612.2

System of Systems

SANS | GIAC CERTIFICATIONS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

All reference links are operational in the browser-based delivery of the electronic workbook.

ICS612.2 | ICS Cybersecurity In-Depth

SANS

System of Systems

SANS ICS612: This course is focused on the implementation and support of a secure control system environment through a hands-on, in-depth course that is designed to change how students engineer and support ICS environments.

Jeffrey Shearer
Mr. Shearer is a member of the SANS Institute ICS team focused on developing courseware in support of the ICS curriculum. Jeffrey also acted as a Subject Matter Expert (SME) for the Global Industrial Cyber Security Professional (GICSP) certification and is a content contributor for ICS NetWars. He also participates as an advisory board member for the ICS Security Summit and Training events.

Prior to joining SANS Institute, Mr. Shearer worked at Rockwell Automation for 23 years, where his most recent role was a Senior Security Architect for Rockwell Automation's Commercial Engineering group focused on network and security designs for Industrial Automation Control Systems (IACS) and Industrial Demilitarized Zones (IDMZ). Mr. Shearer was a contributing member of the Rockwell Automation and Cisco Systems Converged Plantwide Ethernet (CPwE) team, where he participated in architecture design and validation efforts. He also co-authored publications such as Deploying Industrial Firewalls within a Converged Plantwide Ethernet Architecture, Site-to-Site VPN to a Converged Plantwide Ethernet Architecture, and Securely Traversing IACS Data across the Industrial Demilitarized Zone.

Jason Dely
Jason Dely is responsible for leading the critical infrastructure and industrial control systems (ICS) security practice for Cylance. Prior to joining Cylance, Jason held many roles at Rockwell Automation, where he assisted clients with their research, design, integration, testing, and response activities across a variety of application, security, and infrastructure initiatives. During this time, Jason gained in-depth ICS product, protocol, and operational experiences that are invaluable when it comes to evaluating and building defenses within critical infrastructure organizations. His security passion over the past 18 years of experience with ICS is founded upon balancing business requirements against people, process, and technologies unique to each organization to ensure their operations are safe, reliable, and secure.

Tim Conway
Tim Conway is currently the Technical Director – ICS and SCADA programs at SANS, and is responsible for developing, reviewing, and implementing technical components of the SANS ICS and SCADA product offerings. Additionally, he performs contract and consulting work in the areas of ICS cybersecurity with a focus on energy environments. Recognizing the need for ICS-focused cybersecurity training throughout critical infrastructure environments and an increased need for hands-on training, Tim assisted in authoring, and instructs, the ICS curriculum's newest courses and ICS NetWars challenges. Outside of SANS, Tim continues to work on projects that blend cybersecurity, operations technology, and critical infrastructure protection with a focus on the energy sector.

Chris Robinson
Chris Robinson graduated from the United States Naval Academy with a B.S. in Computer Science and served over 6 years in the United States Navy. He then began his IT security career as a consultant for Booz Allen Hamilton before he attended graduate school full time at San Diego State University, earning an M.S. in Computer Science. Following graduation, Chris worked as Computer Scientist for the Navy and was an Adjunct Professor at San Diego's Mesa Community College. Chris then transitioned into ICS security, where he is currently an ICS Principal Consultant at Cylance, applying his expertise to various ICS cybersecurity projects to ensure solutions meet the needs of a modern industrial control system. Chris has learned firsthand the unique requirements and operational constraints for securing ICS environments. Chris currently holds and maintains multiple certifications, including the CISSP, OSCP, GICSP, GISP, GISF, and CEH. Chris teaches both the SANS MGT414 and MGT415 courses and currently resides in London, UK.

Contributor
Ted Gutierrez
Ted Gutierrez, CISSP, GICSP, and GCIH, is the ICS & NERC CIP Product Manager at the SANS Institute. Mr. Gutierrez was most recently the Director of Operations Technology & NERC Compliance at Northern Indiana Public Service Company (NIPSCO), where he was responsible for compliance to NERC 693 and CIP Standards and the support of the related operations technology systems. Mr. Gutierrez has more than 25 years of experience working in the electric utility, information technology, and manufacturing industries.

ICS612 Course Outline

- Section 1: The Local Process
- Section 2: System of Systems
- Section 3: ICS Network Infrastructure
- Section 4: ICS System Management
- Section 5: Covfefe Down!

Throughout this section, we will be interconnecting the various local systems that we built in Section 1, with an overall objective of connecting the various independent systems into a larger interdependent system of systems.

ICS612 Section 2 Outline (1)

- Head End Process Overview
- Lab 2.1: Connect Pods to Level 3 Infrastructure
- Lab 2.2: Remote I/O
- Lab 2.3: Validate Functionality
- ICS Secure Architecture
- Lab 2.4: Network Infrastructure Configuration
- Process Communication and Data Flow Mapping
- Lab 2.5: Map Communications for the Environment
- Lab 2.6: Configure Connections to Process Visualization
- Local Attacks and Process Manipulation
- Lab 2.7: PLC Device-Level Attack
- Lab 2.8: OPC Discovery Attack
- Lab 2.9: Local Network MITM Attack

This page intentionally left blank.

ICS612 Section 2 Outline (2)

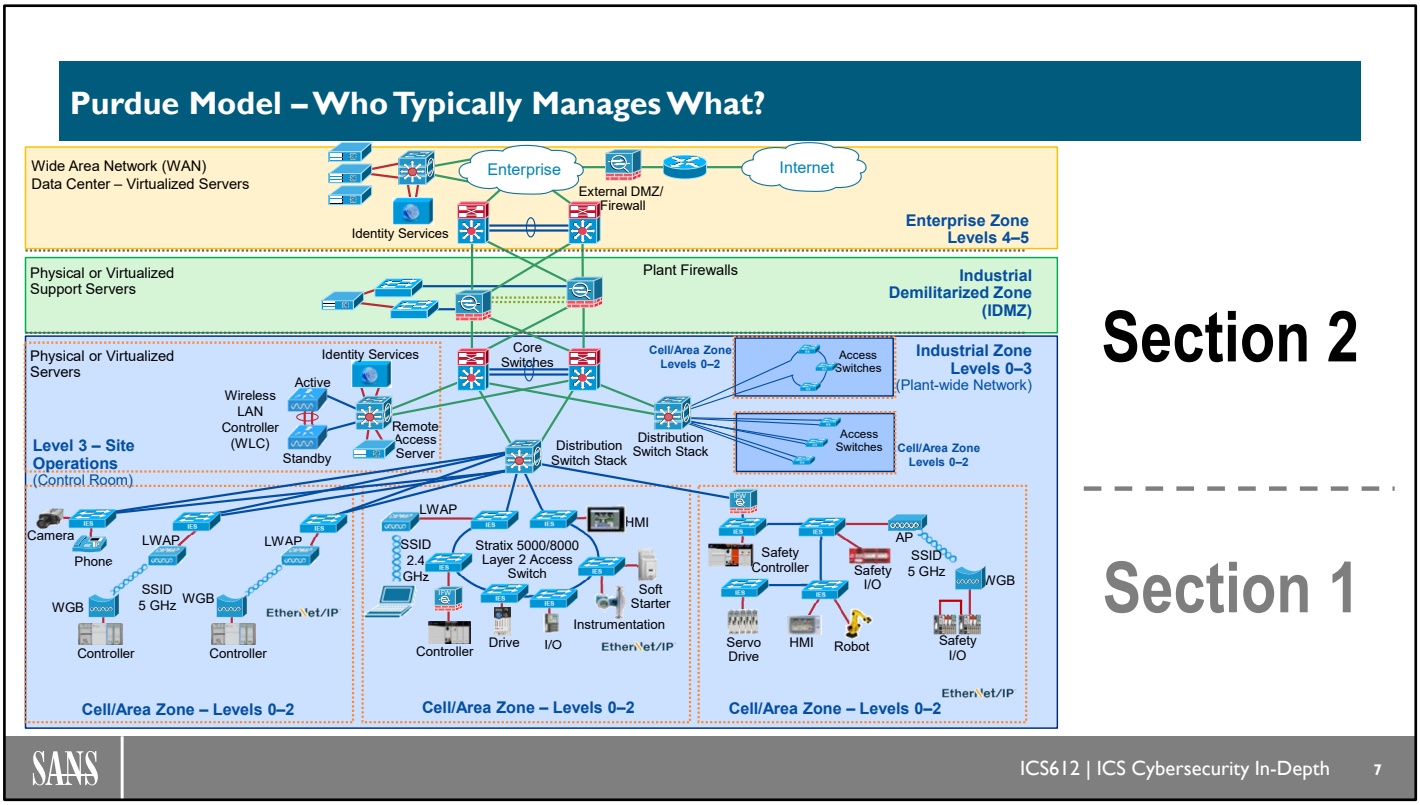
- Head End Process Overview
- Lab 2.1: Connect Pods to Level 3 Infrastructure
- Lab 2.2: Remote I/O
- Lab 2.3: Validate Functionality
- ICS Secure Architecture
- Lab 2.4: Network Infrastructure Configuration
- Process Communication and Data Flow Mapping
- Lab 2.5: Map Communications for the Environment
- Lab 2.6: Configure Connections to Process Visualization
- Local Attacks and Process Manipulation
- Lab 2.7: PLC Device-Level Attack
- Lab 2.8: OPC Discovery Attack
- Lab 2.9: Local Network MITM Attack

This page intentionally left blank.

Head End Process Overview

Hardware Components
Infrastructure Components
Applications

This page intentionally left blank.



As directional map in Section 1, we focused on the lower levels of the Purdue Model and in Section 2, we will continue to add elements to our environment and expand into Level 3 and beyond.

Enterprise Zone – Level 5: Enterprise
Level 5 is where the centralized IT systems and functions exist. Enterprise resource management, business-to-business, and business-to-customer services typically reside at this level. Often, the external partner or guest access systems exist here, although it is not uncommon to find them in lower levels (e.g., Level 3) of the framework to gain flexibility that may be difficult to achieve at the enterprise level. However, this approach may lead to significant security risks if not implemented within IT security policy and approach.

Enterprise Zone – Level 4: Site Business Planning and Logistics
Level 4 is where the functions and systems that need standard access to services provided by the enterprise network reside. This level is viewed as an extension of the enterprise network. The basic business administration tasks are performed here and rely on standard IT services. These functions and systems include wired and wireless access to enterprise network services such as the following:

Access to the Internet, access to Email (hosted in data centers)

- Non-critical plant systems such as manufacturing execution systems and overall plant reporting, such as inventory, performance, etc.
- Access to enterprise applications such as SAP and Oracle (hosted in data centers)

Manufacturing Zone – Level 3: Site Manufacturing Operations and Control
Level 3, the site level, represents the highest level of the IACS. The systems and applications that exist at this level manage plantwide IACS functions. Levels 0 through 3 are considered critical to site operations. The applications and functions that exist at this level include the following: Devices found in Level 3 are often responsible for managing control plant operations to produce the desired end product. Applications, services, and systems that are found at this level include:

- Level 3 IACS network
- Reporting (For example: Cycle times, quality index, predictive maintenance)
- Plant historian
- Detailed production scheduling
- Site-level operations management
- Asset and material management
- Control room workstations
- Patch launch server
- File server
- Other domain services, e.g. Active Directory (AD), Dynamic Host Configuration Protocol (DHCP), Dynamic Naming Services (DNS), Windows Internet Naming Service (WINS), Network Time Protocol (NTP), etc.
- Terminal server for remote access support
- Staging area
- Administration and control applications

The systems and applications in Level 3 communicate with the systems in the Enterprise zone through an Industrial DMZ. Direct communication between systems in Manufacturing and Enterprise zones is discouraged. Additionally, systems in Level 3 may communicate with systems in Levels 1 and 0.

Cell/Area Zone – Level 2: Area Supervisory Control
Level 2 represents the applications and functions associated with the Cell/Area zone runtime supervision and operation. These include the following:

- Operator interfaces or Human Machine Interfaces (HMI)s
- WGB – Work Group Bridges
- LWAP – Lightweight access points
- Alarms or alerting systems
- Control room workstations.

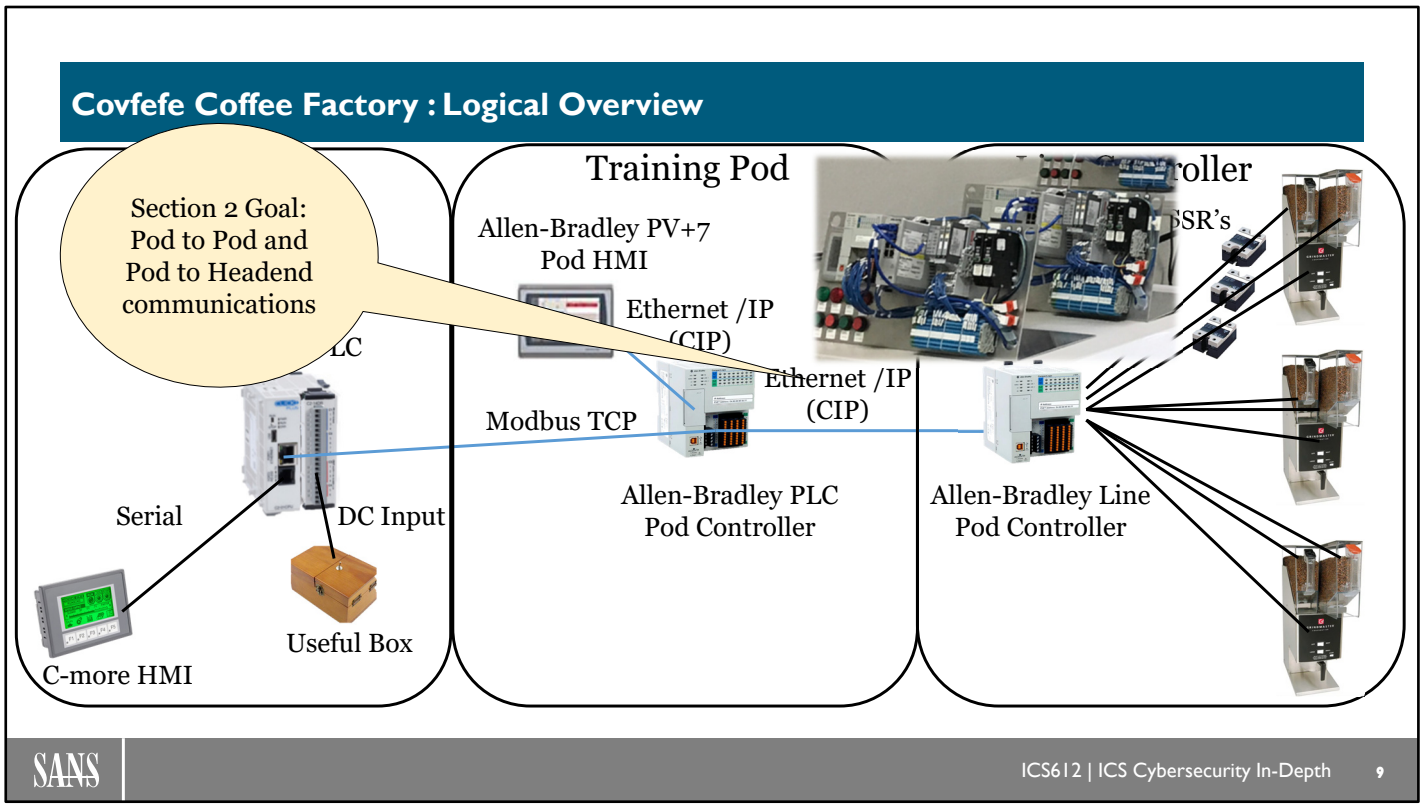
Depending on the size or structure of a plant, these functions may exist at the site level (Level 3).

Cell/Area Zone – Level 1: Basic Control
Level 1 consists of controllers that direct and manipulate the manufacturing process, the key function of which is to interface with the Level 0 devices (e.g., I/O, sensors, and actuators). Historically in discrete manufacturing, the controller is typically a programmable logic controller (PLC). In process manufacturing, the controller is referred to as a distributed control system (DCS). The terms controller or programmable automation controller (PAC) refer to the multidiscipline controllers used across manufacturing disciplines. These include discrete, continuous process, batch, drive, motion, and safety controllers.

Cell/Area Zone – Level 0: Process
Level 0 consists of a wide variety of sensors and actuators involved in the basic manufacturing process. These devices perform the basic functions of the IACS, such as driving a motor, measuring variables, setting an output, and performing key functions such as painting, welding, bending, and so on.

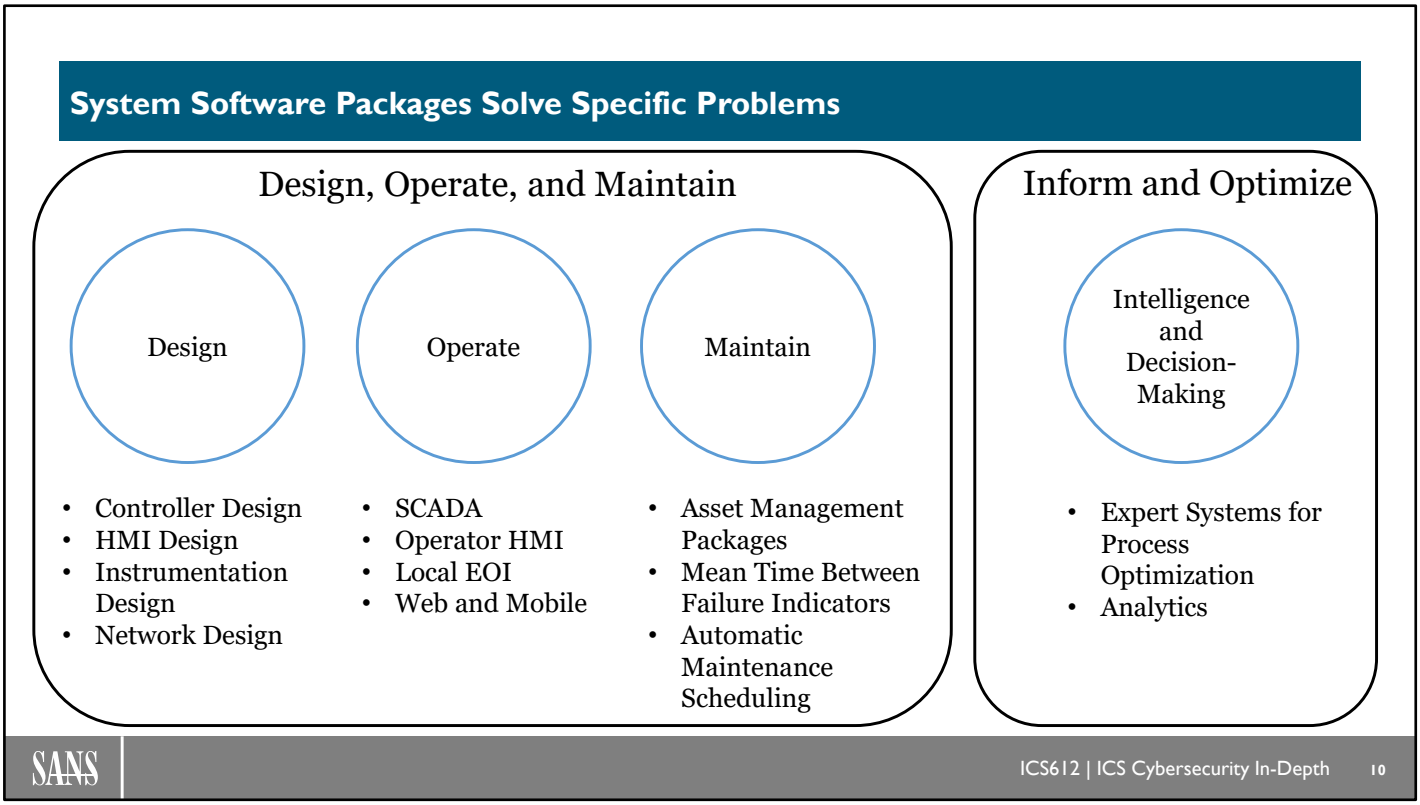
Reference:

- Cisco and Rockwell Automation (2011). *Converged Plantwide Ethernet (CPwE) Design and Implementation Guide*. Cisco Systems, Inc. (n.d.). Retrieved from https://literature.rockwellautomation.com/idc/groups/literature/documents/td/enet-td001_-en-p.pdf



As we look at Levels 0 and 1 hands-on exercises, we will use two systems to achieve the PLC and HMI learning objectives. In the middle of the slide, you will see the training Pod hardware that consists of an Allen-Bradley PanelView HMI and the Allen-Bradley (A-B) CompactLogix PLC. The training Pod also contains push buttons, indicator lights and remote breakers that the A-B PLC will use for input and output control.

The student kit as shown on the left contains the Click Plus PLC and the C-more HMI that will be used during student labs. The Click Plus PLC will communicate with the A-B PLC via Modbus TCP sharing data register information and I/O status. The student kit also contains a Useless Box that will be transformed into a Useful Box that will be controlled by the Click Plus PLC in order to show you how "useful" a simple input switch, motor circuit, and power source can be to gain knowledge about PLC systems. The student kit also contains a K-type thermocouple to demonstrate analog input capabilities of the Click Plus PLC.



ICS and SCADA vendors are in the business of helping automation customers solve their processing problems. The software packages that have evolved are intended to allow a customer to design their control strategy, visualize their process, and maintain the assets so they can continue to make products in a consistent and predictable manner.

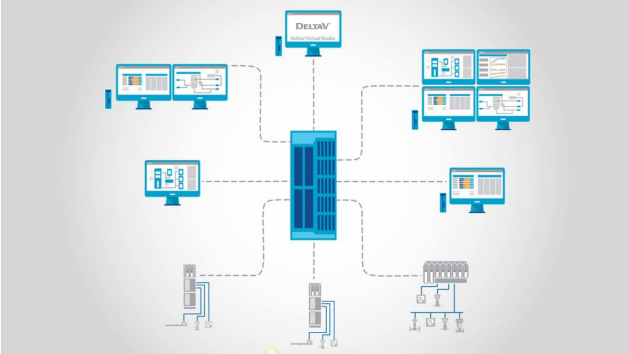
In our modern age of data connectivity, the expectation of information connectivity from our plant floors to our board rooms is the bar which everyone expects to achieve. Because this type of connectivity has been realized, automation vendors have also emerged with improved analytics to help everyone from the plant floor supervisor to the plant manager make better decisions.

In this slide, we want to stress that the automation vendor offerings are brought to market with the sole purpose of helping the customer make their product in a more consistent manner. Network connectivity has been a disruptive technology that enabled customers to instrument processes and allowed automation vendors to provide better analytics and visualization software offerings.

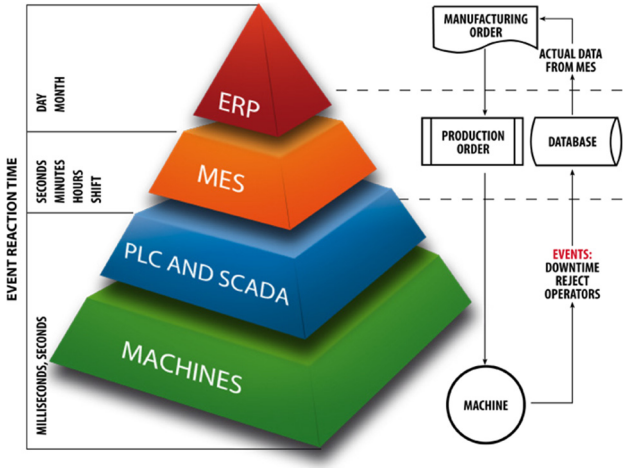
Common Production and Manufacturing Types

Hybrid

Process



Discrete



All industrial organizations and sectors can be categorized into one of three production and manufacturing types. The three types are Process, Discrete and Hybrid. The most distinct difference is the type of "product" produced and the method required to produce it. Another differentiator is the length of time spent to perform a specific production function. The hierarchical level of and the associated interactions with the organization's business systems can be used to determine the production and manufacturing type.

