop in order to operate continuously. A decentralized controller implemented in Simulink, developed by Ricker [3], is used as the process controller. The Ricker implementation matches the Plant Simulator accurately, and the controller is a separate software process that runs on a separate computer from the Plant Simulator.

To provide communication between the Plant Simulator and the Controller, a hardware Programmable Logic Controller (PLC) with industrial network protocol capability is used. The industrial protocol is used to communicate between the Plant Simulator and the PLC. The Plant Simulator sends its sensor information to the Controller, and the Controller algorithm uses the sensor inputs to compute the desired values of the actuators and sends them back to the Plant Simulator.

In the Plant Simulator computer, a multi-node DeviceNet card was installed. DeviceNet is a common industrial protocol used in the automation industry to exchange data between control devices. The multi-node card allows a single hardware device to emulate multiple virtual DeviceNet nodes. In our case, each sensor and actuator point is a dedicated node. Therefore, 53 virtual nodes (41 for sensors and 12 for actuators) were configured in the system. A software interface was developed to send and receive sensor and actuator values between the Plant Simulator and the PLC through DeviceNet.

An OPC DA Server is running in a Windows 7 computer, acting as the main data gateway for the PLC. The PLC communicates to the OPC DA server to update and retrieve all the sensor and actuator information, respectively. This sensor and actuator information is also known as a “tag” in PLC terminology. The Controller has a MATLAB Simulink interface that communicates with the OPC DA server directly.

The communication data flow is described in more detail in Section 2.4.

A Human-Machine Interface (HMI) and a Data Historian are implemented in the system. The HMI provides a graphical user interface to present information to an operator or user about the state of the process. The HMI screen is shown in Figure 4. The Data Historian serves as the main database to record all the process sensor and actuator information. Both HMI and Data Historian have built-in interfaces to establish connections to the OPC DA to access all the process information.

An engineering workstation is used in the system for engineering support, such as PLC development and control, HMI development and deployment, and Data Historian data retrieval.

2.3. Network Architecture

The Process Control System network is segmented from the main Testbed network by a boundary router. The router is using a dynamic routing protocol, Open Shortest Path First (OSPF), to communicate with the main tested router. The network architecture is shown in Figure 3.

All network traffic needs to go through the boundary router to access the main testbed network.

There are two virtual network segments in the system. Each network is managed by an Ethernet switch. The HMI and the Controller are in virtual network VLAN-1, while the Plant Simulator, Data Historian, OPC DA Server, and PLC are in virtual network VLAN-2.

VLAN-1 simulates a central control room environment that the HMI and the controllers are virtually located in the same network segment. VLAN-2 simulates the process operation environment which typically consists of the operating plant, PLCs, OPC server, and the Data Historian.

These network switches and routers are highly reconfigurable, and therefore, allow the system to implement various network topologies for performance testing.

A Tofino Xenon security appliance, a firewall specially designed for industrial control system application, is installed to protect the PLC. The firewall rules are configured to only allow certain network nodes and specific protocols to access the PLC, and deny all other traffic.

All the computer nodes in the system have the Windows firewall enabled. Rules are configured to only allow traffic specific to their applications to access these computers. For example, the firewall of the OPC Server computer only allows a restricted range of Remote Procedure Call (RPC) and Distributed Component Object Model (DCOM) ports for the OPC clients to access, and restricts the source Internet Protocol (IP) address of the OPC clients.

The main Testbed network provides a Network Time Protocol (NTP) server. All systems in the Process Control System are synchronized with the Testbed NTP server.

A network packet analyzer tool is installed in all the computers in the system to capture and analyze network packets. Other specialized software tools are also used to monitor the system. For example, an OPC data analyzer is used to monitor OPC data exchange, and DeviceNet logging is used to log DeviceNet level traffic.