Process manufacturing typically requires a constant monitoring and correction to maintain physical variabilities such as temperature, pressure, and flow. This manufacturing type runs mostly autonomously and typically does not require continuous instructions from other business systems to function properly. Pushing quality assurance, regulation data, and even optimization data out to other systems is common. Other common descriptions for this type include "continuous process" and "batch process." A continuous process is one that continuously moves and/or produces a product as it flows through the mechanical system with minimal product changes. Examples of continuous processes include power generation, pipelines, refineries, and steel blast furnaces.

A batch process is one where a product is produced in a distinct vessel following a distinct but changeable recipe. Batch processes are typical of breweries or pharmaceutical manufacturing.

Discrete manufacturing typically requires a steady interaction with a business system, such as Enterprise Resource and Planning (ERP), to receive a part order on what to produce or assemble. Some level of autonomy may exist at the machine level with use of automation and robots, but many manufacturing facilities still rely on people operating equipment at stations within the assembly line. Even with a queue of instructions at the PLC level, discrete manufacturing typically can only operate for a short period without interaction with upper business systems. As shown in the diagram above, the Enterprise Resource and Planning (ERP) system in corporate IT is the primary repository of the main business process. ERPs are typically behemoths and are unable to service the timely data requests from the shop floor. Manufacturing execution systems (MES) have been introduced to assist with this problem. MESs also track work orders across disparate PLCs (a historical issue) and they track quality assurance and regulation data, for which there has been an increased demand.

Hybrid manufacturing is a combination of the process manufacturing and the discrete manufacturing types. This type could be a facility with two distinct manufacturing areas such as a brewery that runs a batch process in the brewhouse and discrete manufacturing in the bottling and package area. This type could also be a small process operating as a "machine" within a discrete manufacturing environment.

How the control systems for each of these manufacturing types are engineered and implemented plays a significant role in understanding the way these systems operate and how to effectively apply security to these environments. A process manufacturing type is usually designed, implemented, and maintained by fewer vendors and engineering firms and the vendors have significant influence on the overall design. A discrete manufacturing type can have many engineering firms, vendors, IT personnel (typically known as the Information Systems (IS) department), and an ever-increasing cloud utilization. As depicted in the above pyramid, the number of groups and companies involved increases significantly closer to the product creation activity and operations at the bottom of the pyramid. For example, the individual machines for a single production line typically involve multiple engineering companies known as OEMs. The control systems vendor equipment used by the OEMs can sometimes vary but will usually have a common communications interface card and protocol used by another company, known as a Systems Integrator (SI), that integrates these individual machines together within the production line. The IS department may work independently from the SI, or employ another SI, to integrate the production lines into the MES.

Image References:

- https://www.visschers-consulting.com/manufacturing-execution-system-mes
- https://www.emerson.com/resource/image/1317262/landscape_ratio16x9/1200/675/99d6f6aa9f417de88fa1368a16aeafe1/Sk/c014-deltav-dcs-item-1.jpg

Sector and Industry Common Terminology

- These terms can have general associations with a specific sector or industry, but may have alternative organization-specific names
- An organization or environment may have a combination
- Common System Terminologies (only a few of many)
  - DCS – Distributed Control System
    - Power, Oil and Gas, Water / Wastewater, Chemical, Pharmaceutical
  - SCADA – Supervisory Control and Data Acquisition
    - Power, Oil and Gas, Water / Wastewater, Transportation
  - MES – Manufacturing Execution System
    - Manufacturing (Automotive, Food and Beverage, Pharmaceutical, etc.)

Many of the terms that are used to describe the functions of an ICS have a historical basis that has evolved over time, blurring the lines between the vendors, their systems, and the types of environments in which they are found. Since many organizations still use these terms today, it is important to understand the environments in which they are typically applied.

A distributed control system (DCS) hosts a tight integration of a design tool, an asset manager, a visualization system, and a controller to interact with the production environment. These systems historically monitor, and control assets contained within a facility or vessel and have a primarily analog-based processing environment, involving elements such as temperatures, flows, and valves. Traditional DCS vendors tend to provide a service level agreement by dictating the design, applications, network, and security of these systems. For those vendor-provided solutions, the end user has little choice but to logically isolate the entire system into a trust boundary. We will cover trust boundaries later in this section.

A supervisory control and data acquisition (SCADA) system can have a lot of similarities to a DCS with a primarily analog-based processing environment. Some environments use a DCS as part of the SCADA system. A SCADA system is used to monitor, and control remote assets spread over a geographical area. Such assets could include pipelines, power transmission/distribution stations, or transportation systems. SCADA systems include telemetry capabilities for collecting data from these remote locations. Due to the remote nature of these systems, these telemetry networks operate over a variety of communication types including dial-up, satellite, cellular, radio, etc. Some newer systems within reasonable reach of a telecommunications system have incorporated a fiber Ethernet network. Since many of the remote assets can represent a mix of control device manufacturers and models, some level of customization may be present with regards to the telemetry communication system.

A manufacturing execution system (MES) is typically found in discrete and batch manufacturing. Although the MES is mostly an information system with extended functionality of the corporate Enterprise Resource and Planning (ERP), a MES also has mostly automated interactions.

These interactions include issuing open work orders to the production area (a.k.a. shop floor) and collecting closed work orders with production data about the product for archival purposes such as regulatory, quality, and supply chain tracking. A production facility utilizing an MES can be the most difficult to segment due to the complexity of the system and operational relationship between the ERP and the shop floor. The MES can also reside in or have components that reside in cloud-based environments.

Definition of Head End Process

Definition: Head End Process

- The head end process is a centralized group of assets that
  - May reside within:
    - A control system's OT environment (i.e., Level 3 of the Purdue Model)
    - A corporate IT environment
    - A cloud infrastructure
  - Typically has the closest data and user relationship to external systems; namely the enterprise network
  - Can be either part of or a combination of
    - central control for direct operation of an ICS environment
    - central production and data management for indirect operation of an ICS environment

The head end process (HEP), like the assets we have in the front of the classroom, is a collection of related assets used to monitor, manage, and even optimize the entire production environment. Many variations exist across sectors and industries regarding what assets make up the head end processing and where they reside. The head end processing may reside directly "below" the IT network, within Level 3 of the Purdue Model, and is commonly referred to as the OT environment. It is typical for a DCS and most SCADA systems to operate in an OT environment, though they can also operate in the corporate IT environment or within a cloud infrastructure. An MES can be found operating in any of these three environments, but a new trend is for SCADA to operate in a cloud infrastructure, which is an approach that smaller municipalities have been exploring.

Most external data and user interactions happen at the head end processing. The role these groups of assets play within the process can vary but usually receive commands directly from central control or indirectly from central production and data management.

Components

The head end processing components typically consist of traditional computer assets but also commonly contain a few dedicated embedded assets

- Traditional Computer Components
  - Data Server
  - Historian Server
  - Visualization Server
- Embedded Components
  - Process Controller
  - Data Concentrator
  - Area/Line Controller

Two types of processing components make up the head end: Traditional computer components and embedded components. Traditional computer components are those typically found in an IT system made up of standard servers and workstations. These components can use a combination of commercial off-the-shelf (COTS) hardware, software, and applications as well as custom-built components. The custom components are used to support the unique aspects found within an ICS such as embedded devices, proprietary protocols, and unique applications suited for the needs of a production system.

Embedded components may be included specifically to support traditional computer components.

Traditional Computer Components

- Data Server
  - Transacts data between embedded devices and traditional computer environments (i.e., Windows)
- Visualization Server
  - Stores (short-term) and delivers the graphics and tag (data reference) used for graphical user interfaces
- Historian Server (or Database Server)
  - Stores (short-term and long-term) and archives business-sensitive data for use in regulation, quality, and product tracking
- Analytics and Scheduling Server
  - Correlates and analyzes data from multiple sources for production use

Traditional computer components contain four primary functions: The data server, visualization server, historian server, and analytics and scheduling server.

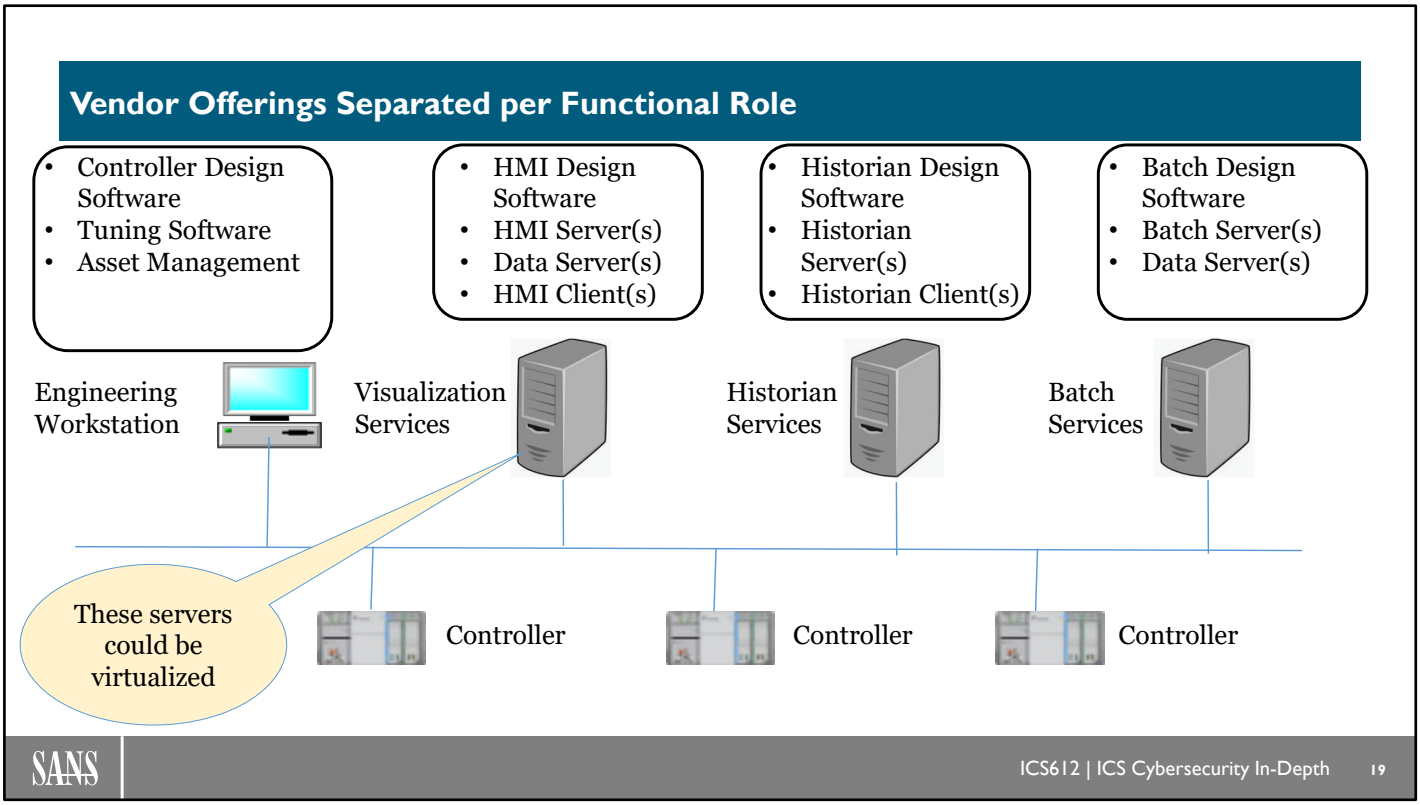
The data server transacts data between the ICS devices and the other traditional computing components. This is accomplished through at least two protocol stacks. At minimum, one protocol stack (e.g., Modbus) communicates to the ICS devices and another protocol stack (e.g., OPC) communicates with other traditional computers. Typically, the data server acts as a real-time protocol gateway and does not store data values.

The visualization server provides a graphical user interface, also known as a human machine interface, to the operations personnel. This interface is typically a visual representation of the current machine operations within the environment but may also be used as an interface to other production systems such as quality assurance, laboratory information management systems (LIMS) and overall equipment effectiveness (OEE). A visualization server may store short-term data, leverage long-term storage from a historian server, or interact with real-time data from a data server.

A historian server (a.k.a. database server) stores short-term, long-term, and archival data needed for regulatory, quality, and product tracking (i.e., product recall). This server may utilize many types of proprietary and COTS data storage technologies, usually depicted by the overall visualization and/or analytics and scheduling application vendor. Data can be stored as event-based (or triggered) or time-series data. With the large analog data sets associated with time-series data, a common trend has included the use of data compression techniques such as those used in the OSISoft Pi historian application.

The analytics and scheduling server includes more advanced services and is increasing in use across many sectors and industries today. This service correlates and analyzes both archived and real-time data to actively identify operational parameters of the production system that could be changed to reduce operating costs.

Such changes focus on improving production quality, increasing effectiveness, and extending longevity of the equipment. The resulting improvements can be pushed to the ICS either manually or automatically. Many of these more advanced services utilize the processing capabilities of cloud infrastructure for efficient analysis and aggregation of sanitized data across multiple similar operations.



Many of the automation vendors, especially distributed control system (DCS) vendors, have bundled the Design, Operate, Maintain, Inform, and Optimize software into functional offerings in support of solving customer automation challenges. For instance, it's typical for an automation vendor to package the design software on a separate server (virtual or physical) to keep the design role separated from the visualize or operate role.

It is also typical for these software bundles to be separated based on the quantity that will be deployed in the system. For example, there may only be one Engineering Workstation with the design software deployed per system while the HMI clients may have many workstations deployed with the design software.

Since the core service of these systems is to interact with data stored on ICS devices through ICS protocols, the use of one or more data servers, as well as the use of one or more database technologies, may be present across the servers and workstations.

Many of the automation vendors offer their software bundles for a virtualized environment, increasing security and resiliency capabilities in the head end process.

Specific Applications Used in Head End Processing

Multiple applications make up production systems (i.e., DCS, SCADA and MES)

- Data Transaction
  - RSLinx, Kepware, Matrikon, AutoSol, etc.
- Data Storage
  - MS SQL, OSI Pi, MS Access, MS Excel, etc.
- Production Management
  - Ignition, FactoryTalk, Opcenter, Oasys, DeltaV, CIMPLICITY, etc.
- Operator Support
  - Training simulators, test environment, contingency analysis

Most recognize the system-level terms and the primary vendor names behind them; however, many do not realize the individual applications used by the vendors to pull the systems together. Many vendors use both proprietary applications and applications from other vendors, including those traditionally found in a corporate network (e.g., Microsoft SQL). System integrators sometimes have, or require, more flexibility to put a system together using multiple vendor applications available on the market to meet specific production requirements.

Compiling a detailed list of applications operating in the head end process is necessary for determining what communications should exist and what vendor vulnerabilities are truly applicable to the environment. This knowledge can also be applied when working with the control systems design team to minimize duplicate functionality, enforce application segmentation, and improve the overall security of the head end process.

Embedded Components (1)

- Process Controller
  - Typically associated with and supplied by a DCS vendor
    - E.g., ABB AC 800M, Emerson Ovation, Rockwell ControlLogix, Schneider Electric Foxboro 280, DeltaV, Siemens S7, etc.
- Data Concentrator
  - Found in remote operations across a large geographical region
  - A data concentrator, when used, centralizes the flow of telemetry data
  - Typically associated with remote terminal unit (RTU)
    - Bristol 3305, Honeywell RC500, MiCOM C264, etc.

For some tightly integrated systems, such as a DCS, an embedded processing component (often referred to as a process controller) is required. A SCADA system may utilize a data concentrator to centralize the flow of data over the telemetry system. An MES may utilize an area or line controller to minimize the direct interaction, or changes, to machine PLCs or to standardize communication interfaces with the various types of shop floor equipment.

The direct association between the embedded component technology, the vendor/model, and their typical industry use case has retained some historical relevance but there are situations where crossover of a product line is blurring those associations. For example, Rockwell Automation's ControlLogix platform and Siemens S7 platform can perform multiple functions. These advanced, traditional PLC devices have adopted the label "programmable automation controller" (PAC).

Embedded Components (2)

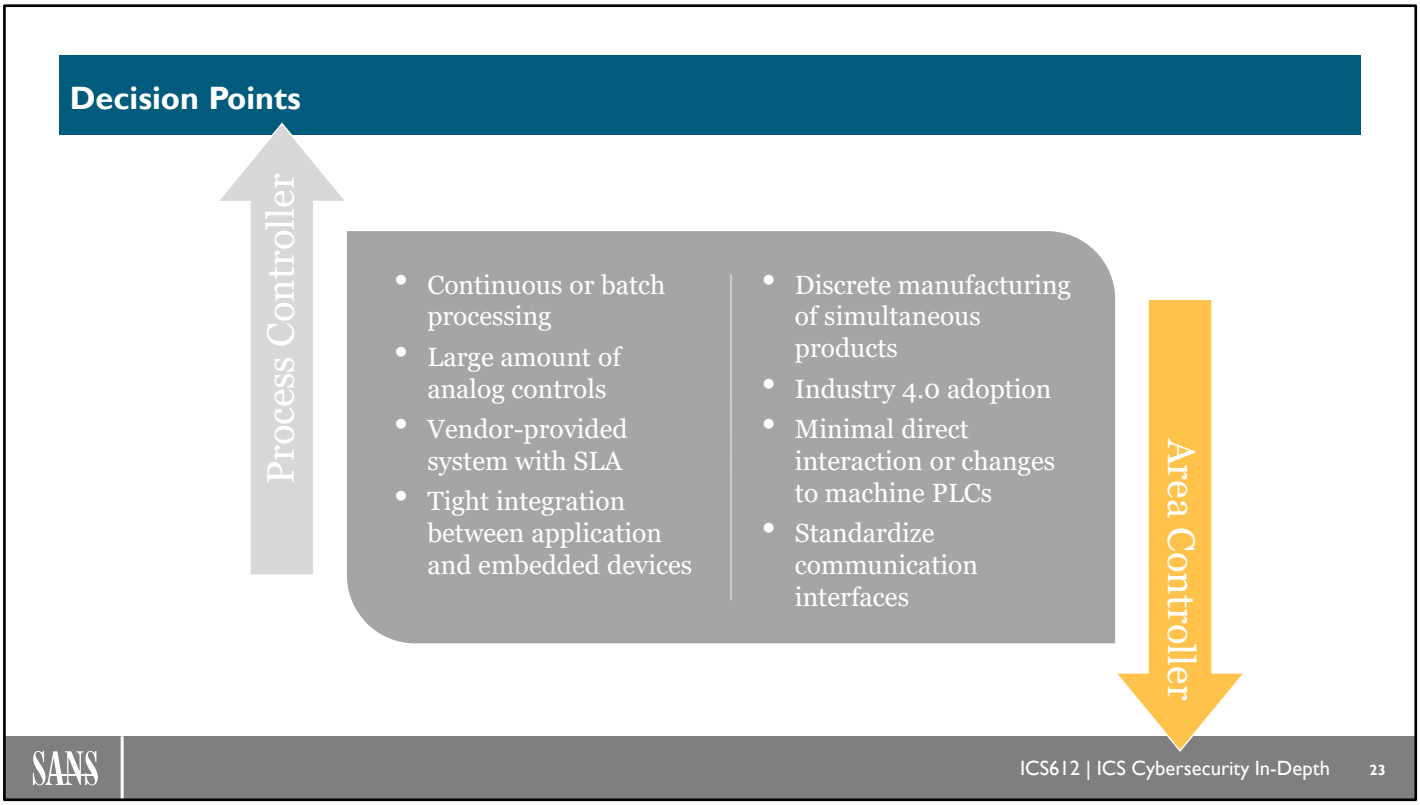
Area and/or Line Controller

- Found at larger facilities with multiple production areas or remote operations across a large geographical region
- The use of area or line is interchangeable (i.e., an area controller might be connected to multiple line controllers)
- Typically, a dedicated PLC (a.k.a. programmable automation controller)
  - Rockwell ControlLogix, Siemens S7, Schneider Electric Modicon, etc.

Many ICS environments operate ancillary systems which can easily be overlooked and are not generally associated with the primary control systems, including DCS environments. The services these ancillary systems provide are typically thought of as benign but could cause undesired and significant impacts to production.

Reference:

- https://www.controleng.com/articles/plc-vs-pac/



Consider the various decision drivers when designing and engineering automation and process control solutions for a given operating environment. Process controllers will typically be implemented within continuous or batch processing environments that rely strongly on the support and service of the control system vendor. Area controllers would typically be found within discrete manufacturing process environments with infrequent changes and a fairly stable low-interaction process environment.

Common Head End Process Network Infrastructure Components

Network infrastructure can be simplistic or complex

- Server Environment
  - Bare metal -> virtual servers -> or combination
- Storage Environment
  - Local USB (Backup) -> Local RAID 5 -> NAS -> SAN
- Network Environment
  - Basic Layer 3 switches -> core switches
  - Local routing -> VRRP
  - Coded IP -> hostname files -> DNS
  - Unmanaged time -> external time source -> GPS time server

The network infrastructure supporting the head end process can vary greatly between a simplistic design and a complex enterprise-style environment. A network infrastructure that supports a head end process has different needs than those of the IT network. Some vendors, such as DCS vendors, dictate the network infrastructure for their environment. The pros of vendor dictation of the network infrastructure is the increased level of service gained in the operation of the system. The cons include increased challenges for an organization to implement a common security architecture across their varying systems.

There are also large discrete and hybrid organizations that have relied on their IT department to supply the infrastructure for their head end process. Theoretically this should provide the organization with increased network reliability and improved issue resolution. Concerns with this arrangement, however, include conflicting business requirements between the operations and IT environments as well as integration challenges when introducing production equipment or operations changes. This type of arrangement commonly, although not always, results in a shadow IT network (one that is independently owned and operated by operations staff), though this can be resolved.

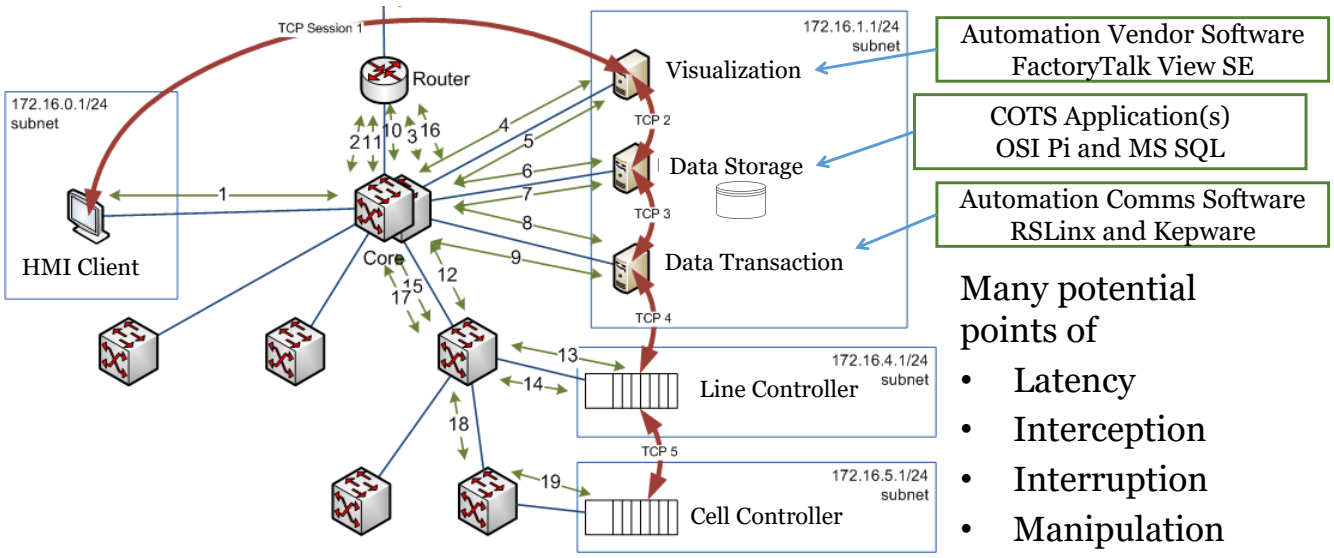
There are many head end process environments, including ancillary networks, where the selection of the network infrastructure is more open. This increases the likelihood of having minimal network and security design principles in the design. When project budgets allow, incorporating good network and security design principles into these more open environments pays off with increased availability and protection.

OT Trends

	Control Centers	Control Rooms	Field Environment	Plant Floor	Level 3 and Beyond
Physical Servers			X	X	
Virtual Servers	X				X
Hybrid		X			
RAID			X	X	
NAS		X			
SAN	X				X
Unmanaged Switch			X	X	
Layer 3 Switches		X			
Core Switches	X				X
Local Routing		X	X	X	
VRRP	X				X

The topic often comes up of old legacy equipment. ICS is typically painted with one broad brush that would lead one to believe that all environments are the same and have very little innovation in technology. In general, you will find current virtual environments, systems, and network infrastructures in the OT environments that have few or no ICS devices such as control centers. In operational facilities where you may find control rooms where operators sit and manage the overall process, you may find a hybrid of newer technologies and traditional physical servers, systems, and network infrastructures. Down on the plant floor or out in remote field environments you will not typically find the more innovative technologies; you would be more likely to discover physical servers, ruggedized systems, and industrial environment network components.

The Life and Risks of an ICS Data Point



As the components that make up the head end process are designed and integrated together, it is important to understand the flow of data between these components as a system. As depicted, the number of component interactions, data hops, and communication sessions required to support a single operation can be quite extensive. The level of this complexity can have a direct influence on the overall system latency as well as the attack surface for data interception, interruption, and manipulation. This depiction will vary between vendor applications and systems as well as sectors and industries; however, taking the time to learn the underlying data flows and application relationships can make it possible to improve the application of security controls and monitoring. In many situations this level of understanding has been used to determine where "ghost" issues reside in the environment, thereby improving the availability and reliability of operations.

Reference:

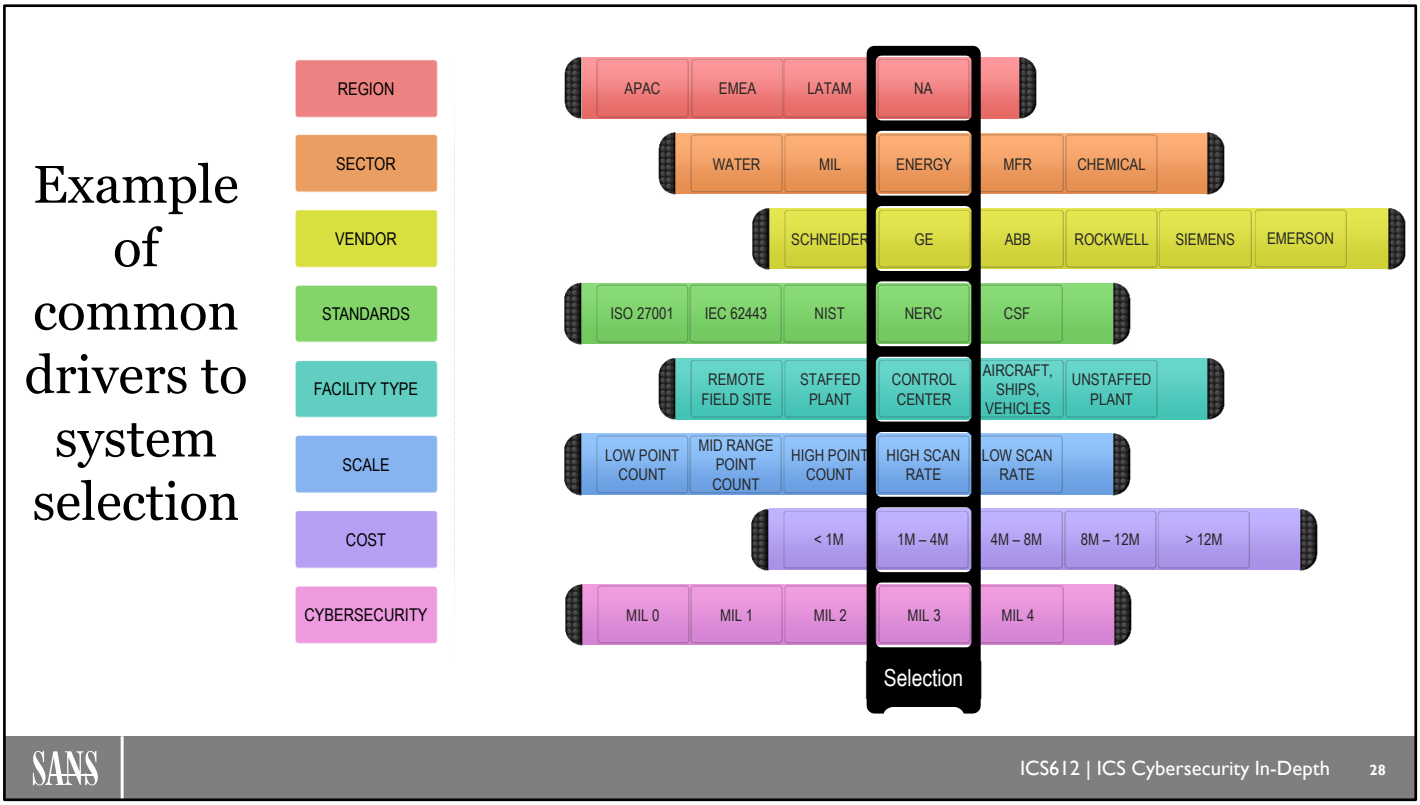
- https://docplayer.net/5720012-Securing-process-control-systems.html

Identifying Head End Process Components

- Vendor manuals and publications
- Engineer drawings and documents
- Interviews with system and process owners
- OT application databases and tag lists
- Project files and device configuration files
- Screenshots of active operator systems
- Software inventory
- Network traffic capture

Identifying the head end process components within an environment requires studying multiple sources. Some of these sources of information may not be readily available but through persistence and validation these components can be identified.

This identification process can provide an improved understanding of why these components exist and the purpose they serve and can clarify what type of security controls can or should be applied. This understanding does not fall with one vendor or one individual in the facility. Many folks in the industry have just enough of an understanding of the components to fulfill their daily activities. Many operations staff additionally rely on vendors to have a more complete understanding and ability to solve any unforeseen problems that arise. However, since these systems are built using multiple interacting vendor applications, this quickly becomes a shortfall. Searching for the "why," like a detective determined to seek out the truth, is essential to understanding what network and security improvements can be applied. Using passive network and process monitoring tools can be effective in unraveling the causes and effects between clients, services, protocols, and devices, but only when following a rigid methodology.



As you consider the various elements of an operational technology environment and the decisions that need to be made, think about some of the key determining factors that may guide your solution and engineering options. Starting with the region where your operating environment is, you may not be allowed to purchase from certain manufacturers, or they may not be allowed to sell to your organization. Next as you look at the sector that you are in, there may be specific requirements for which only a limited number of solutions and approaches exist, as with the nuclear sector for example. Depending on your region and your sector, there may be specific vendors that are available and there may also be specific standards in place that must be complied with.

After evaluating those four leading drivers, you will need to consider some specific variables that include your design facility type, the scale of your system, your budget, and finally your specific desired cybersecurity maturity capabilities.

Covfefe Coffee Automation Requirements

- Process requirements drive automation vendor selection
- In our requirements, we wanted Ladder Logic support as this is used in the majority of discrete applications
- We also wanted Function Block support as this is seen in most DCS offerings
- The Rockwell Automation – Allen-Bradley PAC / PLC met our requirements

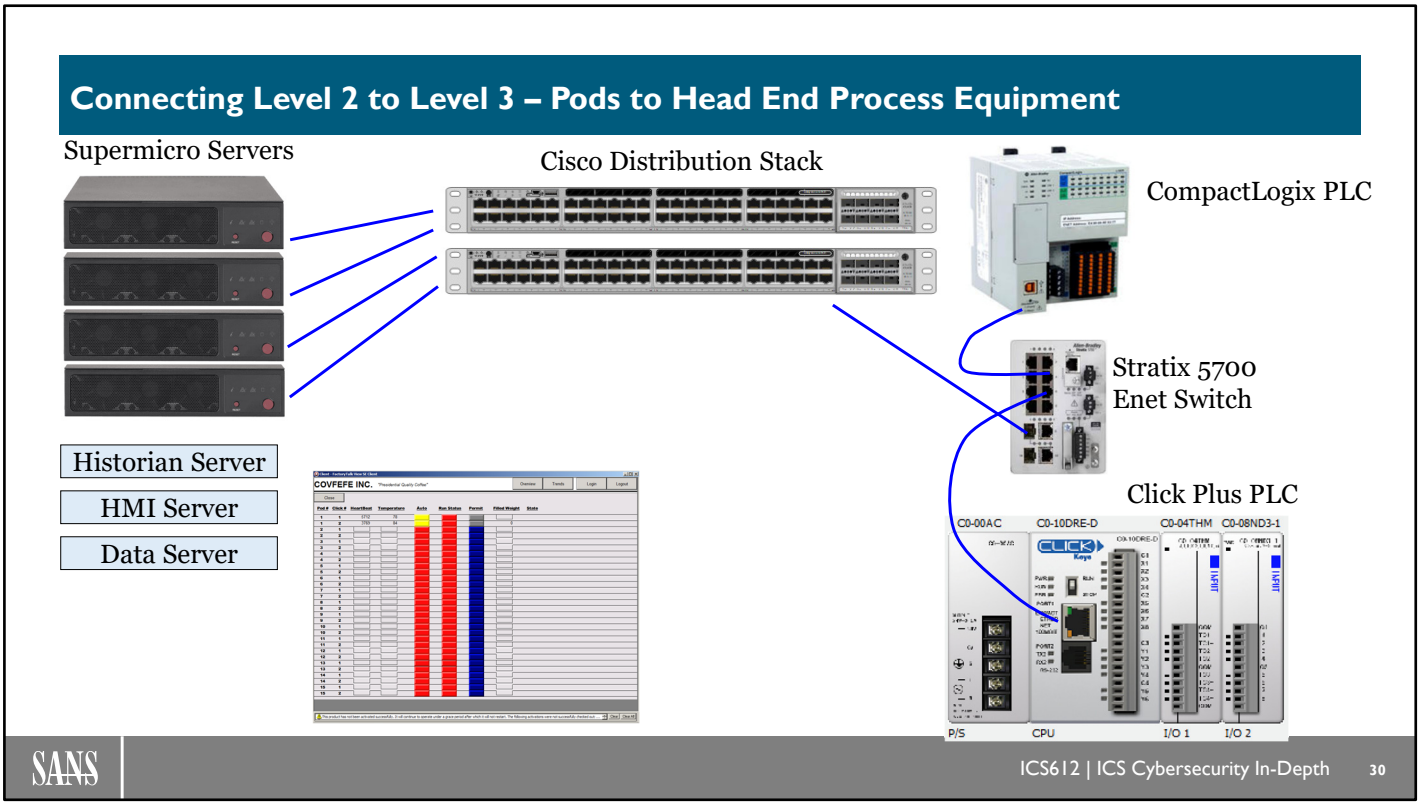
Requirement	DCS	PAC	PLC
Digital I/O (Sw, PBs, Relay Outputs)	X	X	X
Analog I/O (0-10 VDC & 4-20 ma, T/C)	X	X	X
Ladder Logic for Sequence Control	-	X	X
Function Block for Continuous Process	X	X	-

Process requirements drive the automation requirements. In the classroom coffee factory setting, we need to interface with digital inputs and outputs like switches, pushbuttons, lights, and relay outputs. We also need to be able to read thermocouples, drive analog outputs, and read analog inputs.

We specified the Ladder Logic language because we see this language used in a majority of discrete applications. To be specific, Ladder Logic is used in a majority of sequential operations. Also, Ladder Logic is well known by maintenance personal and other automation engineers who may not have had exposure to higher-level programming languages like C, C++, C#, Java, etc.

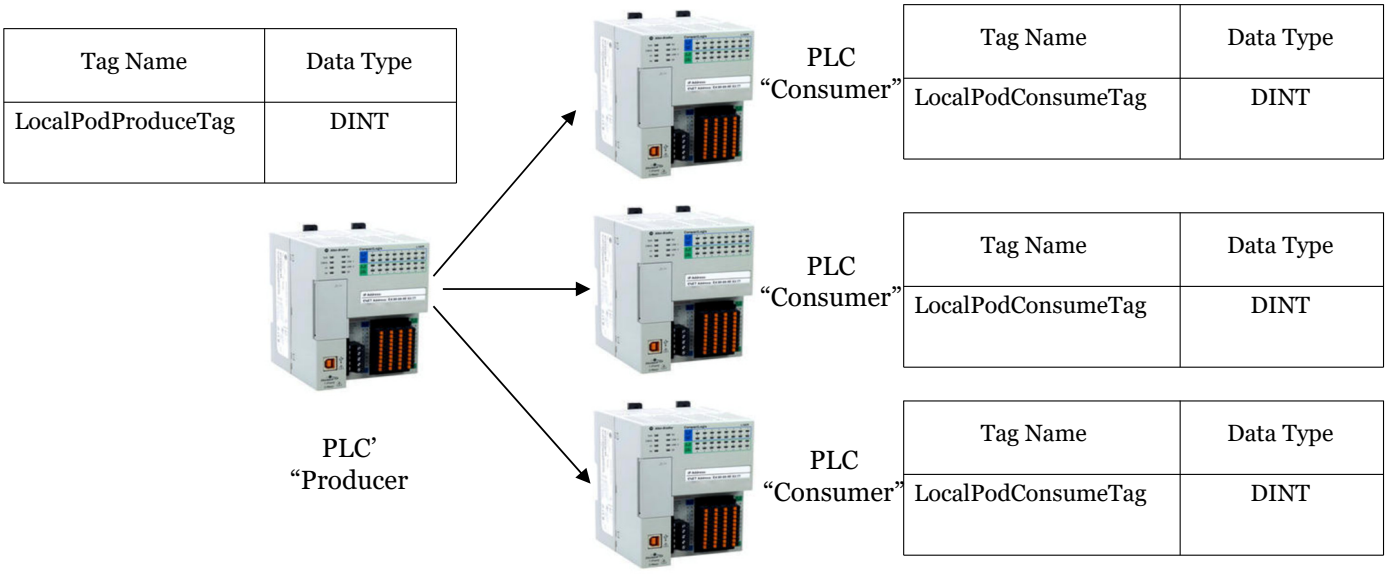
In our classroom coffee factory, we wanted to be able to program using Function Block language as this is used by most DCS offerings, as DCS are used for controlling continuous processes.

We chose the Rockwell Automation Allen-Bradley solution because it offers the hybrid functionality of a programmable automation controller (PAC), supporting both Ladder Logic and Function Block programming languages. The traditional PLC platform, such as the Click Plus PLC, doesn't offer Function Block programming while the DCS platform doesn't offer the Ladder Logic language for sequential operations.



We have worked with our Click Plus PLC and the CompactLogix PLC as Level 0, 1, and 2 assets. We are now going to perform labs that connect the CompactLogix controller to Level 3 assets as described in our previous slides and classroom discussion.

Produce / Consume Tag Architecture



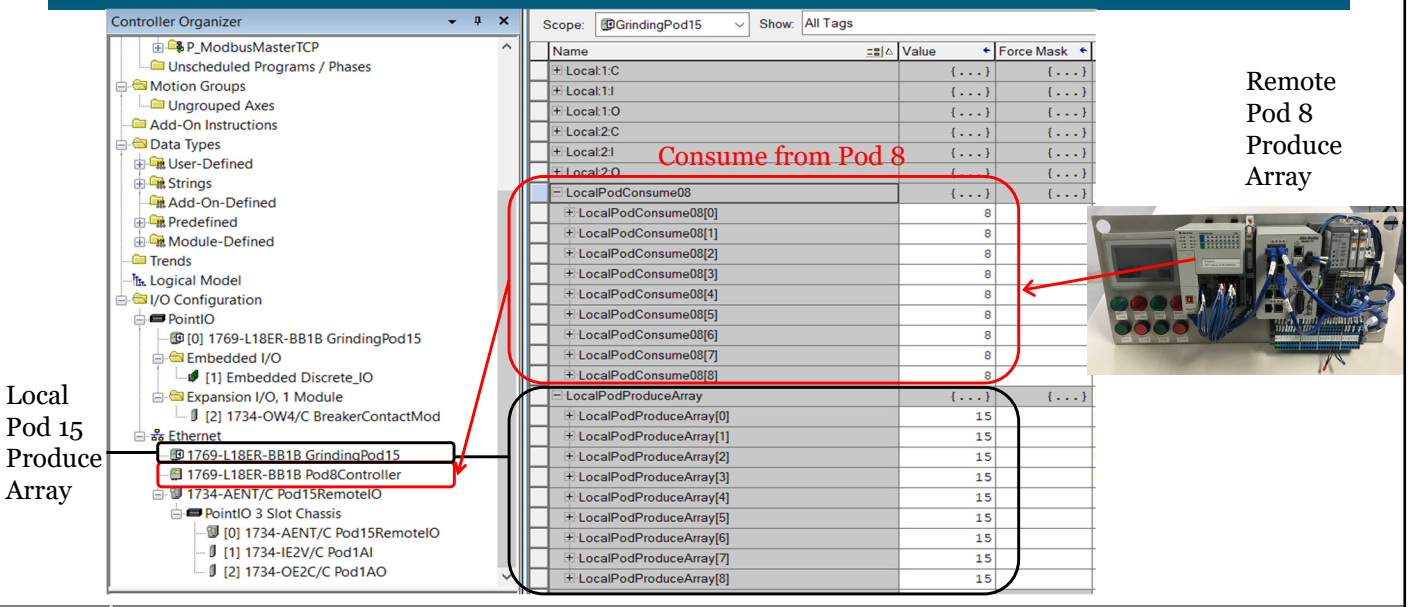
Produce Tag: A tag that a controller makes available for use by other controllers. Multiple controllers can simultaneously consume (receive) the data. A produced tag sends its data to one or more consumed tags (consumers) without using logic

Consumed Tag: A tag that receives the data of a produced tag. The data type of the consumed tag must match the data type (including any array dimensions) of the produced tag. The RPI of the consumed tag determines the period at which the data updates.

Reference:

- https://literature.rockwellautomation.com/idc/groups/literature/documents/pm/1756-pm011_-en-p.pdf

No Ladder Logic – Just Configured Like I/O Modules



In order for a Rockwell Automation PLC to consume another PLC's tag, it must be mapped as a remote Controller in the I/O configuration tree. Likewise, if the local PLC needs to produce tag information for another PLC to consume, that tag must be configured as a "Produced" tag in the tag property type.

Lab 2.1: Connect Pods to Level 3 Infrastructure

Go to the Lab Workbook: Lab 2.1

This page intentionally left blank.

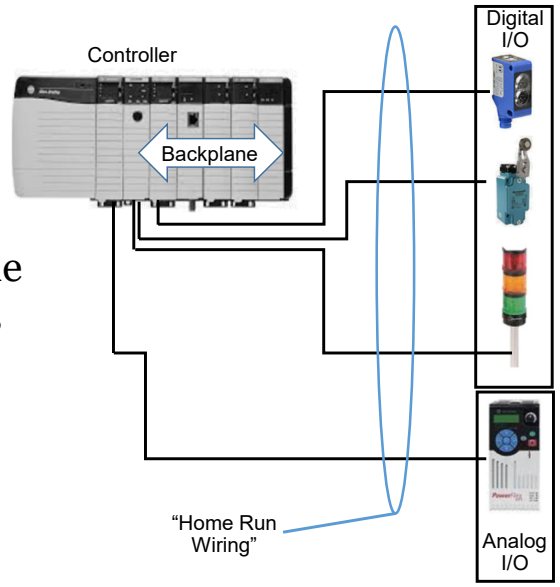
Head End Process Overview Checkpoint 2.1

- Earlier in the course, we established local control components and connected those to shared local process components.
- We have begun connecting those local elements together to enterprise-level infrastructure for additional visibility, optimization, management capabilities, and cybersecurity tools.
- To connect these systems into a larger operational system, there is information that we need:
  - Identify data that needs to be collected from the PLC
  - Identify the associated PLC address and tag names
  - Configure the data server to pull data from the PLC

This page intentionally left blank.

Local I/O – Local to What?

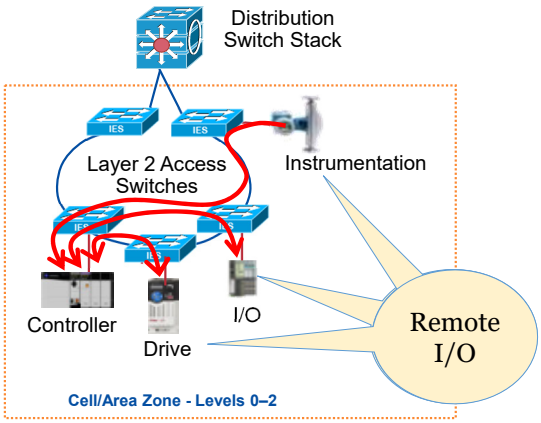
- Local I/O is oftentimes considered I/O points that are wired directly to cards placed in the local rack
  - "local" meaning it is wired to cards located in the same physical rack as the PLC
- Communications to the I/O modules in the rack are considered at "backplane" speeds and don't have to take in consideration network "latencies" or network "jitter"
- The downside is that all the devices have to be wired to the chassis and this can be quite expensive



Before networking capabilities could support ICS applications and networking I/O platforms became widely accepted, digital and analog I/O subsystems were wired directly to a module located in the local PLC Chassis. While this eliminates the traditional networking headaches and considerations for latency and jitter, it is a costly solution to wire all the system sensors back to one central location.