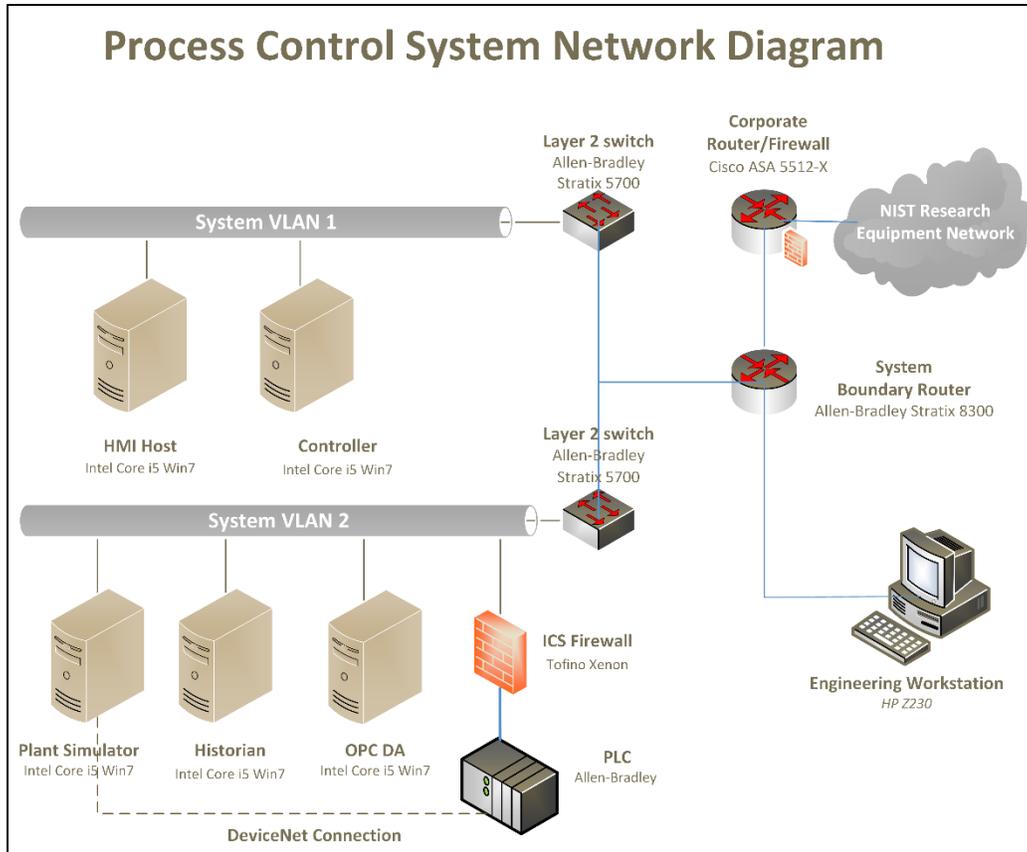


Figure 3 - Network Architecture Diagram

2.4. Process and Data Flow

Figure 4 shows the HMI view of the Process Control System. The HMI displays the process information in a graphical interface to allow the operator to easily monitor. The HMI can also alert the operator. For example, if the reactor pressure is above the predefined threshold, the reactor icon will flash red.

The Process Control System is initialized in four stages: Stage one is the initialization of application software. In this stage, the OPC DA server, HMI, Data Historian, and the MATLAB Controller are executed and initialized. Stage two is the initialization of the hardware devices. In this stage, the configuration file of the multi-node DeviceNet card and the PLC are loaded and the devices enter run mode waiting for data input. Stage three is the industrial network activation stage. In this stage, the Plant Simulator is activated and all the sensor and actuator nodes are connected to the PLC. The last stage is the simulation start stage. In this stage, the Plant Simulator and Controller are started and all the sub-systems enter run mode.

Once in the run mode, the Plant Simulator will start sending and receiving sensor and actuator values to and from the PLC. The OPC server polls the PLC at a data rate of 10Hz to exchange sensor and actuator values. The OPC server will send updated sensor values to the subscribing clients, namely, the Controller, HMI, and Data Historian. The OPC server operates in exception-report mode, meaning it will only send updates to clients when the data changes. When the

Controller receives sensor values, it runs through the control algorithm and computes the desired actuator values and sends them back to the OPC server. The OPC server will send the updated actuator values to the PLC during the next poll cycle. The DeviceNet interface will poll the PLC periodically for the latest actuator values and sends them to the Plant Simulator.

The dataflow diagram in Figure 5 shows the communication path of the system. The Plant Simulator communicates with the PLC through DeviceNet. The PLC communicates with the OPC server using EtherNet/IP as the application layer protocol, which is encapsulated in the Transmission Control Protocol/Internet Protocol (TCP/IP) packet.

The Controller, HMI, and Data Historian use the Distributed Component Object Model (DCOM) interface of the Remote Procedure Call (RPC) protocol to communicate with the OPC server to access information.