Remote I/O

- It is possible and sometimes desirable to put the I/O card near the physical I/O points and run a network cable for connectivity back to the controlling PLC
- This eliminates "home run" wiring
- We call this, "remote I/O"
- The PLC "connects" to the I/O over some network and the I/O is controlled by the PLC
- Normally mapped to the PLC in the software

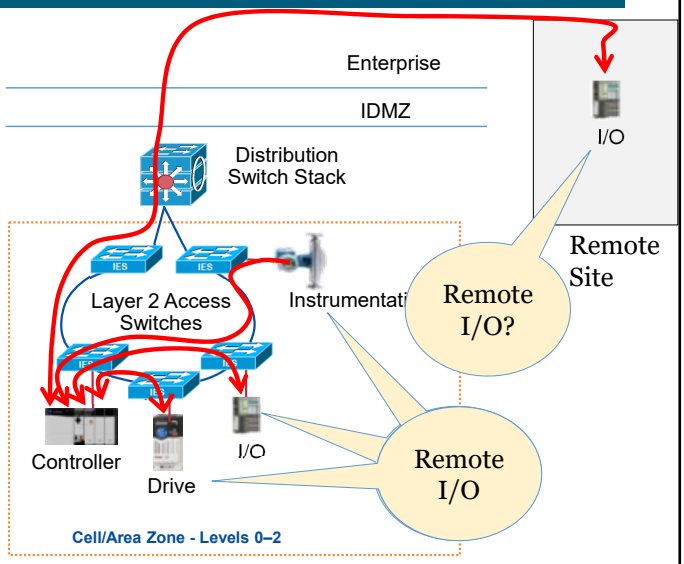


Sometimes it saves on wiring if you place an I/O card in a remote panel near the physical I/O and run a network cable from the remote panel back to a main panel where the PLC is located.

Networking has enabled financial cost savings insomuch as the physical devices are wired to a local panel, eliminating a bundle of wires running from the physical I/O back to the panel where the PLC is housed. This wiring scheme, sometimes called "home run" wiring, has been simplified by running a network cable in place of all the home run wires.

Remote I/O over Nondeterministic Networks

- Sometimes, customers want to control remote I/O over an unreliable, nondeterministic link (like an internet VPN connection) or I/O located many network hops from the controller
  - This is not a good idea!
- Most "slow" I/O connections will time out between 250 and 750ms
  - The Controller will think the I/O has died or something has happened if the I/O doesn't report back within some predetermined time



Some customers want to connect remote I/O points to a controller over unreliable and nondeterministic networks. This can arise when remote systems or remote processes have a relationship to a controller. It is natural to want to connect the I/O points to the related controller, but one must consider the network hops and the reliability and determinism of the network. So, what criteria should determine if the remote I/O points should be controlled by a local controller or if they can be connected and controlled via a remote controller?

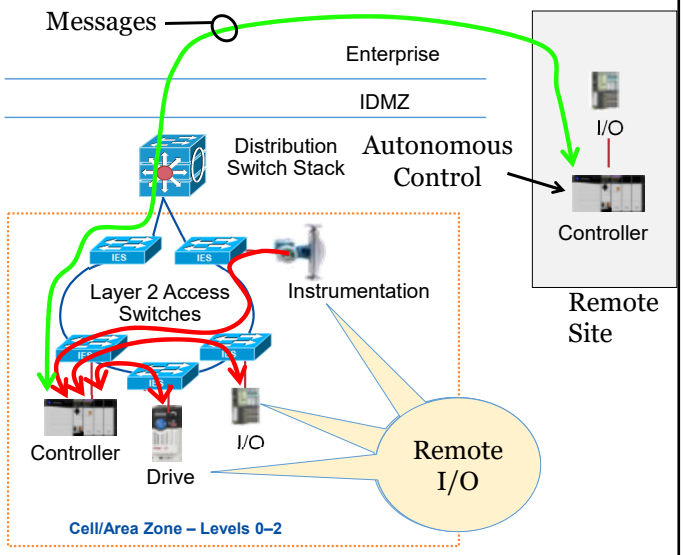
There are a couple of criteria that one can consider when attempting to determine if an I/O should be mapped to a local controller. The number of network hops will add latency and depending on the process being controlled by the PLC, you must determine if the network latency is acceptable for controlling the process. Second, some PLC controllers will not allow for a latency above 750ms because the PLC wants to have a heartbeat response from the I/O before a predetermined watchdog timeout. Some controllers will not tolerate a large network latency.

It should be noted that some customers have tried to use the internet with a VPN connection to a remote router to attempt to control I/O points. This is a solution that has been fraught with unreliable results and is therefore not recommended.

In this example, the customer wanted to add the I/O points at the remote site and have the remote process controlled by the main site controller. The customer wanted to use a site-to-site VPN connection and hoped this solution would work. It turned out, the internet connection with the site-to-site VPN was too nondeterministic and unreliable to map the remote I/O to the main site controller. A site-to-site VPN connection will work fine for HMI communications and other less time-critical connections, but for PLC communications, the communications must be reliable and deterministic.

Remote I/O vs. Using an Autonomous Controller

- Remote connections that cannot support the I/O connection timeouts because of unreliable network connectivity will need to use a second controller for autonomous control
- The autonomous controller will control the process and report status back to the interested clients through messages or database transactions
- Don't confuse this with just reading an input value like a tank level without controlling the tank level



If the network latency exists such that a PLC controller is many network hops away from the I/O points or the network is simply unreliable, then it is suggested to add an autonomous controller that controls the local process and reports status through messages or other reliable protocol means.

In this example, the remote site has a process that requires control and therefore I/O points have been added at the remote site. A remote site controller was added to this architecture to control the remote site process; the remote site controller will report status back to the main site controller. This architecture allows the remote site controller to work autonomously from the main site in the case where the remote site cannot successfully communicate with the main site.

In your prior labs, your Click Plus PLC was acting as an autonomous controller and it could do work autonomously from the Allen-Bradley PLC. They communicate via Modbus messages, all the while being able to control their local processes.

In your next lab, you will map local I/O to a local controller.

Lab 2.2: Remote I/O

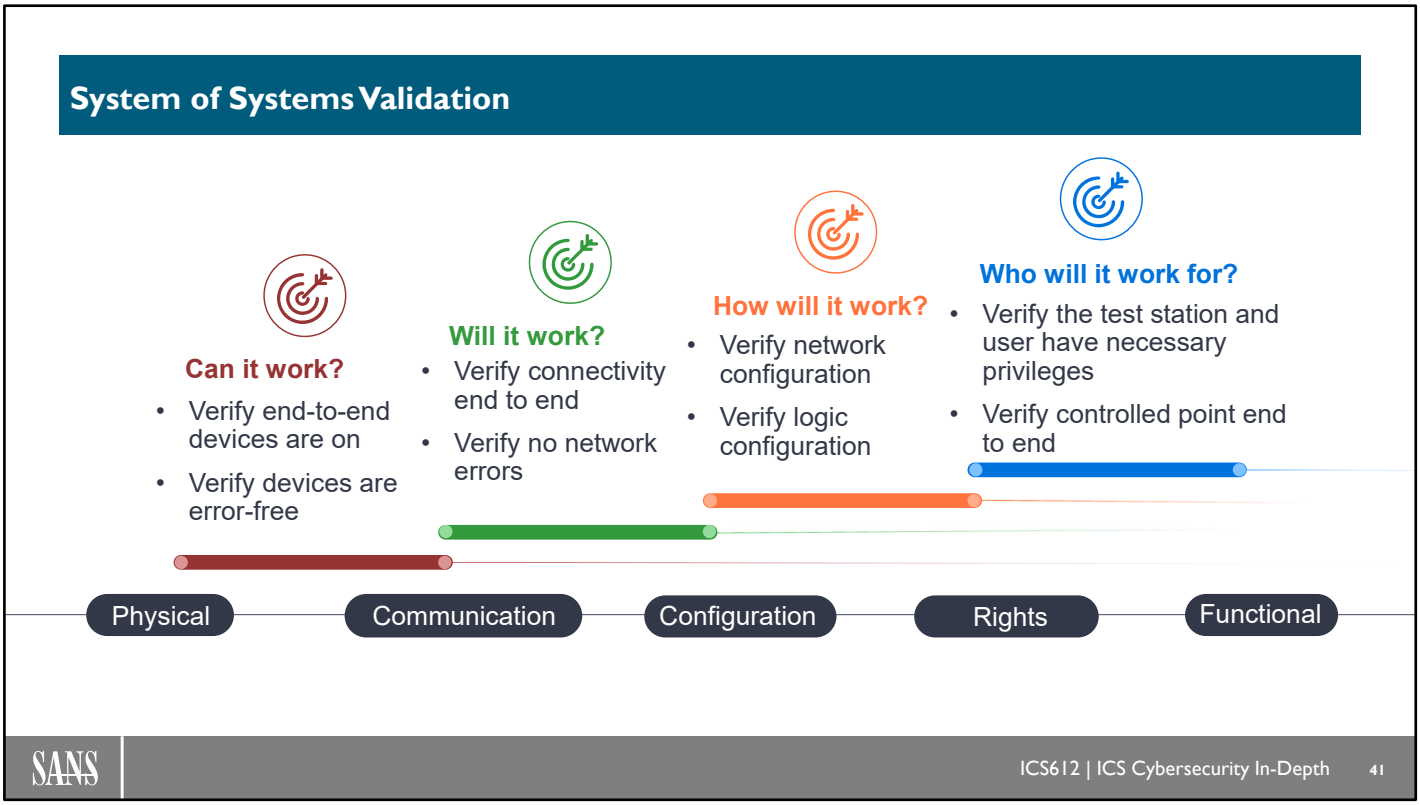
Go to the Lab Workbook: Lab 2.2

This page intentionally left blank.

Head End Process Overview Checkpoint 2.2

- In process environments where I/O modules are near the field devices and control capabilities are distant, it is helpful not to have to wire all control cabling over a distance
- In our classroom environment, it is also easier to run network cables from all the shared student Pods and control the head end process items through remote I/O rather than running control cabling throughout the classroom
  - Identify the passthrough functions being performed by the various ICS devices
  - Understand the protocol and signal conversions occurring in the remote I/O

This page intentionally left blank.



Developing an approach to validating a system's functionality can in many ways mirror approaches used to troubleshoot operational systems. Always start by verifying the physical elements end to end before moving on, as a physical problem will present itself in many inconsistent ways if not dealt with initially. After you have validated the physical device functionality and determined that the system can work, it is time to determine if it will work, by ensuring the communications path is available and error-free. With an available and functional communication path, you will need to verify how the system will work by reviewing the various configuration settings to ensure the system is configured as it is intended. Verify you have the appropriate rights as a user, with appropriate Area of Responsibility control if needed, and ensure the applications and services are also operating with the necessary level of access rights to operate as required.

The last component is to validate operational functionality, which may require some special equipment, wiring changes, or special operating conditions.

Example Validation of Live System

- New or upgraded electric or natural gas transmission SCADA system
- After system specification, build, FAT, unstructured tests, SAT, performance tests, and ultimately preparation for cutover testing
- You need to validate the integrity of all of the underlying technology: Physical, Communications, Configurations, Rights, and Functional

An example of a large-scale SCADA system is a great case study to consider when thinking about system validation and a process to ensure integrity within a system. Performing a system integrity validation on a geographically diverse SCADA system that is operational can be complex and a risk to reliability. Operations personnel and leadership need to be well informed of the validation effort, and a well communicated and highly coordinated plan is necessary as field environments may be impacted and unavailable for operational control during specific site testing.

But How Do We Really Validate – Safely?

- Ultimately, you will need to validate output control without actually operating the elements in use – this can be performed by placing the field environment in a control inhibit bypass or by unwiring the controlled points, while measuring the changes to the points during validation
- The same can be done in reverse, by unwiring inputs – applying various signal generators on the points to validate that the operator screen is showing the correct changes on the correct points
- This process may be performed on a sample set of field points or an entire field point verification list

The validation efforts will typically examine a sample set or full field point checkout. This is done by testing the controlled outputs to ensure the right points are being actuated and utilizing test equipment to generate input ranges on sensor points to ensure the point mapping and scaling is all correct.

Utilize Vendor Tools to Validate Programs

- Once a system is running, verifying that the programs loaded into the PLC, HMI and other systems is often done to make sure program changes have not been done or to verify the proper program is running in the system
- Sometimes, uploading the program and comparing it to the "golden" or validated copy is done
- Sometimes, all we can do is compare the hash of the two files
- Some vendors however will provide comparison tools so you can identify what is different between the two files
- You will use a vendor's compare tool in this next lab

After a system has been commissioned, project files are stored until a system restore is required. We also find that when a machine or a process is running differently, we find ourselves wanting to compare the running systems programs with a known good archived program. One method of comparing a known good archived file and an uploaded program file is to generate a hash for both files and do a comparison. While this method will tell you if the files are identical, it will not tell you what is different between these files if the hashes don't match.

Some automation vendors will provide tools that allow you to do a deeper comparison of two or more project files. Some of these tools will not only tell you if there is a program difference but they will also indicate if any data values are different as well.

Lab 2.3:Validate Functionality

Go to the Lab Workbook: Lab 2.3

This page intentionally left blank.

Head End Process Overview Checkpoint 2.3

- With the initial system of systems interconnectivity and interoperability in place, we can test functionality from end to end
- Many of the tasks we will be performing throughout the course will add additional security capabilities to this operational environment and doing so will require us to reconfigure components and revalidate functionality
- Looking at the system as it is now, consider attack vectors at the various levels:
  - Attacks on the local student kit components
  - Attacks on the Shared Pod equipment
  - Attacks on the head end or targeting the network

This page intentionally left blank.

ICS612 Section 2 Outline (3)

- Head End Process Overview
- Lab 2.1: Connect Pods to Level 3 Infrastructure
- Lab 2.2: Remote I/O
- Lab 2.3: Validate Functionality
- ICS Secure Architecture
- Lab 2.4: Network Infrastructure Configuration
- Process Communication and Data Flow Mapping
- Lab 2.5: Map Communications for the Environment
- Lab 2.6: Configure Connections to Process Visualization
- Local Attacks and Process Manipulation
- Lab 2.7: PLC Device-Level Attack
- Lab 2.8: OPC Discovery Attack
- Lab 2.9: Local Network MITM Attack

This page intentionally left blank.

ICS Secure Architecture

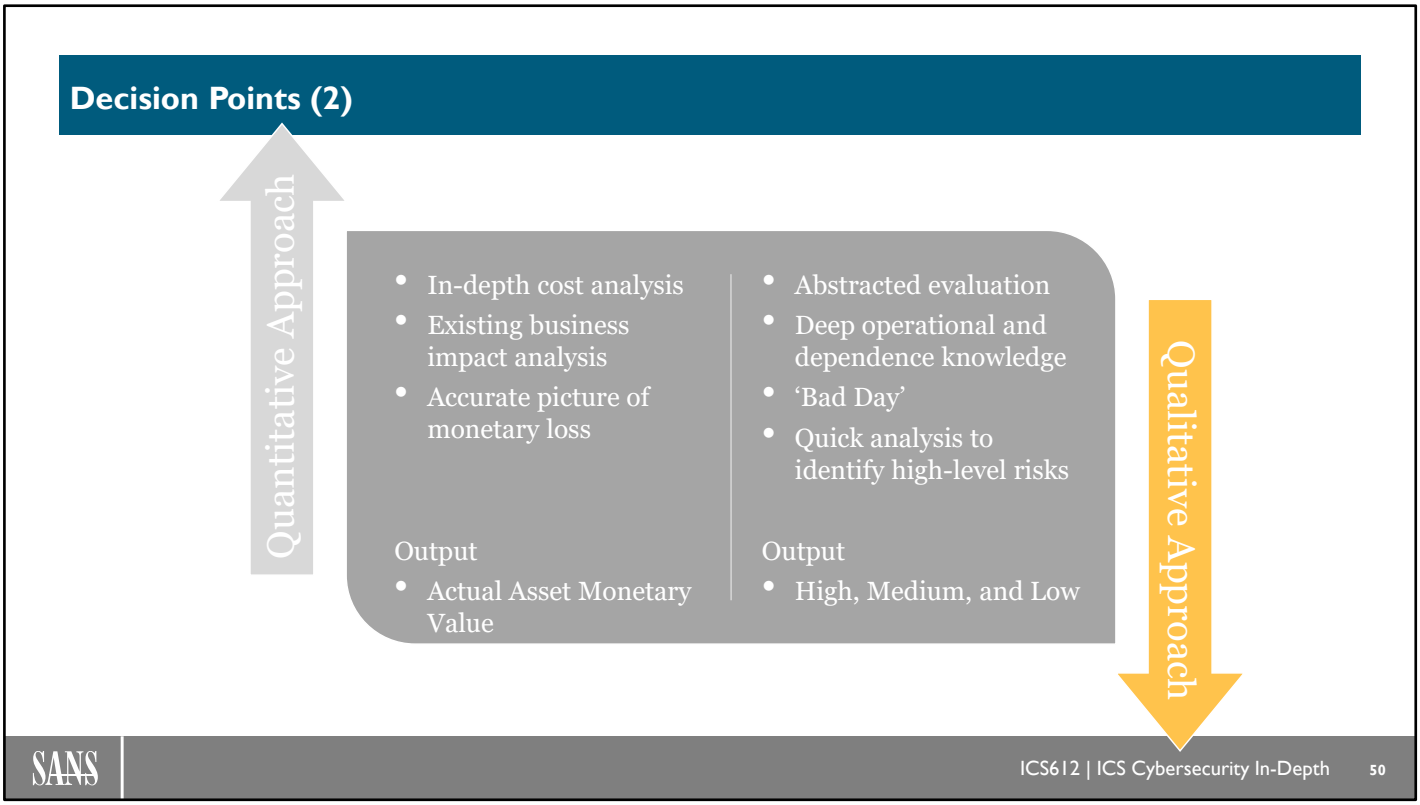
Critical Assets
Security Zones, Conduits, and Trust Boundaries
In-Level Segmentation

This page intentionally left blank.

Critical Assets

- Identifying Critical Assets
  - Quantitative Valuation of Asset
  - Qualitative Valuation of Asset
  - Direct Loss and Indirect Loss
  - Asset Type
- Value to Organization vs. Value to Threat
- Categorization of Critical Asset

Knowing what the critical assets are within the ICS environment and how they support the organization's business goals is necessary to effectively apply security. This information will be used to design, evaluate, maintain, monitor, and respond to security events. Many security standards and frameworks, including remote I, specify that a process is needed for maintaining a documented inventory of critical assets. Since this process can involve different approaches and biases, a methodology must be selected and documented for use across the organization.



A quantitative approach to asset valuation is a thorough study of the asset's monetary value. This involves an in-depth cost analysis with no guess work. It follows a similar process to a business impact analysis and provides an accurate picture of the monetary loss that would result from impact to the function of the asset or its replacement. Accuracy in the replacement costs must include costs associated with equipment, activities, personnel, external entities, raw material, and product/service replacement.

A qualitative approach to asset valuation is a process of abstraction that requires a deep operational and dependence knowledge, also known as the 'Bad Day' scenario. Instead of a specific monetary value, ranges such as High, Medium, or Low, are used to represent an impact correlated back to some monetary value. This can be used as a quick method of risk analysis to identify the high-level risks and quickly justify spending and priorities; it can later be supported by a quantitative analysis.

Direct vs. Indirect Loss

- Direct Loss
  - Replacement cost of the asset
    - Including physical parts, engineering, rebuild, installation, integration, etc.
- Indirect Loss
  - Downstream effect of the direct loss off an asset
  - Contributed loss from a compromised or dysfunctional asset
    - E.g., production downtime, loss of service, impact to quality, regulation, brand and reputation, etc.
  - May have immediate cost and/or compounding costs over time
  - Compounding costs may not stop by recovery of asset alone (e.g., public opinion)

Losses can be associated with replacing tangible physical and intangible logical (i.e., custom configuration) assets. However, these direct losses must also consider all of the associated costs such as the physical parts and time associated with engineering, rebuilding, installation, integration, testing, etc.

In the total picture of loss, one must also consider indirect losses. All direct loss will also have a component of indirect loss. An indirect loss can also be realized through malfunctioning, misuse, and abuse, causing the asset to operate outside of its intended parameters. A disruption of asset availability will also trigger an indirect loss.

Asset Types

- Physical Asset
  - Tangible
  - Impacts are typically straightforward to identify
- Logical Asset
  - Intangible
  - Impacts can be more abstract to identify due to their utility value
- Human Asset
  - Tangible and intangible
  - Immediate 'impact' will vary; loss of key talent can be complex
  - A lasting impact to individual and organization will exist

Assets can be categorized into three primary types. A physical asset is tangible, and impacts are typically straightforward. For example, a centrifuge operating erratically should be easily identifiable as a problem by those familiar with the normal operating characteristics of the asset.

An impact to a logical asset is not as easily identifiable without having a mechanism to baseline and compare it with a configured accepted state. For example, a misconfiguration of a PLC program is not detectable without a process to validate and compare against a known good configuration.

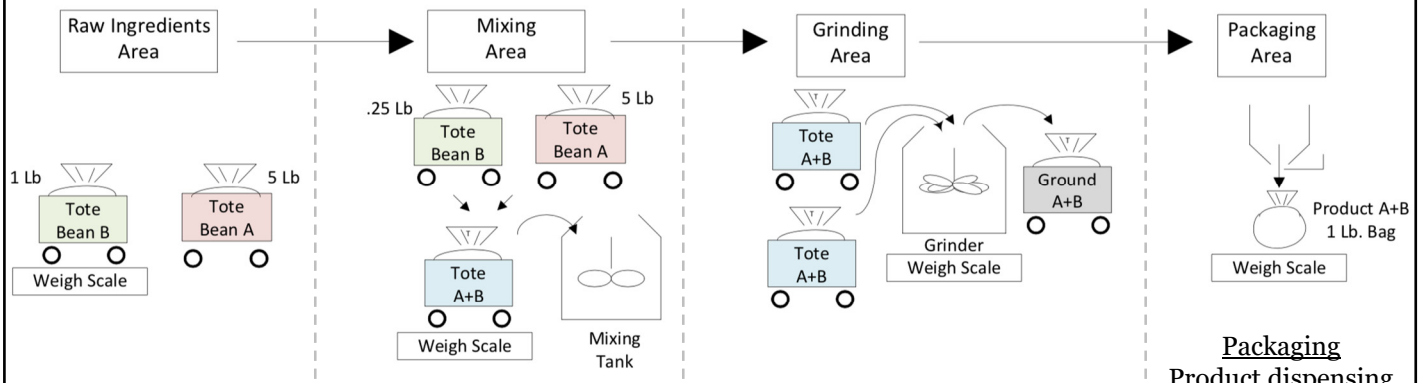
The tangible impact to a human asset can be immediately understood whether this is related to the individual's well-being or the individual's contribution to the operation. There is an intangible impact associated with a human asset that is difficult to identify when considering the type of impact that occurred, the mitigation the organization has in place, and the long-term well-being of the individual.

Perspectives on Value

- Attacks on an ICS are not always about gaining control of the system itself, but about using that access to strategically support external motivational factors of the adversary such as geopolitical or economic events
- Consider different perspectives in the evaluation of value to minimize criticality of asset
- Organizational Value
  - Straightforward and should be well understood
  - Includes failures, misconfigurations, and human errors
- Adversarial Value
  - When aligned to the organization value, straightforward and well understood
  - When not aligned to the organization value, more abstract

Not every threat group intends to just gain a foothold into the environment and attack a specific component; some threat groups build an attack plan in order to cause a desired effect on the process. The outcome could be realized in multiple ways, including quality or availability problems with products or services, creating a hazardous or unsafe condition, causing an environmental disaster, etc. Industrial control systems are engineered in the context of producing a product or service with an assumed level of user and system trust within the production area. An adversary may abuse this trust to create an undesirable situation where the process operates outside the engineered specifications.

Covfefe Coffee Factory Perspectives on Value



Raw Ingredients
No ingredients means no input to product process! This risk can be managed through larger inventories

Mixing
Inability to mix could impact certain hybrid product lines. Manage multiple mixing lines or excess capacity of mixed product

Grinding
No grinding means no output product for packaging and distribution! Manage multiple grinding stations.

Packaging
Product dispensing and packaging are critical from a QC perspective. Manage through multiple manual pack stations

Impacts on each operating area need to be assessed and understood from a quantitative or qualitative perspective in order to develop an effective mitigation approach, which may include design changes, security improvements, or potentially transferring risk through insurance or third-party contracting approaches.

Categorization of Critical Asset

- Operation Area Identifier
  - A common identifier known by the organization associated with where the asset resides (i.e., physical or logical)
- Asset Type / Name / Description
  - A commonly known identifier associated with the asset
- Business Role
  - A commonly known service function (value) the asset provides to the organization
- Concern/Impact
  - Listing of direct and indirect losses associated with the asset

This process of identifying the most critical assets within an organization can be challenging and can be run from a variety of different approaches. The process has been summed up with a phrase "everyone's baby isn't pretty," meaning there are some elements within your operating environment that are more important than others, and for mid- to large-size organizations, some operating environments are more important than others. On an even larger scale, for sectors and countries, there are more important areas or assets servicing the public good or military strategy or economic structure that would have a larger impact if unavailable or damaged. A defined process within your environment is necessary so you can prioritize response and allocate resources appropriately.

Our Critical Asset

Operational Area	Asset	Business Role	Concern / Impact
Plant Floor	Weigh Scales	Provides accurate measurement for material usage and product QA	Failure – Likely production impact Compromise – Likely QA impact Minimal safety hazard
Plant Floor	Mixing Tank	Provides the capability for blended product	Failure – Results in impact to hybrid product availability
Plant Floor	Grinder	Converts product into final form as ground beans for packaging	Failure – Results in potential loss of product for sales
Plant Floor and Remote	OT Assets	Utilized to operate, monitor, optimize, and automate tasks in the process	Failure – Would result in variable cost increases due to additional manual labor tasks and expenses
Infrastructure Components	IT Assets	Leveraged for billing, scheduling, access control, and management visibility	Failure could impact daily operations and create operational outages, with additional variable costs for local operators or expanded remote visibility

This page intentionally left blank.

Security Zones and Conduits

- Definitions Reference: *IEC 62443-2-1*
- **Security Zones** – Grouped assets with "Common security levels in order to manage security risks to achieve a desired target security level"
- **Conduits** – These "connect the security zones and facilitate the transport of necessary communications between the segmented security zones"

The IEC 62443 set of standards provides a common language to reference when discussing technical implementation concepts. Without the standard language available many different sectors or geographies would frequently discuss groupings of assets with terms like enclaves, segments, networks, environments, levels, etc. The communications between those groupings would sometimes be referenced by the communication type like serial, routable, 4–20ma, bus, or trusts. IEC 62443 allows us to discuss the groupings as Zones and the communications between those Zones as conduits.

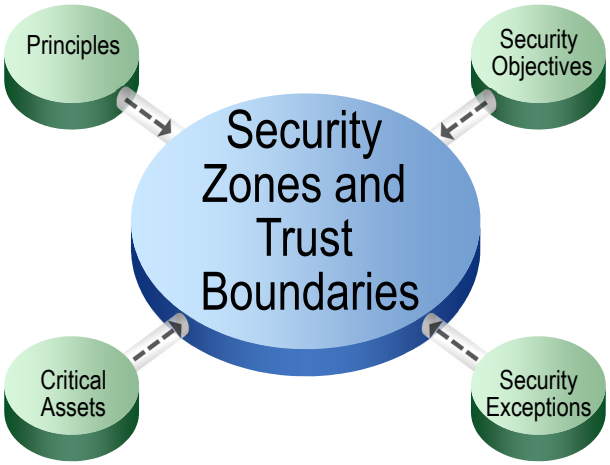
Trust Boundaries

- Conduits describe the means of whether and how zones reach each other.
- **Trust Boundaries** – Extend the function and security of conduits through a measurement of trust
  - Describes each unidirectional and bidirectional trust relationship
  - Defines the governance on how network operations and security controls operate
  - Used to draw the electronic fence around the zone

When examining the identified, allowed, and required conduits that exist there is an additional step required: Any communications that occur need to be controlled as strictly as possible without impacting operations, but security restrictions need to be in place to limit misuse of trusted communications paths where capable. The end result may include a variety of different layered approaches that utilize distinct technologies based on need and a defined level of trust. Some communications may leverage a unidirectional gateway or a proxy node, while others may leverage ICS-aware firewalls, and yet others may rely on traditional firewalls or routing infrastructure. Each solution and approach would be specific to the communication need.

Identifying Security Zones and Trust Boundaries

- Principles
  - Operational relationships and dependencies
- Critical Assets
  - Direct or indirect loss
- Security Objectives
  - Common security levels and controls
- Security Exceptions
  - Common deficiencies and weaknesses

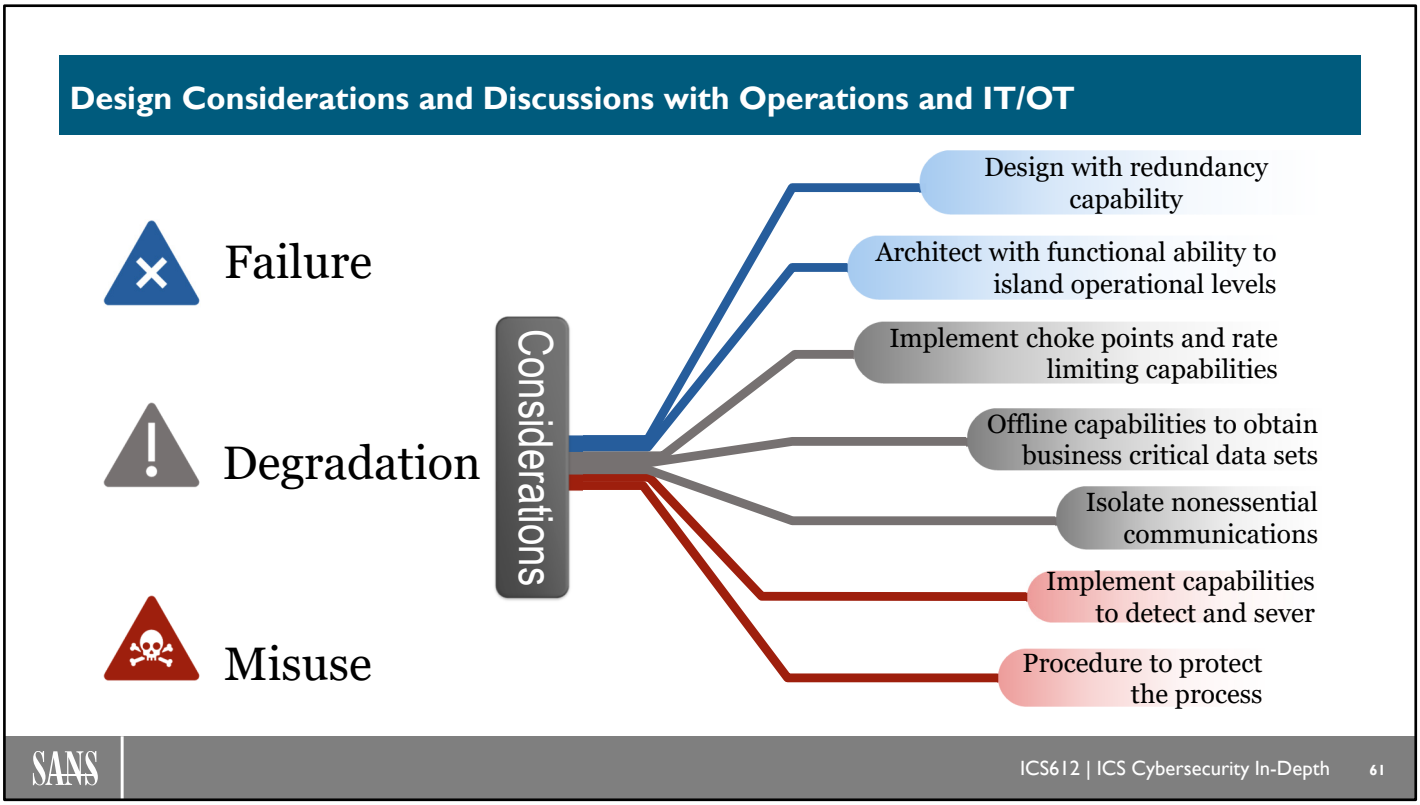


Utilizing the security design approaches outlined above, you will leverage the appropriate security design principles, security objectives, and prioritized critical assets to manage risk in a manner that suits your business objectives. The key item to be aware of is the identified security exceptions: Where security exceptions are identified they need to be continuously monitored and verified with the vendor supporting the device. The exceptions need to be tracked and reassessed in a way that provides a mitigation framework approach to ensure that they are not simply forgotten about because they are classified as exceptions. A tracked list of exceptions can be a very useful tool during incident response activities and may very likely lead to necessary containment and eradication steps.

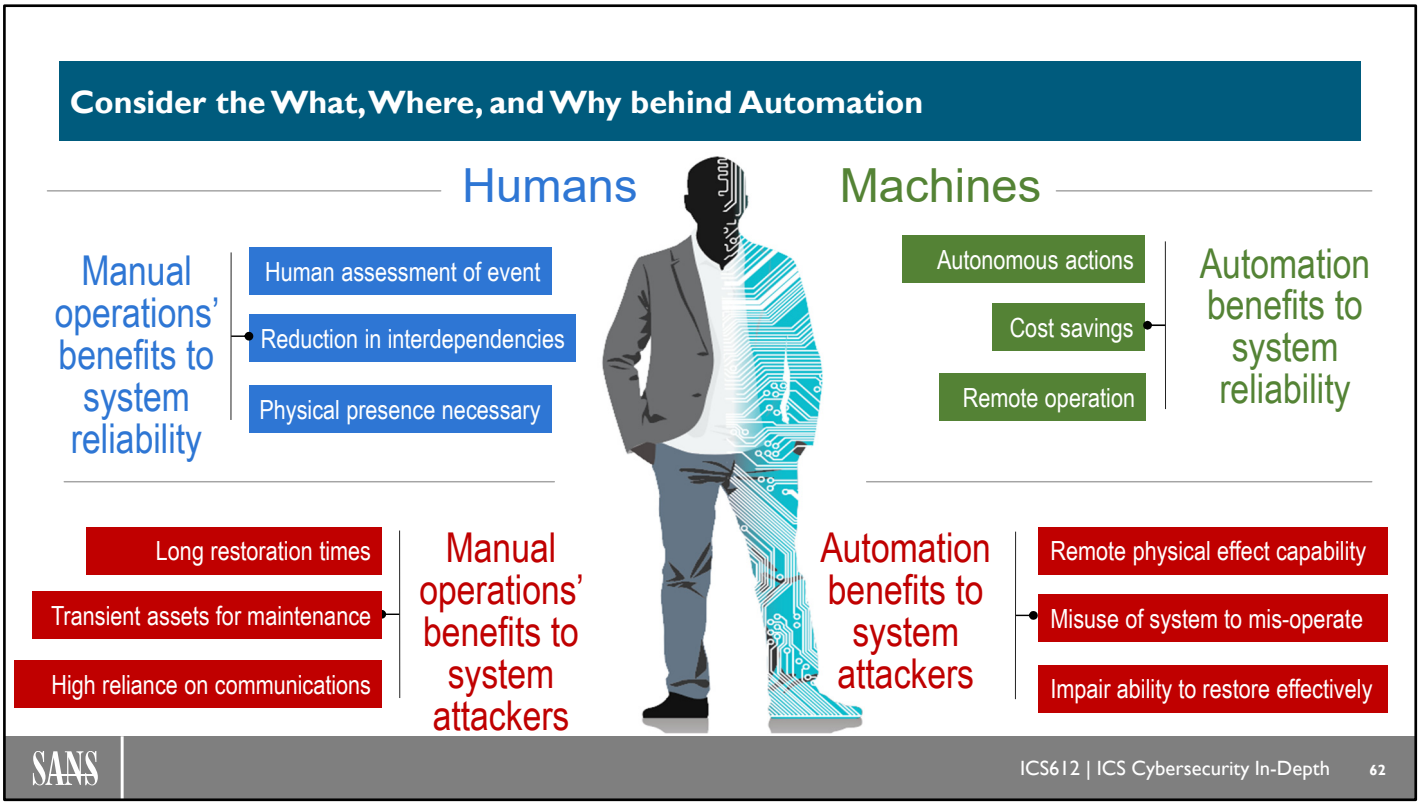
Principles

- Published principles are a "guiding light" for defining objectives and exceptions but they cannot chart the course as each organization is unique
- A security zone that is either too granular or too broad can have varying impacts on availability, integrity, and confidentiality
  - Many zones = increases in security complexity
  - Few zones = increases in attack surface
- Physical operations may not align with logical operations
  - Broad-reaching or overlapping control systems (e.g., plant operates on 1 PLC)
- Consider the cumulative mean time between failure (MTBF)
  - Addition of technology, or 'bumps in the wire', introduces additional risk
- Must understand all systems, their autonomy, their relationships, and their interdependencies

Published standards such as IEC 62443, ISA/IEC99 and NIST Cybersecurity Framework provide a guiding light for organizations to achieve an end goal of security. These standards, however, do not provide a step-by-step instruction or roadmap to an organization's specific needs. They provide a framework for an organization to begin the security journey and act as a reference for the organization as it derives its process of defining security objectives and exceptions.



As you architect your OT environment, consider how the system would be impacted by individual device failures, as well as how it would be impacted if individual device functionality was degraded. Most importantly, consider how the environment could be impacted if the individual devices and systems were all up and functioning, but they were being misused by an adversary.



When you consider the impact of an individual device(s) or system(s) failure, degrading, or being misused, you realize that your defense and risk management approach needs to take special consideration of the operational capabilities and components that are too important to fail. For such cases it is worth weighing the need for automated vs. manual controls. Consider how your implementation will benefit you as the asset owner and operator, but also keep in mind the benefits you are enabling for the adversary.

Security Objectives

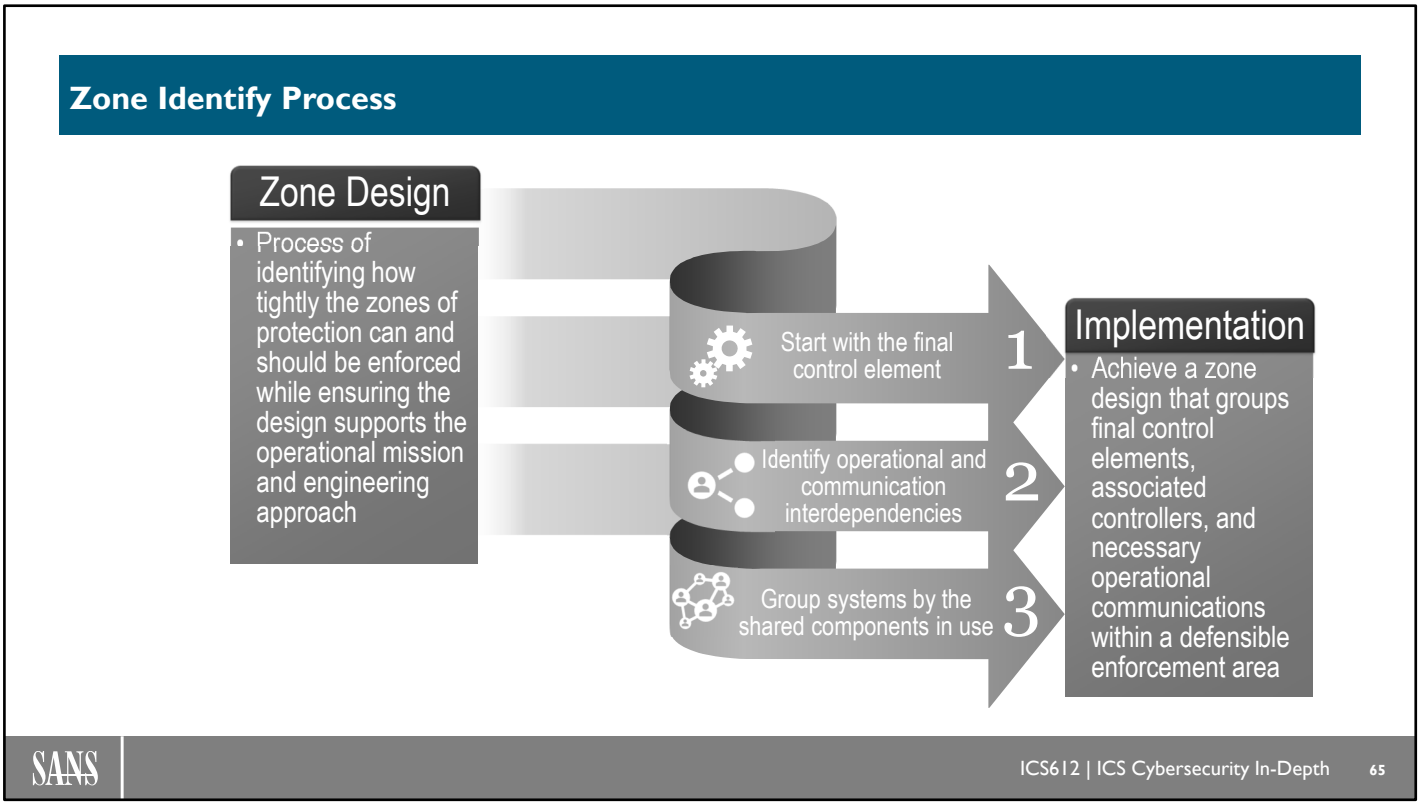
- Documented outline of what the security controls must support
- Common objectives include:
  - Environment and Personnel Safety
  - Access Controls
  - Remote Access
  - Vulnerability Management
  - Data Protection
  - Authorized Services and Interfaces
  - Network and Event Logging

You will need to scale your security objective goals based on what your operational risk tolerance approach will allow and based on where your current cybersecurity maturity level is. There are dozens of frameworks, approaches, controls implementations, and standards that exist for various IT and OT environments. Many of those approaches overlap with each other, and in some operating environments would be impossible or risky to implement. It is important that you assess your capabilities and bite off the level of controls and security objectives that have the greatest impact, can be achieved, and more importantly, can be maintained.

Security Exceptions

- Documented outline of exceptions that must be made against the security objectives
  - Can become requirements for future improvement projects and system upgrades
- Unsupported, or missing, features from the vendor that do not meet security objectives
  - May need to be supported by additional monitor/audit
- Common exceptions include:
  - Availability and integrity requirements
  - Legacy components
  - Legacy protocols

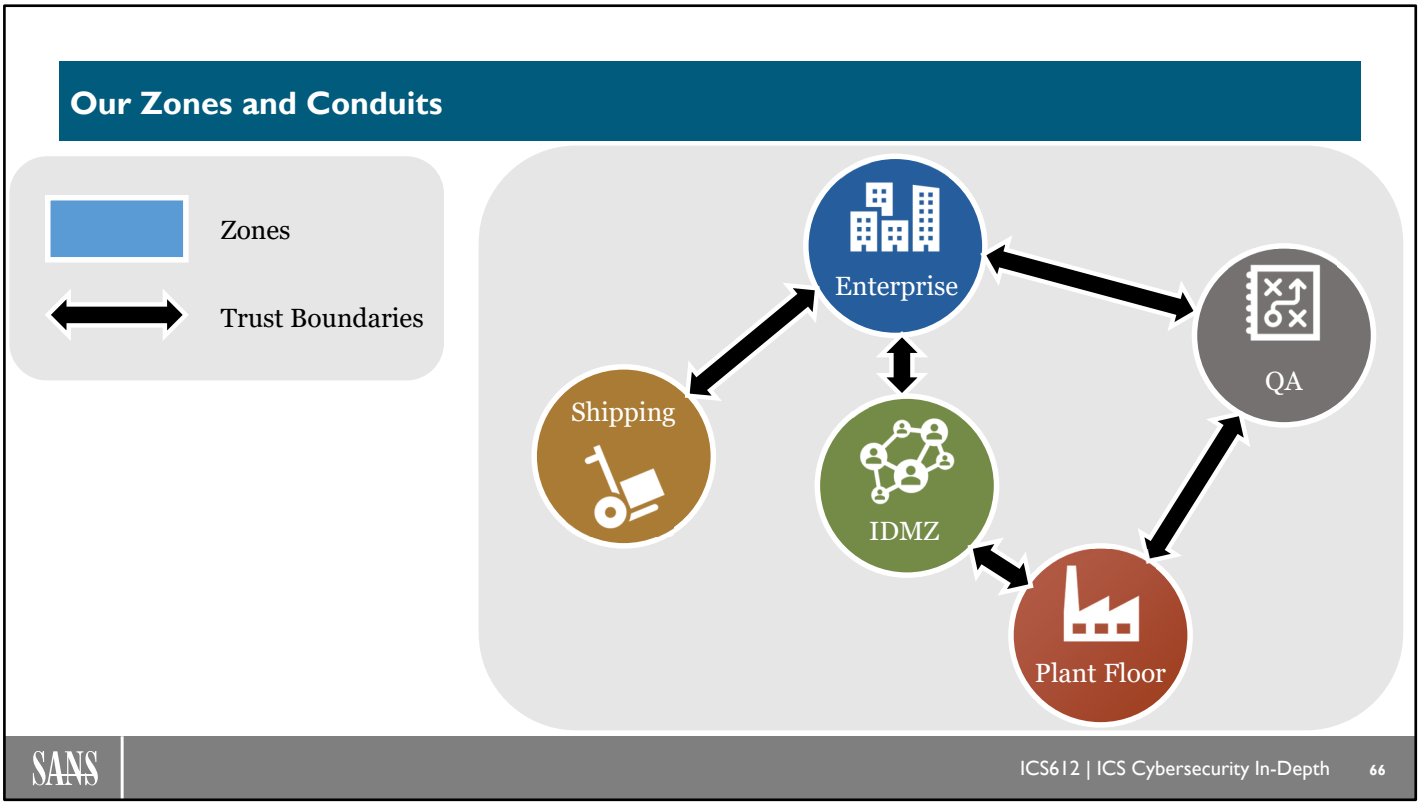
It is important to remember that the devices in use will often have some identified security exceptions and as you work with vendors to resolve or develop compensating measures, keep in mind that the vendors have built and designed these products as operational assets. By design, the purpose of the operational asset is to ensure reliability and safety with every available processor cycle and scan. These operational assets are now being expanded to include security controls that asset owners can use if their environment allows for it. For the identified security exceptions that exist, you may need to develop compensating measures that secure around the device for those specific capabilities that have not been included within the operational device from the vendor.



Sometimes a physical facility will produce multiple unassociated products that have clear physical boundaries and do not have any obvious operational interdependences such as shared equipment, lines, or automation. However, these facilities may share a warehouse and business systems. How are those systems then divided up to support the overall operational goals, communicate with shared ICS devices, and leverage common shared support infrastructure?

This process will help focus the relationship of systems while also identifying established trust boundaries.