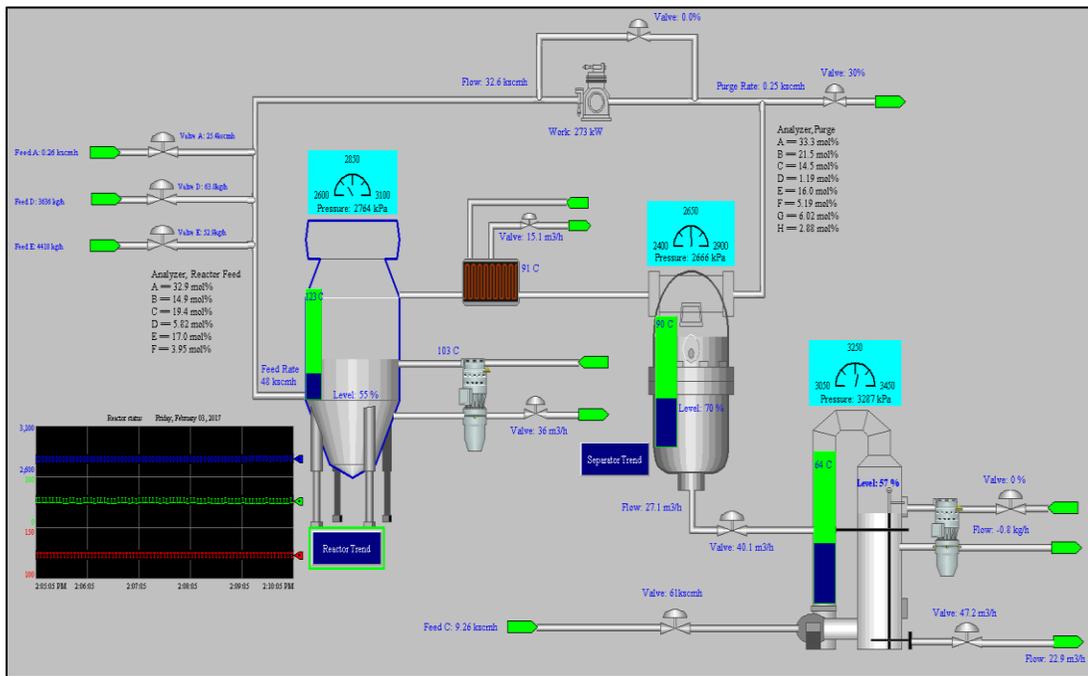


Figure 4 - Diagram of the Human Machine Interface (HMI)

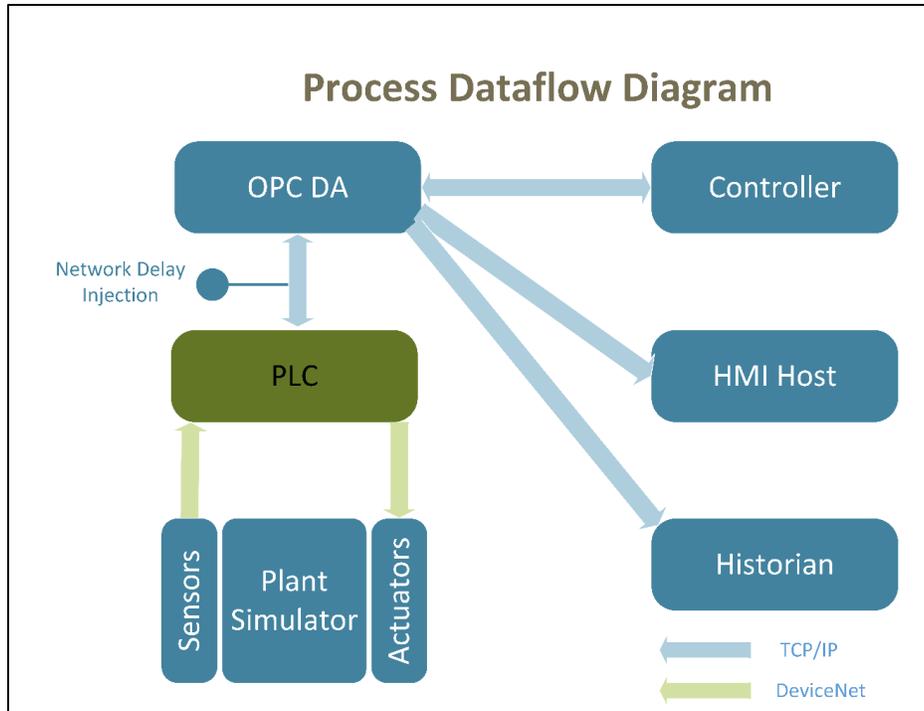


Figure 5 - Process Dataflow Diagram

3. Key Performance Indicators

The Key Performance Indicators (KPI) of the Process Control System are used to measure the performance of the system and its sub-systems. These KPIs are calculated in the baseline measurement, in which no additional cybersecurity measures are implemented, to form the basis of the performance of the system. In subsequent experiment runs, different cybersecurity measures are implemented and the KPIs are calculated again and compared to the baseline measurement. Analysis can be performed to understand the performance impact caused by the implementation of the cybersecurity measures.