1. Identify the lowest autonomous operational level of a critical system. This could be an individual automation component, or a collection of automation components coupled together to effectively perform a task with some level of autonomy. Defining autonomy is not exact, but it should consider an acceptable period of operation without oversight or external influence.
2. Identify the operational and communication interdependences between upstream, downstream, and supporting systems. Interdependences can include 'interlocking' operations; direct human or automated oversight; or indirect instruction to alter the operation of the task.
3. Group strongly related systems in a zone where those shared components exist with each other for operation.



The depiction of our classroom environment shows the defined zones and conduits communicating through the established trust boundaries. The trust boundaries that we created are defined by traditional firewalls.

Each defined zone needs to be considered because failure, degradation, or misuse within a lower trust zone could potentially impact higher trust zones through allowed trust boundaries. The moral of the story: Don't ignore any operational zone regardless of how unimportant you may perceive it to be, as you truly need to consider how a particular zone can impact other zones or the operational mission.

Engineered Zone Networks

- IT networks rely on a multitude of network services for the standard function of the environment
  - Some of these services rely on internet communication to function properly
- ICS networks also rely on many of these network services but are generally overlooked in the engineering of these systems
  - Some of these services can be avoided; however, most, if not all, of these services should not require external communication to function properly
- Engineering Zone Networks
  - Engineered with an anticipated level of autonomy
  - Both default and standard IT deployments and configurations, such as name resolution, must be reconsidered

Many ICS zones are defined based on operational need and interdependencies that exist through the defined conduits. In some cases, other dependencies are not fully understood or validated in regard to the impact that they could have if they were unavailable or degraded and they are frequently not considered at all from a misuse perspective. Another consideration to keep in mind is that performing the necessary level of analysis on external dependencies is an ongoing process throughout the life of the ICS assets. As updates and configuration changes occur, new applications are added that may require external service dependencies. Also, additional assets may be engineered into the environment for process improvement purposes, which could impact the overall operation if not fully accounted for. Performing this task is not a one-time engineering analysis project; rather this is an ongoing managed process.

Host Name Resolution

- The use of a Domain Name System (DNS) is the de facto standard for the internet and within IT networks to resolve host names to logical network IP addresses
- Since the origins of DNS, the default deployment configuration anticipates the necessity to resolve any address or forward resolution requests to upstream DNS servers
- ICS networks have adopted this technology for its simplicity in dealing with ICS applications as well as the greater adoption of IT technology that relies on name service
  - Many ICS applications use hardcoded IP addresses

- **Problem:** DNS is well known for abuse by malware and command and control
- Unlike in IT networks, name resolution within ICS networks is deterministic
- Engineer name resolution to meet the basic needs and reduce the risks
- Sinkhole all DNS in an ICS as an early indicator of incidental malware

Name resolution in ICS environments can be complicated and could have serious impacts on the operations environment if not architected and accounted for correctly. It is important to understand how each ICS device performs name resolution, the resolution methods used, timeout parameters for non-response before moving on to secondary or tertiary resolution methods, resolved asset cache timeouts or refreshes, as well as how they fail when a name is resolvable, but the service is non-responsive. In a number of critical asset hosts or nodes you will find that local host files are used for static entries of essential communications nodes; however, as assets are joined into external service and support environments like Active Directory, you may find it challenging to operate with local host files in an ongoing manner for name resolution. Where DNS use may be necessary on some hosts and workstations, DNS reliance should be avoided on individual ICS devices and components.

Engineered Trust Boundary Networks

- Internet edge network communications are only blocked in one direction
  - By default, all communications (destination and protocol) are allowed outbound to the internet, but communications inbound from the internet are blocked by default
  - The variability of users
- Zone network communications are blocked in both directions
  - By default, all inbound and outbound communications are blocked by default
- Engineered Trust Boundary Networks
  - Both default and standard IT deployments and configurations, such as routing, must be reconsidered

As we cyber-engineer our industrial control systems and work to develop teams of capable cyber operators who have the responsibility of providing a safe and reliable infrastructure in which process engineers and system operators have the ultimate responsibility of ensuring a safe and reliable operating environment, we are increasingly finding ourselves in a position where cyber operators need to fully understand what normal communications should look like for a particular trust boundary. When an established trust boundary is defined and the communications in and out are controlled, you can begin to expect trained cyber operators to identify potentially malicious or abnormal communication. After certain suspicious communications have been identified, cyber-mature organizations will develop playbooks and actions for the cyber operators to take. Ultimately there will be a time when cyber operators and system operators are working together and routinely communicating to jointly ensure the reliability, safety, and security of the operation.

Default Route

- Many network best practices emphasize the use of a default route for routers to ensure basic routing capabilities for all unknown routes to a destination
- Since network paths in an IT network and internet are dynamic, packets must be successfully forwarded even when no specific route can be determined
- ICS networks have adopted this technology for its simplicity in dealing with segmentation and for its simplicity in dealing with adoption of cellular and other public networks for operations
  - Blind adoption is a source of internet-accessible control systems

- **Problem:** Malware and other threats rely on the fact that a routable network path always exists between networks
- Unlike IT networks, routes within ICS networks can be engineered
- Engineer static routes only as required or anticipated
- Sinkhole default routes from the ICS as an early indicator of an attack

Default routes support the process of moving a communication along in an effort to get the traffic to its destination even if you do not know where the destination is. Where traditional IT communications are very much non-predictable and nondeterministic, we look at ICS communications in a different light: They should be fairly predictable and deterministic. With that understanding, it should not be a typical expectation to simply move unidentified communications from within an ICS space out to an unknown destination requiring default routes or likewise from an unknown location into an ICS zone.

Lab 2.4: Network Infrastructure Configuration

Go to the Lab Workbook: Lab 2.4

This page intentionally left blank.

ICS Secure Architecture Checkpoint 2.4

- Understanding what communications paths are essential to a safe, reliable operating environment is necessary to configuring your network segments and enforcement zones
- Network Zone Configuration
  - Review configured DNS entries, forwarding capabilities, and local DNS cache
  - Configure host file in lieu of DNS config on student win image
  - Configure NTP in firewall; direct student win image to pull from FW
- Trust Boundaries Network Configuration
  - Disable default route in firewall
  - Configure static routes
  - Block all two-way traffic

This page intentionally left blank.

ICS612 Section 2 Outline (4)

- Head End Process Overview
- Lab 2.1: Connect Pods to Level 3 Infrastructure
- Lab 2.2: Remote I/O
- Lab 2.3: Validate Functionality
- ICS Secure Architecture
- Lab 2.4: Network Infrastructure Configuration
- Process Communication and Data Flow Mapping
- Lab 2.5: Map Communications for the Environment
- Lab 2.6: Configure Connections to Process Visualization
- Local Attacks and Process Manipulation
- Lab 2.7: PLC Device-Level Attack
- Lab 2.8: OPC Discovery Attack
- Lab 2.9: Local Network MITM Attack

This page intentionally left blank.

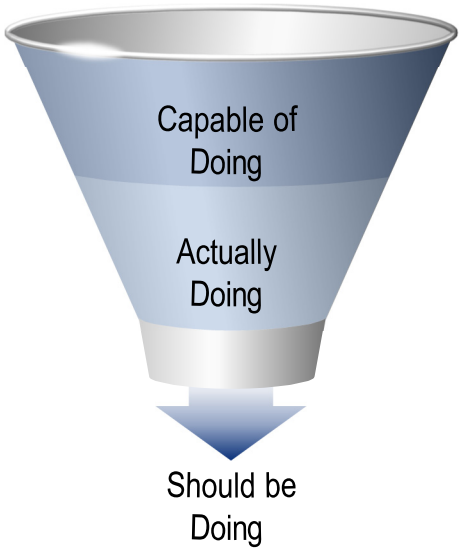
Process Communication and Data Flow Mapping

Student Kit Communications and Protocols
Local Student Pod Data Flow
Head End Equipment Communications
Process Component Communications

This page intentionally left blank.

Value of Communications Mapping

- Understanding beyond network diagrams
- Identify all available services, not just those associated with operations and operational data
- Understanding network attack surface
- Disassociating confirmed trust from blind trust
  - Capable: Vendor-provided communication features
  - Actual: Actual communications in use
  - Should: What is actually needed for operations (critical and non-critical)



Identifying and lowering the attack surface of a network and assets requires a study of the available services at different postures within and outside of a zone. Within a zone, the ICS operations staff typically have a general understanding of the types of services communicating; however, knowing exactly what is available requires either working with the vendor or performing a network study on the networked assets themselves.

In many cases there will be a number of services available but not in use within the network. Additional work with a vendor and a thorough study, or continuous ongoing study, of network traffic can reveal services that are actually in use. This list of services can finally be reduced to the minimal number of services that must be available. This approach can be used against both the individual assets operating in the zone as well as the individual communication conduits available between the zones. The attack surface can then be reduced by either disabling the service on the host applying additional mitigating controls to limit access to those services.

See Things as the Data Flows

- See things not by how they are connected (i.e., networked) but by how and why they communicate

- Data relevance
  - **Direct Relevance**
    - Data that has a direct operational dependency
  - **Indirect Relevance**
    - Data that has an indirect business need

Understanding the data flow helps to understand the operational requirements for the data as well as what exposure levels exist for the data as it traverses throughout the network. In general, the organization should have a business justification for the creation and movement of data. However, not all data is valuable to all adversaries to meet their specific objectives. For example, data read from a PLC that displays the count of the products produced over the last hour has an unlikely value to cause physical damage to the machine. Lastly, the direction the data flows does not always follow to the direction of which communication is initiated.

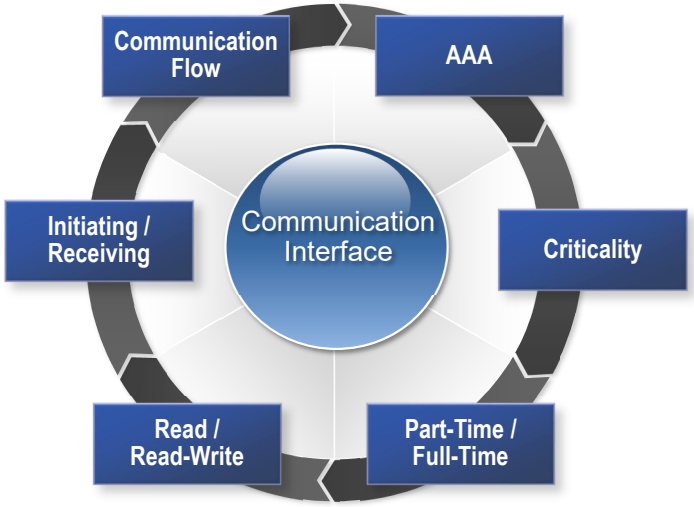
Communication Interfaces

- Used to describe each interface for evaluation of necessity and security capabilities/weaknesses
- Discovered through vendor documentation and NMAP
  - NMAP should only be used in offline environment like a lab

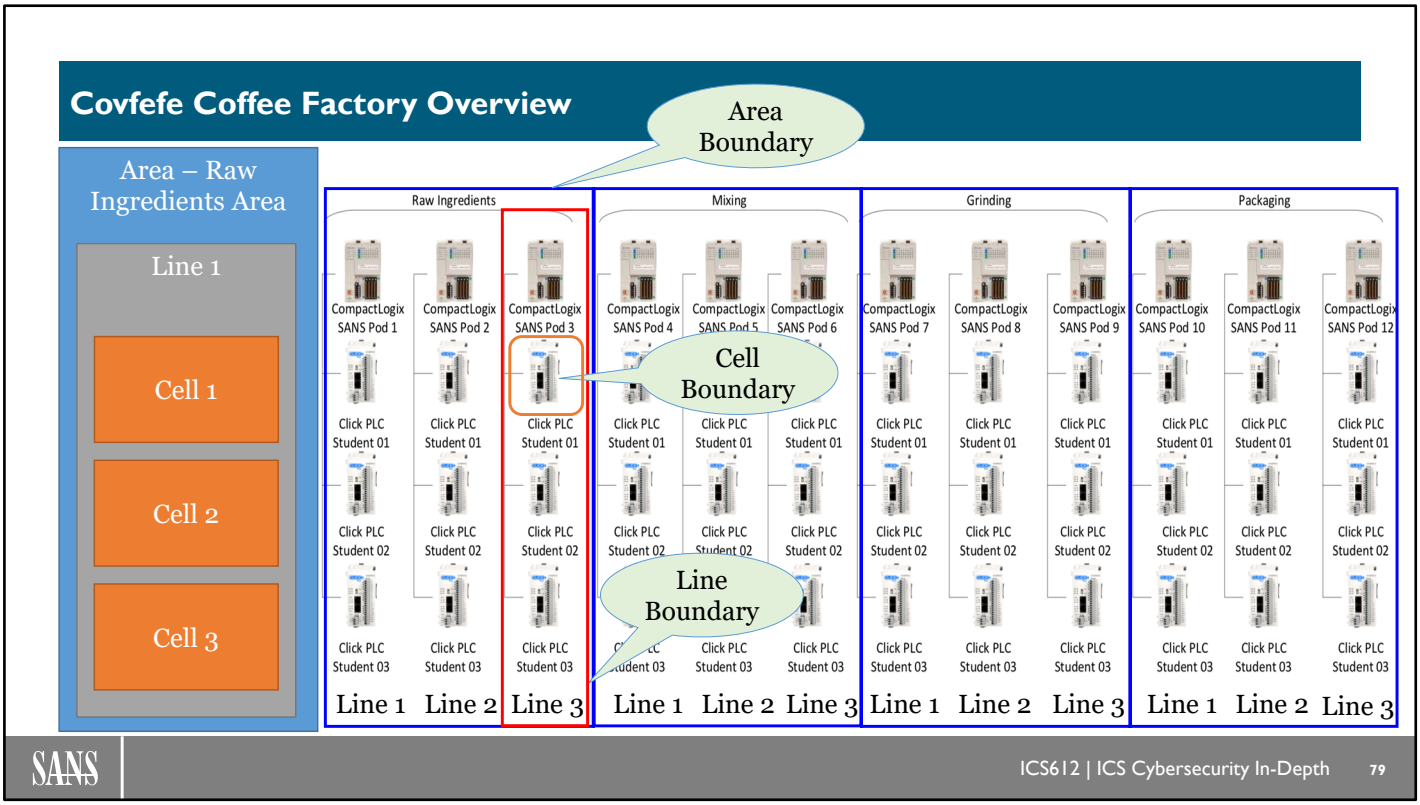
- User Interface
  - Direct user GUI interaction (e.g., HMI)
- Data Interface
  - Data service interaction (e.g., data server, etc.)
- Maintenance Interface
  - Direct user or data service interaction (e.g., EWS, telnet, backup system, etc.)

All communications can be categorized under three fundamental interfaces. The user and data interfaces are usually, or mostly, in constant communication. The maintenance interface is usually the most difficult to track and to correlate relationships with. Maintenance interfaces are not used unless required and large periods of time (e.g., years) may pass before the interfaces are accessed. This makes establishing a baseline communication for maintenance activities much more difficult to predict. The primary reason for this is because of the variety of applications, protocols, and methods provided by ICS vendors. Even when working directly with the operations team to walk through all of the potential scenarios they may use to perform maintenance, many communications will be overlooked. Finally, it is important that when performing network security monitoring of an ICS network to not simply whitelist maintenance communications as they are reviewed and validated. Since this activity is sporadic, it is best to continuously track and verify this activity as it occurs following a trusted and expedient workflow.

Parameters of Communication Interfaces

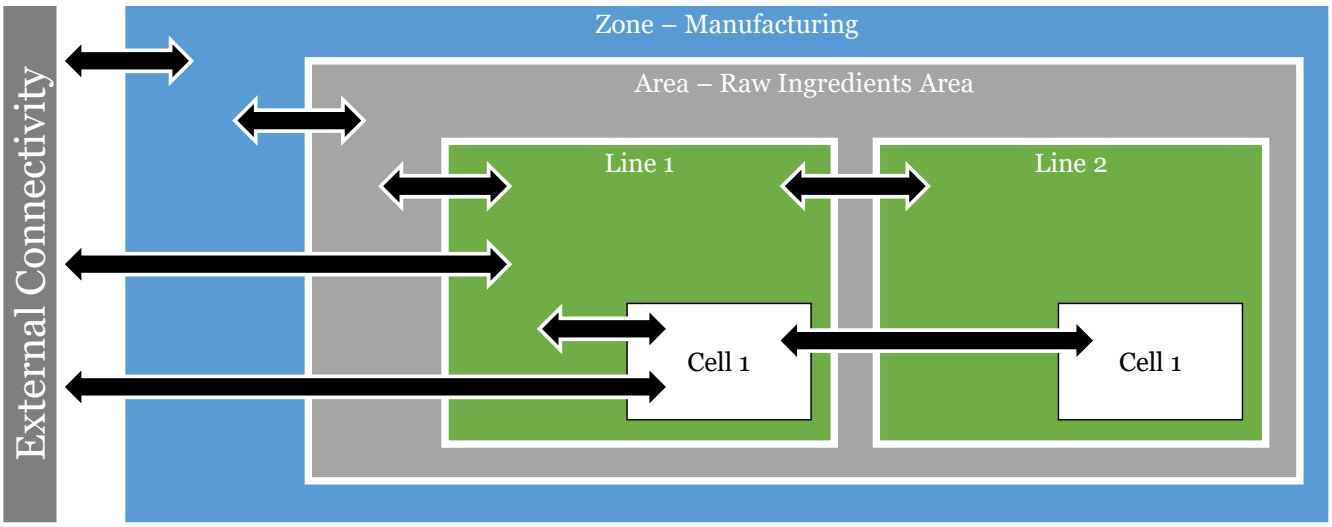


When recording strategic or critical communication interfaces you should record the following parameters as they relate to your operation: AAA (authentication, authorization and accounting); criticality; part-time/full-time; read/read-write, communication flow, and initiating/receiving. These parameters will help in the selection and implementation of security controls to protect the access to the interfaces, the protection of the communication, the integrity of the data and the asset, and the criticality used for manipulation during incident response activities.



This page intentionally left blank.

Communication Interfaces



This page intentionally left blank.

Lab 2.5: Map Communications for the Environment

Go to the Lab Workbook: Lab 2.5

This page intentionally left blank.

Process Communication and Data Flow Mapping Checkpoint 2.5

- As a process environment expands, is connected to, and dependent on more and more subsystems, it is necessary to update data flow documents to reflect all communications across the various network levels
- Consider how the process will be impacted if network changes are made, if infrastructure devices fail, or if an adversary were to misconfigure a device or misuse a necessary service.
  - Consider each device and communication flow and the operations components they support
  - Now consider if the device or communication flow was unavailable, degraded, or misused

This page intentionally left blank.

Lab 2.6: Configure Connections to Process Visualization

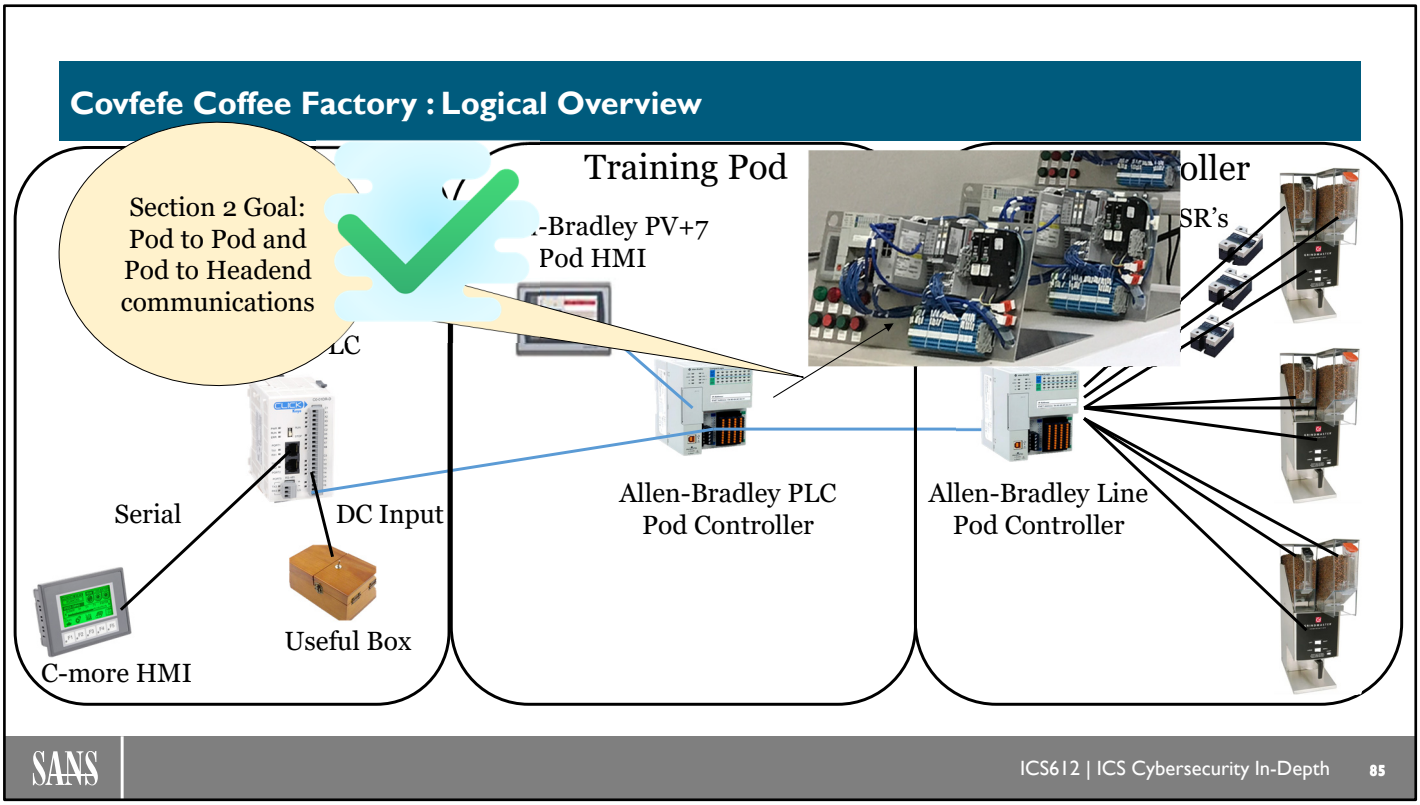
Go to the Lab Workbook: Lab 2.6

This page intentionally left blank.

Process Communication and Data Flow Mapping Checkpoint 2.6

- Operators make decisions routinely based on information provided to them through tabular displays of data driven graphics.
- For interconnected and interdependent systems, it becomes quite difficult for operators to have the ability to see the wide area view of the overall process and individual devices in a manner that helps them make rapid decisions
  - Taking display values from lower-level field controllers and mapping those tags into a common display can be very helpful to operators
  - Pulling in alarm tags and allowing remote response capabilities for an operator may be necessary depending on your business demands

This page intentionally left blank.



As we look at Levels 0 and 1 hands-on exercises, we will use two systems to achieve the PLC and HMI learning objectives. In the center of the slide, you will see the training Pod hardware that consists of an Allen-Bradley PanelView HMI and the Allen-Bradley (A-B) CompactLogix PLC. The training Pod also contains push buttons, indicator lights and remote breakers that the A-B PLC will use for input and output control.

The student kit as shown on the left contains the Click Plus PLC and the C-more HMI that will be used during student labs. The Click Plus PLC will communicate with the A-B PLC via Modbus TCP sharing data register information and I/O status. The student kit also contains a Useless Box that will be transformed into a Useful Box that will be controlled by the Click Plus PLC in order to show you how "useful" a simple input switch, motor circuit, and power source can be to gain knowledge about PLC systems. The student kit also contains a K-type thermocouple to demonstrate analog input capabilities of the Click Plus PLC.

ICS612 Section 2 Outline (5)

- Head End Process Overview
- Lab 2.1: Connect Pods to Level 3 Infrastructure
- Lab 2.2: Remote I/O
- Lab 2.3: Validate Functionality
- ICS Secure Architecture
- Lab 2.4: Network Infrastructure Configuration
- Process Communication and Data Flow Mapping
- Lab 2.5: Map Communications for the Environment
- Lab 2.6: Configure Connections to Process Visualization
- Local Attacks and Process Manipulation
- Lab 2.7: PLC Device-Level Attack
- Lab 2.8: OPC Discovery Attack
- Lab 2.9: Local Network MITM Attack

This page intentionally left blank.

Local Attacks and Process Manipulation

Targeting ICS Components
Targeting ICS Protocols
Targeting ICS Network Infrastructure

This page intentionally left blank.

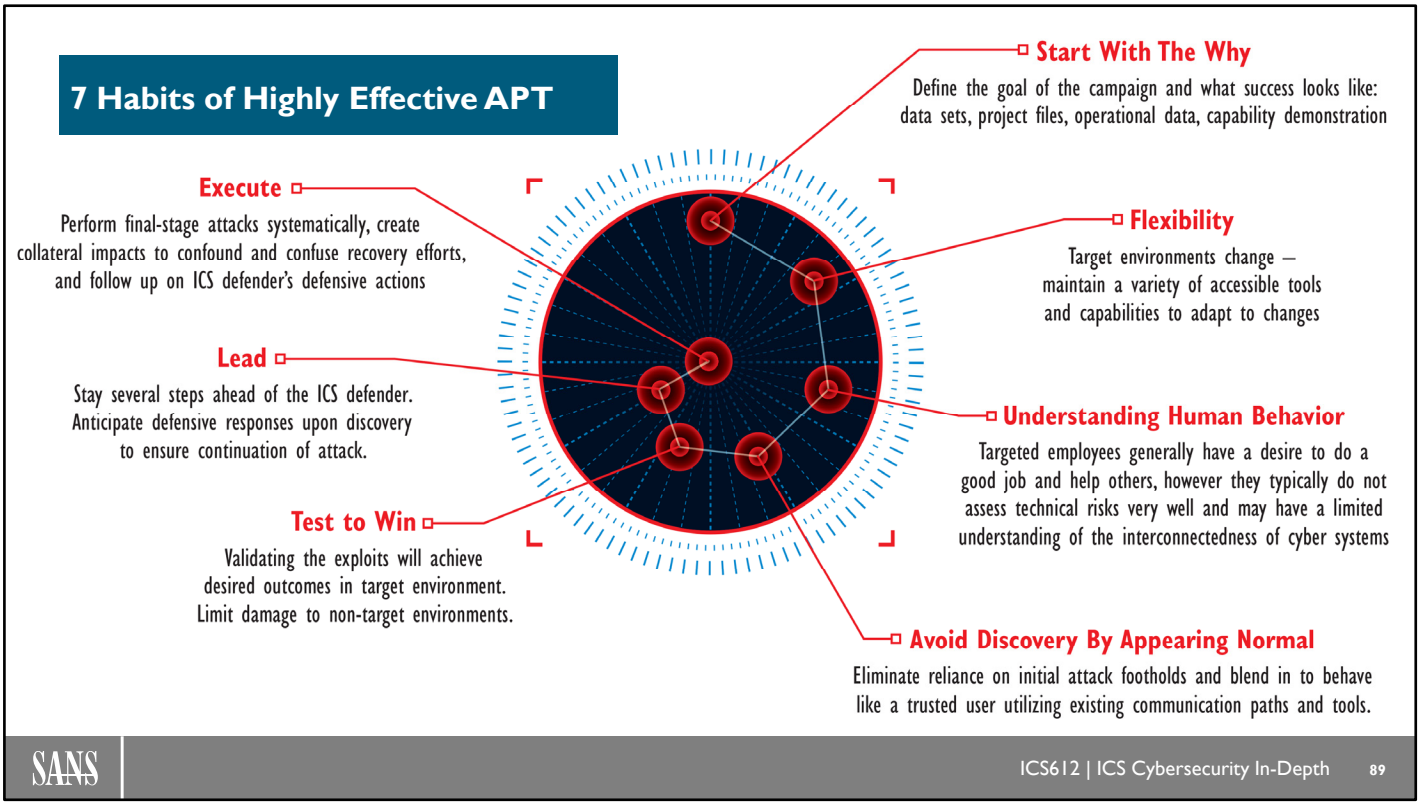
Attack Vector Selection

Consider how an adversary develops an attack

- Attack selected based on available vulnerabilities – "Targets of opportunity"
- Attack approach selected due to an objective to be in the environment for a very long time, relying on local tools and capabilities to reduce the chance of being detected – "Living off the land"
- Attacker intent is to demonstrate a capability and willingness to leverage the capability – "The deterrent"

What do you believe the attacker goals were in Ukraine?

Without having access to an attack team for interrogation and questioning, it can be challenging to try and determine the ultimate goal of an adversary's actions. Oftentimes we speculate about an adversary's goal, but we could learn more by gathering information about the attack. What did the attacker target? Did they target other organizations? How did they gain access? How long did they have access? Based on what they did, what may the goal have been? What could they have done with the positioning that they demonstrated?



Targeted attacks from well-funded nation states (APT Advanced Persistent Threats) that target operational technology environments typically are evidenced by adversary groups that show some level of coordination and direction from a commanding officer. Consider their formal strategies and approaches depending on the overall tactical and strategic objectives.

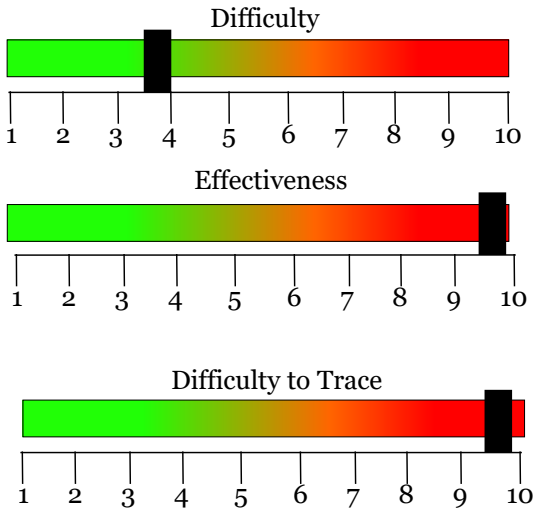
Device-Level Attack Goals

- Intellectual property theft – save on R&D
- Understand operating environment completely to enable an attack with an effect later
- Utilize the device as an attack launch point
- Supply chain attack with capabilities for later use
- Deliver an attack that impacts operations directly
- Combination attack that causes damage

As an adversary decides on an attack vector of choice, the approach taken will largely be driven by the overall concept of operation that is developed to achieve a goal. Targeting ICS devices will send a big message to the ICS community: That a particular attack or adversary group has specific capabilities that could impact process environments in multiple geographies and across multiple sectors. We must also consider an adversary's willingness to expose the community to specific device-level attack approaches and exploits. Typically when an exploit or attack approach is discovered in the wild, there will be a rapid response from the vendor community, and in general there is typically always a slow mitigation rollout effort by the asset owners when it comes to device-level attacks.

Works as Designed but Not as Intended

- Sensor calibration or setpoint manipulation
- Intention: Let the system work against itself
  - Works as designed but not as intended
- Very hard to detect a data table change
- In most cases, PLC programmers have not limited high/low limits



A control system such as a PLC or PAC is designed to control a process through the means of code execution. The PLC or PAC will read sensor inputs, execute the code based on these values, and then set the output values. The inputs and outputs could be discrete, analog, or internally declared variables.

Typically an analog input sensor or an analog output to a multivariable valve is not linear and requires calibration, especially if process accuracy is critical to producing a quality product. If we manipulate either the calibration variables or a critical operational setpoint through the data table, it can directly affect the process and thereby affect product quality. Imagine this scenario has just occurred in a food or beverage process.

Also, data table manipulation is quite hard to trace as the data table can be constantly changing. Some automation vendors do allow for data table changes to be tracked but it is often not a "real-time" alarm.

Lab 2.7: PLC Device-Level Attack

Go to the Lab Workbook: Lab 2.7

This page intentionally left blank.

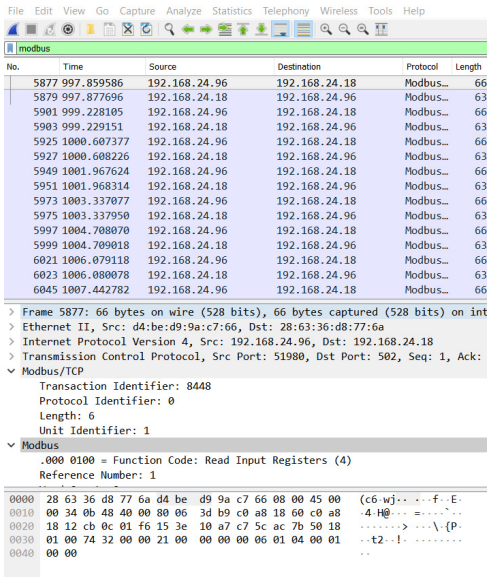
Local Attacks and Process Manipulation Checkpoint 2.7

- With the network communications paths in place, consider what an adversary can achieve at a local-device level
- As you examine attacks targeting local devices, ask the following:
  - What did the attacker achieve?
  - What information did they need to achieve it?
  - Where could they obtain the information?
  - What access would they need to perform the attack?
  - Can they perform the attack again with success?
  - How could you have detected it?

This page intentionally left blank.

Protocol-Level Attack Goals

- Collect traffic in an effort to understand the environment
- Attempt to locate a protocol that can traverse zones of trust
- Use operational protocols to move data into or out of an OT environment
- Use operational protocols to map the assets within the environment
- Utilize operational protocols to hide in the normal for extended periods of time



Think of attackers who choose to target the communications protocols like visitors to your community who want to commit a crime. They are working to learn the local language in the foreign land that they are visiting, and they wish to understand the normal operating environment. This way they can blend in and understand how to achieve specific goals in the environment that they are in without being detected.

OPC

- OPC is the interoperability standard for the secure and reliable exchange of data in the industrial automation space and in other industries
- Platform independent and ensures the seamless flow of information among devices from multiple vendors

Object Linking and Embedding (OLE) for Process Control, or more commonly known as OPC, became popular as customers wanted data and commands to interoperate between different vendor control systems and HMI packages. While the ease of use for connectivity exploded with vendors embracing the OPC standard, this opened the door for unintended clients to read and write any control system that supports OPC.

Reference:

- https://opcfoundation.org/about/what-is-opc/

OPC Classic

- Microsoft Windows Only
- Uses COM/DCOM (Distributed Component Object Model) to exchange data
- Not firewall-friendly because of dynamic port assignments
  - However, some vendors have developed a 'tunneling' capability to better support firewalls
  - Firewall vendors have evolved to provide better support for dynamic port assignments
- In an attempt to get communications to work across different OS versions and vendor applications, many integrators have configured network-level permissions to 'everyone'

Most control system vendor tools operate on the Windows platform because of the wide home and industrial adoption of this platform. While not entirely without merit, plant operations personnel are more likely to have familiarization with the Windows OS and therefore are more likely to be able to support the Windows OS at the plant. This has proven to be the most popular OS that supports control system design tools.

OPC in its original incarnation is COM/DCOM based, which can provide challenges for getting the technology to work. In a lot of installations, the COM/DCOM settings are set to allow 'everyone" in order to get OPC running.

OPC does present challenges for setting an OPC client to communicate with the OPC server through a firewall because OPC uses dynamic port assignments. Some firewall vendors have addressed this behavior by creating a firewall rule that tracks the dynamic port assignments and dismantles the session once communications have been completed.

Reference:

- https://opcfoundation.org/about/opc-technologies/opc-classic/

OPC Unified Architecture (1)

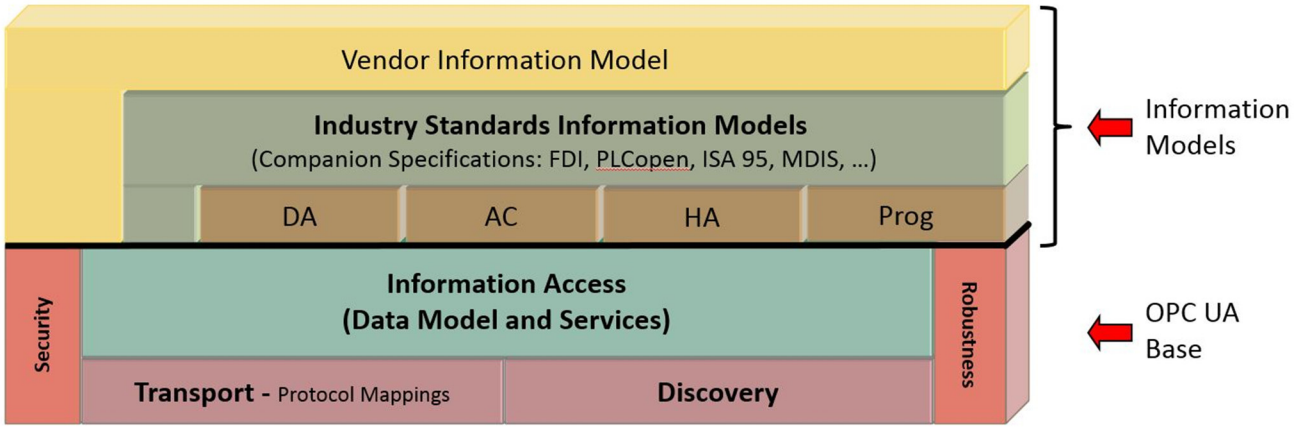
- **Functional Equivalence**
  - All COM OPC Classic specifications are mapped to UA
- **Platform Independence**
  - From an embedded micro-controller to cloud-based infrastructure
- **Secure**
  - Encryption, authentication, and auditing
- **Extensible**
  - Ability to add new features without affecting existing applications
- **Comprehensive Information Modeling**
  - For defining complex information

OPC Unified Architecture (UA) has built on the success of OPC and addressed some of the original OPC shortcomings. For instance, OPC UA allows for encrypted communications, authentication mechanisms of an OPC UA client to the OPC UA server, and more robust auditing capabilities. In support of legacy implementations and legacy devices, it is very common to find OPC Classic still in use, as the features made available in OPC UA had to be downgraded in order to support the operating environment capabilities.

Reference:

- https://opcfoundation.org/about/opc-technologies/opc-ua/

OPC Unified Architecture (2)



In this OPC UA architecture model, we see the fundamental tenets of the entire specification. Data Access (DA), Alarms and Conditions (AC), and Historical Access (HA) are used by vendor software packages to subscribe to OPC UA servers. Underneath this model, one must remember that the OPC UA is fundamentally a communication stack that converts an ICS protocol to an OPC UA server.

References:

- https://opcfoundation.org/about/opc-technologies/opc-ua/
- https://opcfoundation.org/about/opc-foundation/history/

OPC UA – Additional Capabilities (1)

- **Discovery**
  - Find the availability of OPC servers on local PCs and/or networks
- **Address Space**
  - All data is represented hierarchically (e.g., files and folders) allowing for simple and complex structures to be discovered and utilized by OPC clients
- **On Demand**
  - Read and write data/information based on access permissions

OPC UA has not strayed far from the original OPC mission, that of discovering data sources, enumerating the address space of the OPC server, and interacting with the data through read and write commands. The idea of the interoperability of OPC is that one could query for data with the intent of subscribing to data of interest and perhaps writing new values to the original source if required.

Reference:

- https://opcfoundation.org/about/opc-technologies/opc-ua/

OPC UA – Additional Capabilities (2)

- **Subscriptions**
  - Monitor data/information and report-by-exception when values change based on a client's criteria
- **Events**
  - Notify regarding important information based on client's criteria
- **Methods**
  - Clients can execute programs, etc., based on methods defined on the server

As with the interest in any data source, creating a connection or a subscription to monitor the data is a fundamental requirement. OPC and OPC UA allow for a subscription model so the client can subscribe to a data source. In some cases, it is more efficient to receive the data only when it's changed rather than constantly polling the data source for a changed value. OPC and OPC UA support this functionality.

Event criteria can be configured and once a threshold or other criteria is met, then the client can be notified of this event. OPC UA also allows the client to interact with the server through functions located on the server.

Reference:

- https://opcfoundation.org/about/opc-technologies/opc-ua/

OPC UA Security (1)

- **Transport**
  - Numerous protocols are defined, providing options such as the ultra-fast OPC-binary transport or the more universally compatible SOAP-HTTPS, for example
- **Session Encryption**
  - Messages are transmitted securely at 128- or 256-bit encryption levels
- **Message Signing**
  - Messages are received exactly as they were sent
- **Sequenced Packets**
  - Exposure to message replay attacks is eliminated with sequencing

OPC UA adds various capabilities to secure the communication path, protect the data from modification in transit, ensure the integrity of the message, and prevent OPC data spoofing through replay attacks. Keep in mind however that the devices and the environment need to support the OPC UA protocol in order to no longer need the legacy OPC Classic implementation.

Reference:
- https://opcfoundation.org/about/opc-technologies/opc-ua/

OPC UA Security (2)

- **Authentication**
  - Each UA client and server is identified through OpenSSL certificates providing control over which applications and systems are permitted to connect with each other
- **User Control**
  - Applications can require users to authenticate (login credentials, certificate, etc.) and can further restrict and enhance their capabilities with access rights and address-space "views"
- **Auditing**
  - Activities by user and/or system are logged, providing an access audit trail

Additional OPC UA capabilities that provide access control, authentication elements, and an audit trail of events can be very useful to ensure the right users have access to the right data from trusted devices. Again, keep in mind these additional capabilities add infrastructure requirements and third-party application libraries that may introduce new vulnerabilities.

Reference:

- https://opcfoundation.org/about/opc-technologies/opc-ua/

Lab 2.8: OPC Discovery Attack

Go to the Lab Workbook: Lab 2.8

This page intentionally left blank.

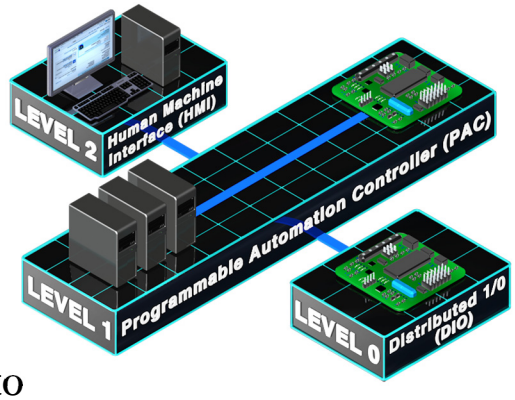
Local Attacks and Process Manipulation Checkpoint 2.8

- Performing attacks directly on local ICS devices or at the network level may not align with an attacker's objective. The attacker may wish to first fully understand the operations environment and verify the configuration to ensure a desired effect.
- An attacker may also wish to gain information from the process in order to steal the intellectual property or possibly leverage native communications protocols to maintain persistence.
- Using attacks on an ICS protocol, an adversary can gather sensitive data – IP addresses, device specific information, tags, tag values, user information, process state, and additional data.

This page intentionally left blank.

Network Infrastructure–Level Attack Goals

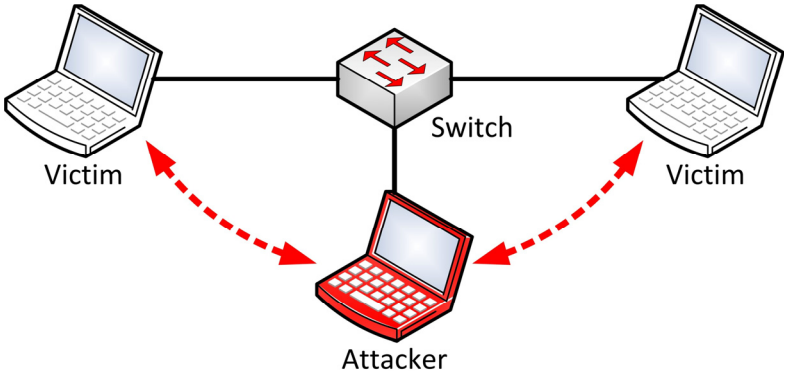
- Manipulate routes or ARP tables to gain visibility to communications
- Manipulate name resolution to modify network communications
- Implement or manipulate proxy servers to establish remote command and control capability
- Quickly impact communications capability to impact operations
- Modify operational data or replay manipulated data to impact operations



Attackers targeting network infrastructure to impact OT environments are likely achieving the goals defined in the operations development phase of an attack. Such attackers specifically leverage many traditional tools and frameworks as well as very basic custom tools that are often modifications of existing tools. This attack approach allows the adversary to retain more capable tools for later use, without exposing this capability to the community.

Machine-in-the-Middle Attack

A **Machine-in-the-Middle attack** (**MITM**) is an attack where the attacker secretly relays and possibly alters the communications between two parties who believe they are directly communicating with each other



A Machine-in-the-Middle attack can operate at multiple layers within the OSI model, however the attack depicted here shows a Layer 2 attack within a switched environment and can be very successful within a traditional flat ICS network implementation.

References:

- "Man-in-the-middle attack," Wikipedia, https://en.wikipedia.org/wiki/Man-in-the-middle_attack
- Image source: https://www.bettercap.org/modules/ethernet/spoofers/

MITM Example Techniques

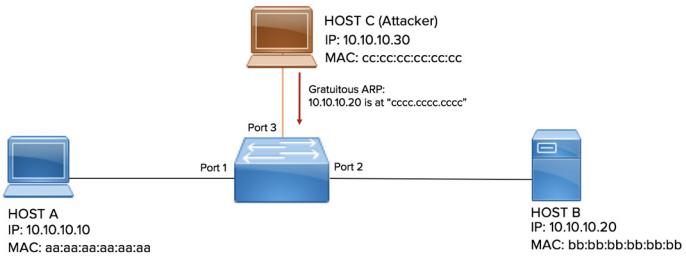
- ARP Spoofing
- DNS Spoofing
- Rogue Access Point / Evil Twin
- Proxy Server
  - Has legitimate uses

A variety of techniques exist that allow an adversary to get in the communication path between the source and the destination. The adversary uses these techniques to gain data visibility or manipulate the data being delivered.

ARP Spoofing

ARP spoofing, ARP cache poisoning, or ARP poison routing, is a technique by which an attacker sends (spoofed) Address Resolution Protocol (ARP) messages onto a local area network

– The aim is to associate the attacker's MAC address with the IP address of another host, such as the default gateway, causing any traffic meant for that IP address to be sent to the attacker instead



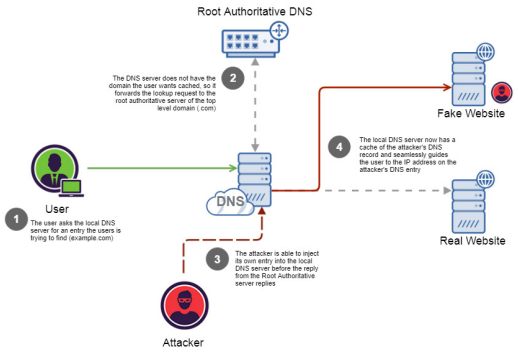
To forward a communication packet, switching infrastructure requires two pieces of information – the source and destination MAC address and the source and destination IP address. This information is stored in the switch CAM table based on ARP communications from the connected hosts. ARP spoofing manipulates the CAM table in a manner that modifies the entries to force traffic to the attacking host.