3.1. Definition

A KPI is defined as a “computable performance assessment, as derived from a combination of metrics.” [4] In other words, a KPI is a computed value based on post-analysis and may utilize one or many primitive values to perform the computation.

For example, the Network Utilization KPI in the Industrial Protocol Performance section is a value computed by using primitive information like the message data size, the message overhead size, the data rate, and the message timestamps. These primitives provide a way to compute the total size of the messages on the network in a specified time. The Network Utilization is then computed by dividing it by the maximum capability of the network.

Many popular KPIs are readily available from the measurement tools. For example, the TCP Packet Round Trip Time is a KPI computed by using the timestamps information of a transmitted packet

and the corresponding received acknowledgment packet. The TCP Round Trip Time can be easily calculated in the network analyzer tool.

3.2. Selection Process

The KPIs are selected based on an examination of the Process Control System to determine which KPIs can be obtained. Literature review for industrial systems, specifically process control systems, was performed to select the KPIs that are most common and representative in the process control industry and could be used to best analyze the process performance.

The KPIs selected are divided into five key performance areas:

1. Manufacturing process performance: measures the performance of the continuous manufacturing.
2. Network performance: measures the performance of the underlying TCP/IP network.
3. Computing resources performance: measures the performance of the computers, hardware, and software processes.
4. Industrial protocol performance: measures the performance of the DeviceNet, an industrial communication protocol.
5. OPC Data Exchange performance: measures the performance of the data exchange mechanism of the system.

Measurements in different areas provide insight of the entire system performance from different perspectives. The manufacturing process performance provides indicators on how well the high-level manufacturing process and overall system perform. However, this may not be able to provide enough detail on the performance of the sub-systems, therefore measurements are also performed at sub-system levels.

For example, typical chemical continuous manufacturing is a relatively slow process in comparison with computer networking. Therefore, a moderate TCP/IP network delay may not reflect in the measurement of the high-level manufacturing process performance. However, such TCP/IP delay may have significant impact on the sub-systems. The effects will not be reflected in the high-level measurement until significant delays are accumulated in sub-systems. Measurements in multiple levels provide details and in-depth understanding to key performance areas of the entire system. It helps to understand how the aggregate effects will impact the performance. Aggregate effects will be important to the high-level manufacturing performance. For example, if a single ICS firewall adds 1 millisecond delay to the PLC operation, multiple firewalls in series could aggregate the delay, which could have a more significant impact on the manufacturing process performance.

4. Measurement Methodologies

The following sections describe the measurement methodology used for each sub-system to collect the KPIs.

4.1. Manufacturing Process Performance

Manufacturing process performance measures the performance of the continuous process. Once the process is started, the manufacturing system tends to run continuously. In the Process Control system, a typical experiment lasts from 24 hours to 48 hours. The KPIs are captured over the entire experiment period. Since the Plant Simulator is a software simulation, the KPIs are captured through software applications. The software logger records all the Plant Simulation 41 sensors and 12 actuators values in each simulation cycle. The Data Historian software records the same sensor and actuator values as published from the OPC server. The Data Historian only records when the data changes. A post processing is performed to compute the desired KPIs.

Table 1 – Manufacturing Process Performance KPI

Key Performance Indicator	Description
1.1 Startup Time	Measures the time to produce the end products from the time input materials are released into the process at the initial startup of the process.
1.2 Throughput Rate	Measures the amount of end products produced over a specific amount of time.
1.3 Input Feed Rate	Measures the amount of input materials consumed over a specific amount of time.
1.4 Output Yield	Measures the percentage of the end product produced at the output of the process.
1.5 Equipment Availability	Measures the availability of the critical equipment. The critical equipment in the Process Control System is the Reactor, Separator, and the Product Stripper.
1.6 Reactor Pressure	Measures the Reactor pressure.
1.7 Reactor Level	Measures the Reactor level.
1.8 Reactor Temperature	Measures the Reactor temperature.
1.9 Separator Level	Measures the Separator level.
1.10 Stripper Level	Measures the Stripper level.
1.11 Duration of Continuous Operation	Measures the time when the production starts to the next unplanned stop in production.
1.12 Number of Unplanned Stops	Measures the number of times the production stops unexpectedly, including emergency shutdown.
1.13 Production Effectiveness	Measures the percentage of time the planned production target is achieved over a specific amount of time.
1.14 Capacity Utilization	Measures the percentage of production capacity utilized over a specific amount of time.
1.15 Unit Cost	Measures the production cost of the end products.