References:

- "ARP spoofing," Wikipedia, https://en.wikipedia.org/wiki/ARP_spoofing
- Image source: https://documentation.meraki.com/MS/Other_Topics/Dynamic_ARP_Inspection

DNS Spoofing

DNS spoofing and **DNS cache poisoning** is a form of computer security hacking in which corrupt Domain Name System data is introduced into the DNS resolver's cache causing the name server to return an incorrect result record, e.g., an IP address. This results in traffic being diverted to the attacker's computer (or any other computer)



The Domain Name System (DNS) is responsible for resolving a request for a network resource by name, typically a computer name, to an IP address. DNS spoofing involves intercepting the request and replying with an alternative IP address of a resource controlled by the attacker. The intent is to direct all communication intended for the original host to the attacker's host for monitoring, manipulation, and spoofing.

References:

- "DNS spoofing," Wikipedia, https://en.wikipedia.org/wiki/DNS_spoofing
- Image source: https://medium.com/iocscan/dns-cache-poisoning-bea939b5afaf

Rogue Access Point / Evil Twin

- A **rogue access point** is a wireless access point that has been installed on a secure network without explicit authorization from a local network administrator, whether added by a well-meaning employee or by a malicious attacker
- An **evil twin** is a fraudulent Wi-Fi access point that appears to be legitimate but is set up to eavesdrop on wireless communications

Wireless is becoming prevalent within the ICS environment. In most cases IT manages all approved wireless within the plant as an extension of their corporate wireless infrastructure. When considering the management of the wireless infrastructure, one must be diligent to actively scan for rogue or unapproved access points. These scans should also identify default-enabled wireless interfaces from newer vendor-supplied equipment.

References:

- "Rogue access point," Wikipedia, https://en.wikipedia.org/wiki/Rogue_access_point
- "Evil twin (wireless networks)," Wikipedia, https://en.wikipedia.org/wiki/Evil_twin_(wireless_networks)
- SEC617 Wireless Penetration Testing and Ethical Hacking https://www.sans.org/cyber-security-courses/wireless-penetration-testing-ethical-hacking/

Transparent Proxy

Also known as an **intercepting proxy**, **inline proxy**, or **forced proxy**, a **transparent proxy** intercepts normal communication at the network layer without requiring any special client configuration.

- Clients do not need to be aware of the existence of the proxy.

A transparent proxy is not known to the client and therefore provides a traffic interception point that can be used, with appropriate certificates, to decrypt sensitive data from the client-encrypted communications. Use of this type of interception would be possible on third-party networks used within external ICS communications.

Reference:

- "Proxy server," Wikipedia, https://en.wikipedia.org/wiki/Proxy_server

Common Platform and Tools to Perform Attack

- Kali Linux
  - Kali Linux is an open-source penetrating testing platform/distro that is maintained and funded by Offensive Security
- Ettercap
  - Ettercap is a useful tool for performing arp poisoning attacks on network switches and gaining visibility to network traffic

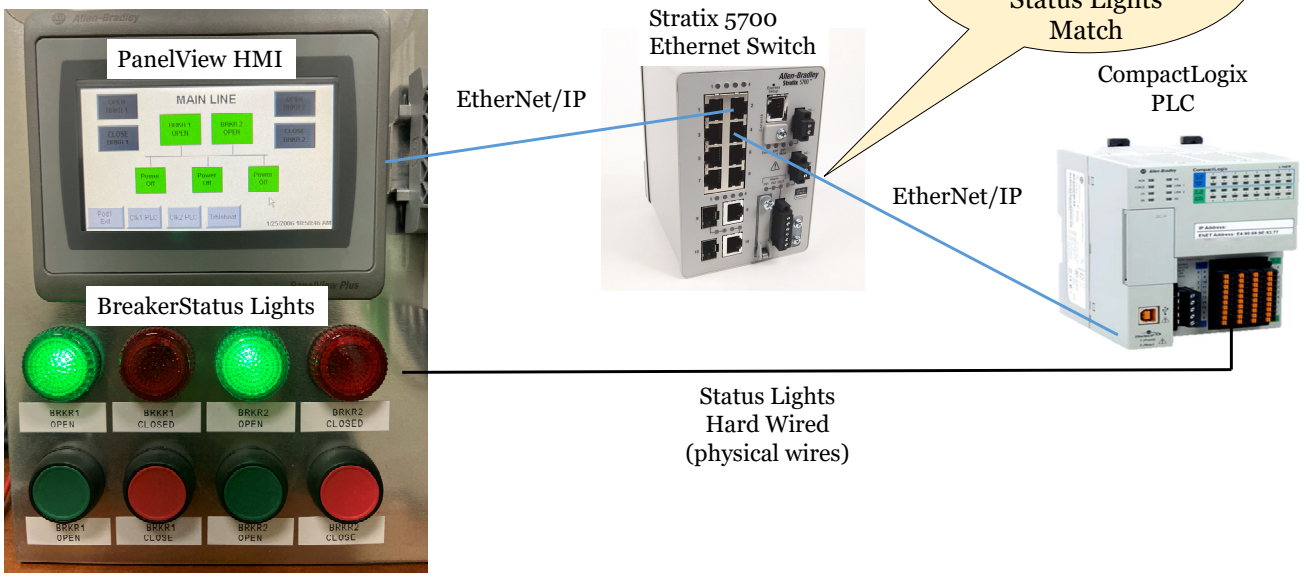
References:
- https://www.kali.org/about-us/

MITM Defenses

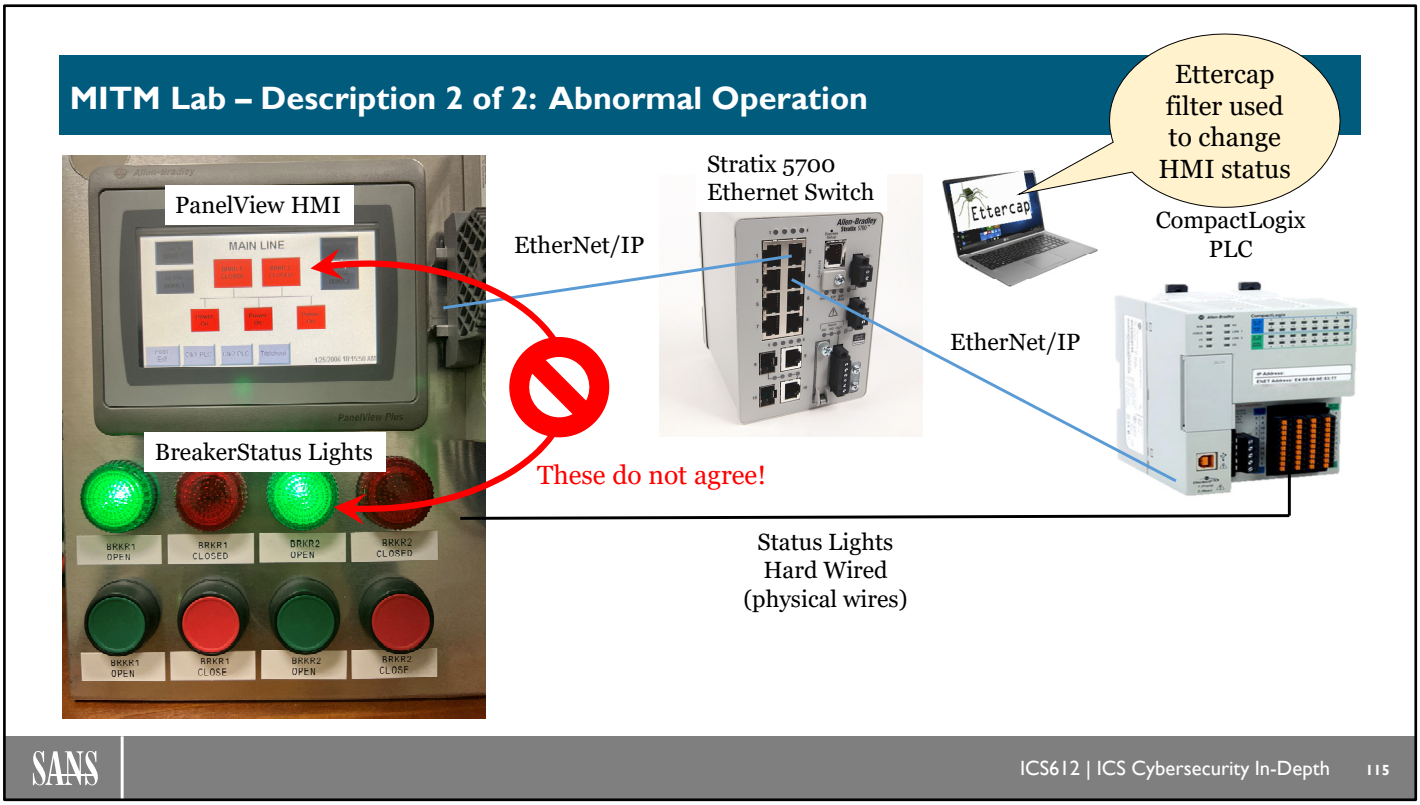
- VPNs
- HTTPs
- Public Key Based Authentication
- Strong Wireless Encryption
- Switchport Security
- Static / Dynamic MAC Assignments to Ports

Understanding the business and adversarial value of the data will help prioritize what communication needs to be protected. Prioritize your defenses of this communication and implement them at the layers of the OSI model where the greatest risks are present based on your communication flow. Also, understanding the business and adversarial value of the data will help prioritize what data needs to be protected. Multiple options exist in how to defend against MITM attacks. The selection and type of defenses requires an understanding of the communication capabilities of the endpoints. A change in the communication architecture might be the best approach when dealing with plaintext protocols.

MitM Lab – Description 1 of 2: Normal Operation

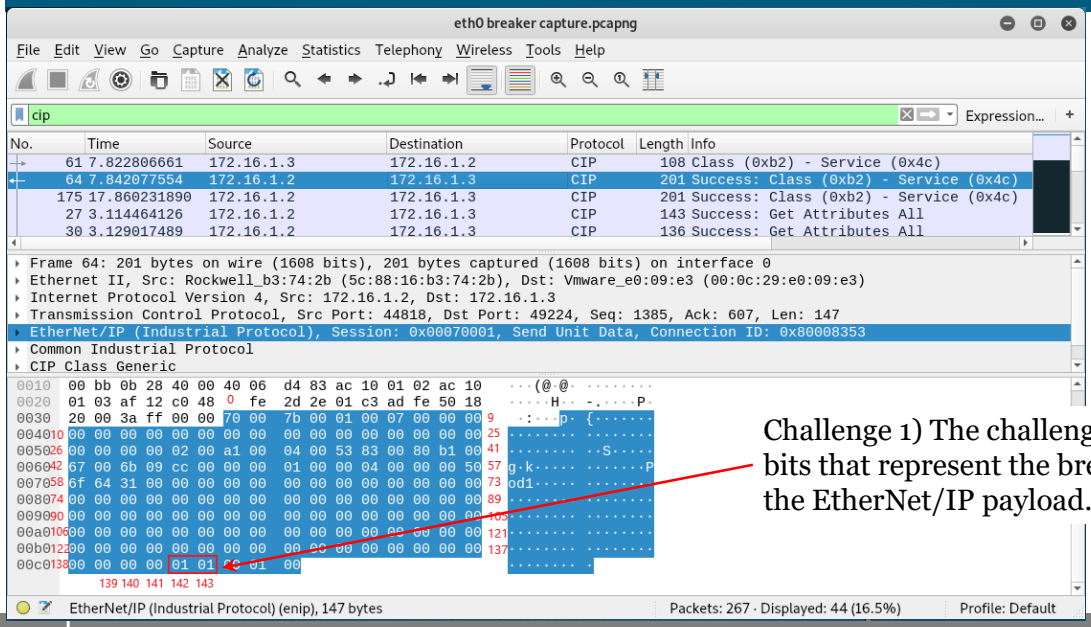


In this lab, we will conduct a Machine in the Middle (MitM) attack to cause the status of the physical breaker lights to not match the status of the PanelView HMI. The PLC sends the status of the breakers to the HMI over the network which provides an opportunity to "spoof" the breaker status to the HMI.



We will use Ettercap to capture the traffic, cerate a custom filter to intercept and change the bit patter that is sent from the PLC to the HMI. We can note that this attack causes the status of the physical breaker lights to not match the status of the PanelView HMI.

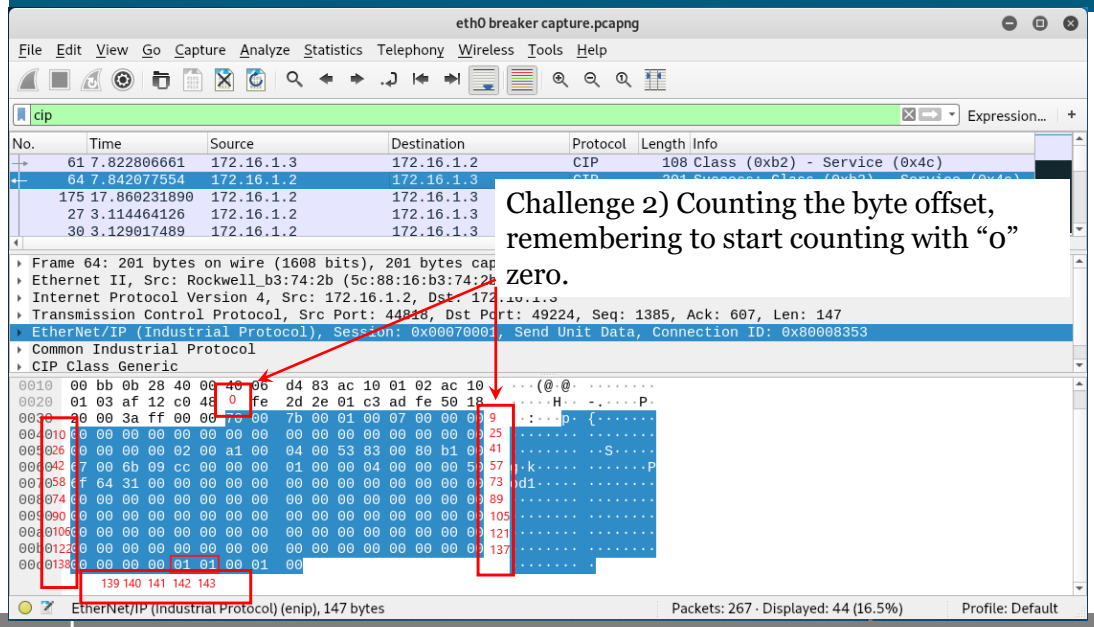
MitM Lab – Challenge 1 of 2



Challenge 1) The challenge is to find the bits that represent the breaker status in the EtherNet/IP payload.

In this lab, we will conduct a Machine in the Middle (MitM) attack to cause the status of the physical breaker lights to not match the status of the PanelView HMI.

MitM Lab – Challenge 2 of 2



In this lab, we will conduct a Machine in the Middle (MitM) attack to cause the status of the physical breaker lights to not match the status of the PanelView HMI.

Lab 2.9: Local Network MiTM Attack

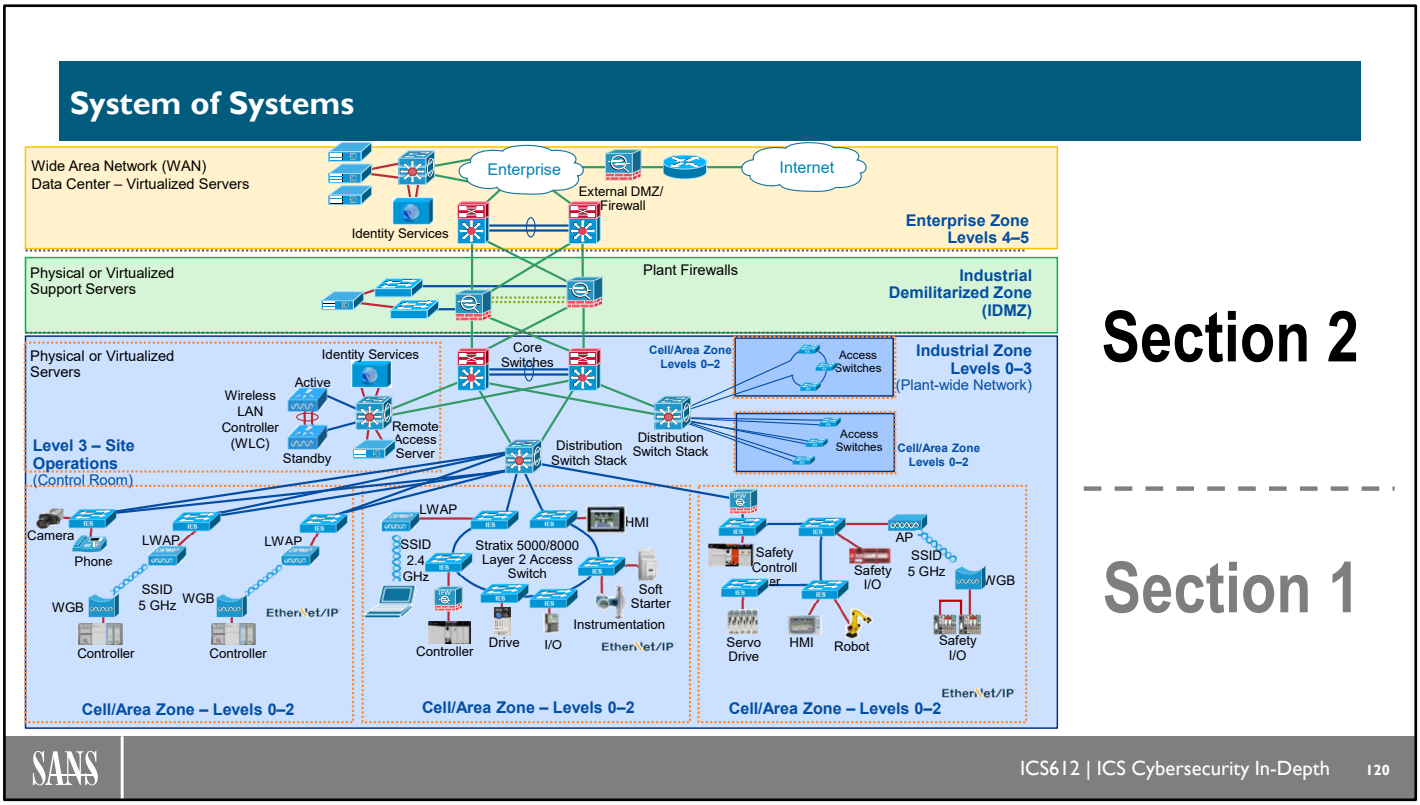
Go to the Lab Workbook: Lab 2.9

This page intentionally left blank.

Local Attacks and Process Manipulation Checkpoint 2.9

- Initially, we performed a local device attack, we have now covered an approach to attack the process through the network components near the local process
- Achieving a network attack through a variety of different approaches
  - Transparent proxy
  - ARP spoofing
  - Modified payload in transit

This page intentionally left blank.

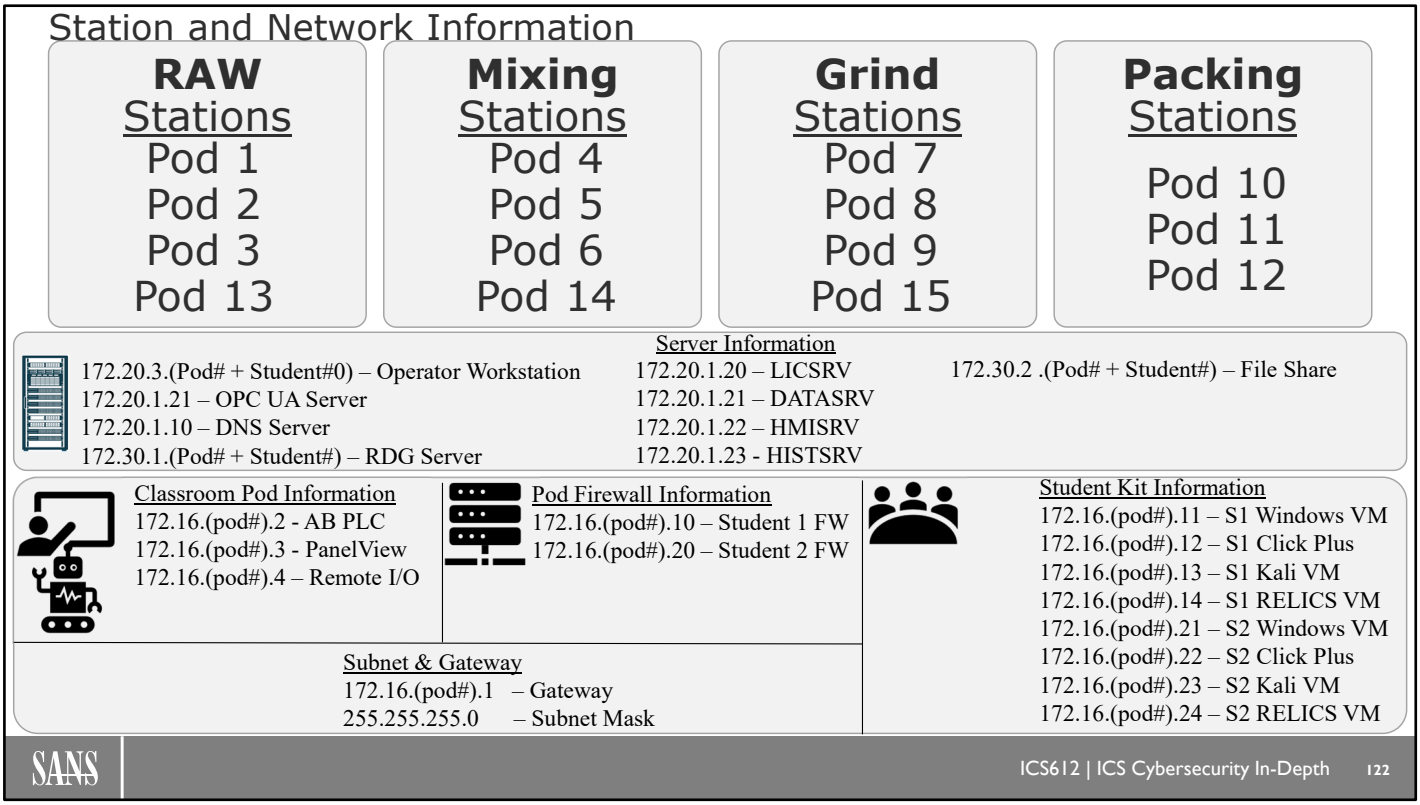


Here is a quick reminder of where you have been working so far in the course.

Section 2 Summary

- We have interconnected and built a system in the classroom that has a series of trusted communications paths
- Mapping and understanding the communications, flows, and interdependencies that exist from Level 0–3 and how the head end process can be impacted is critical as we begin to design and implement enforcement zones and further secure the environment
- It is also important to look at the systems from the assumed breach perspective of an adversary with access and what they can achieve with that access

This page intentionally left blank.



This page intentionally left blank.