4.2. Network Performance

Network performance measures the performance of the underlying TCP/IP network. In the Process Control system, the network is a critical communication link for all sub-systems. All the nodes are connected and communicate through the network connection. As a critical communication channel, any network delay or failure will impact the performance of the entire continuous production system. For example, the Controller relies on real-time sensor information to compute the desired actuator values to maintain stable control. Also the HMI relies on real-time data to display the up-to-date process information for operators. Each computer has a network packet analyzer tool installed to capture all the inbound and outbound network traffic of that computer. Post-processing is performed to compute the desired KPIs.

Table 2 - Network Performance KPI

Key Performance Indicator	Description
2.1 Packet Path Delay	Measures the time delay along the path from transmitter to receiver.
2.2 Inter-packet Delay	Measures the difference between the packet path delay of two packets.
2.3 TCP Packet Round Trip Time	Measures the amount of time for the source node to receive the acknowledgement of receipt (ACK) from the destination node.
2.4 Information Ratio	Measures the percentage of the quantity of process information packets over all packets transmitted.
2.5 Bit Rate	Measures the rate of bits transmitted or received over a specific amount of time.
2.6 Packet Rate	Measures the rate of packets transmitted and received over a specific amount of time
2.7 Packet Error Rate	Measures the rate of packets received with errors over a specific amount of time
2.8 Proportion of Protocol Type	Measures the percentage of a unique packet protocol type observed over a specific amount of time.
2.9 Network Utilization	Measures the percentage of network capacity utilized over a specific amount of time.
2.10 Packet Size	Measures the number of bytes contained in a packet.
2.11 Packet Loss Rate	Measures the percentage of packets that failed to reach the destination node over a specific amount of time.

4.3. Computing Resources Performance

Computing resources performance measures the performance of the computing hardware and software. This is especially important for sub-systems that are software intensive, such as the HMI, Data Historian, and the OPC Server. Major functionalities in these sub-systems are implemented in software. For example, the HMI is a software application powered by the Rockwell *FactoryTalk*

software suite. Software applications consume computing resources to execute. Therefore, the availability of computing resources, such as processor time, memory, disk usage, and network access, directly affect the performance of these software applications. A lack of computing resources will delay the execution of the HMI software, which in turn delays the display of the manufacturing process information.

To measure the performance of the computing resources, the Microsoft Resource Monitor and the Microsoft Performance Monitor are used. These tools are included in the Windows 7 installation and have access to many computing resources and Windows operating system statistics. Another Microsoft tool, TCPView, is used to capture network usage and TCP connections per software application.

Since each computer has a multi-core processor, the software application being measured is pinned to run in a specific core that has the lightest load. To perform the measurement, a data collection set is first defined in the tools describing which KPI and the specific processor core to record.

Table 3 - Computing Resources Performance KPI

Key Performance Indicator	Description
3.1 CPU Utilization	Measures the percentage of central processing unit (CPU) utilized over a specific amount of time.
3.2 Memory Utilization	Measures the percentage of memory utilized over a specific amount of time.
3.3 Disk I/O	Measures the mean rate and standard deviation of read and write operations to the server hard drive per software application.
3.4 Network Throughput	Measures the mean rate and standard deviation of packets transmitted and received per software application.

4.4. Industrial Protocol Performance

Industrial protocol performance measures the performance of the industrial communication network. The industrial protocol being used in the Process Control System is DeviceNet, which handles the communication between the Plant Simulator and the PLC. It is based on the Controller Area Network (CAN) protocol, a serial message-based communication protocol, with an additional application and physical layer specification.

In understanding the performance of the DeviceNet, it helps to better understand the low-level details of the industrial control network, and this information can help to identify the performance impact on the system.

To measure the performance of the industrial protocol, logging capacity is added to the software DeviceNet interface. The interface timestamps and captures all inbound and outbound DeviceNet traffic. A post processing is performed to compute the desired KPIs.

Table 4 - Industrial Protocol Performance KPI

Key Performance Indicator	Description
4.1 Network Utilization	Measures the percentage of network capacity utilized over a specific amount of time.
4.2 Information Ratio	Measures the percentage of the quantity of process information packets over all packets transmitted.
4.3 Packet Path Delay	Measures the time delay along the path from transmitter to receiver.
4.4 Packet Rate	Measures the rate of packets transmitted and received over a specific amount of time
4.5 Data Size	Measures the number of application payload data in bytes in a packet.
4.6 Packet Size	Measures the number of bytes contained in a DeviceNet packet.
4.7 DeviceNet Throughput	Measures the mean rate and standard deviation of packets transmitted and received on DeviceNet.

4.5. OPC Data Exchange Performance

OPC data exchange performance measures the performance of the data exchange process between the PLC and the OPC server, and between the OPC server and the subscribing clients, such as HMI and Data Historian. As the main data gateway, the OPC server is an important sub-system that serves sensor and actuator data to the rest of the system. Any processing delay or failure will impact the performance of the entire system. For example, the Controller relies on real-time sensor information to compute the desired actuator values to maintain a stable control. Also the HMI relies on real-time data to display the up-to-date process information for operators.

OPC data is sent through the TCP/IP network connection, either encapsulated in EtherNet/IP format (between PLC and OPC DA) or in Windows Remote Procedure Call (RPC/DCOM) format (between clients and OPC DA). The OPC packet can be captured using a network packet analyzer tool. However, a decoding program is required to extract the OPC information from the EtherNet/IP or the RPC/DCOM packets to analyze and compute the KPIs for the performance measurement.

A control counter is added to the DeviceNet interface and the PLC. This control counter increments every simulation cycle of the Plant Simulator and in turn updates the PLC through the DeviceNet. By measuring the round trip time of the counter, it serves as a control value for network latency between the Plant Simulator and the Controller.

Table 5 - OPC Data Exchange Performance KPI

Key Performance Indicator	Description
5.1 OPC DA Latency (PLC to OPC)	Measures the amount of time for the information to be available on the OPC server after the information is sent from the PLC.

5.2 OPC DA Latency (OPC to PLC)	Measures the amount of time for the information to be available on the PLC after the information is sent from the OPC client.
---------------------------------	---

5. Appendix A -- References

[1] R. Candell, T. Zimmerman and K. Stouffer, "An Industrial Control System Cybersecurity Performance Testbed, NISTIR 8089," National Institute of Standards and Technology (NIST), Gaithersburg, MD, 2015.

[2] J. J. Downs and E. F. Vogel, "A Plant-Wide Industrial Process Control Problem," *Computers and Chemical Engineering*, vol. 17, no. 3, pp. 245-255, 1993.

[3] L. Ricker, "Decentralized control of the Tennessee Eastman Challenge Process," *Journal of Process Control*, vol. 6, no. 4, pp. 205-221, 1996.

[4] T. Zimmerman, "Metrics and KPI for Robotic Cybersecurity Performance Analysis, NISTIR 8177," National Institute of Standards and Technology (NIST), Gaithersburg, MD, 2016.

[5] I. C. Garretson, M. Mani, S. Leong, K. W. Lyons and K. R. Haapala, "Terminology to support manufacturing process characterization and assessment for sustainable production," *Journal of Cleaner Production*, vol. 139, p. 986–1000, 2016.

[6] F.-L. Lian, J. R. Moyne and D. M. Tilbury, "Performance Evaluation of Control Networks: Ethernet, ControlNet, and DeviceNet," *IEEE Control Systems Magazine*, February 2001.

[7] G. Buckbee, "Realtime Metrics for Process Performance," *Chemical Engineering*, June 2007.

[8] M. Newton, "KPIs for Industrial Automation and Process Control," Opto22, Temecula, CA, 2017.

[9] M. Davidson, "28 Manufacturing Metrics that Actually Matter," LNS Research, Cambridge, MA, 2013.

6. Appendix B – List of Software Tools

Software Tool Name	Description
Wireshark	An open source packet analyzer. https://www.wireshark.org/
Microsoft Resources Monitor	A Windows utility to provide hardware and software resources usage information.
Performance Monitor	A Windows system monitoring program that monitors various activities on the computer.
Process Explorer	A Windows program that collects information about software processes running on the system. https://technet.microsoft.com/en-us/sysinternals/processexplorer.aspx
NetBalancer	An Internet traffic control and monitoring tool. https://netbalancer.com

TCPView	A Windows program that shows detailed listings of all Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) endpoints. https://technet.microsoft.com/en-us/sysinternals/tcpview.aspx
Netstat	A tool to display the networking details of a system.