NIST SPECIAL PUBLICATION 1800-10

Protecting Information and System Integrity in Industrial Control System Environments:

Cybersecurity for the Manufacturing Sector

Includes Executive Summary (A); Approach, Architecture, and Security Characteristics (B); and How-To Guides (C)

Michael Powell
Joseph Brule
Michael Pease
Keith Stouffer
CheeYee Tang
Timothy Zimmerman
Chelsea Deane
John Hoyt
Mary Raguso
Aslam Sherule
Kangmin Zheng
Matthew Zopf

FINAL

March 2022

This publication is available free of charge from
https://doi.org/10.6028/NIST.SP.1800-10

The first draft of this publication is available free of charge from
https://www.nccoe.nist.gov/publications/practice-guide/protecting-information-and-system-integrity-industrial-control-system-draft

NIST SPECIAL PUBLICATION 1800-10

Protecting Information and System Integrity in Industrial Control System Environments: Cybersecurity for the Manufacturing Sector

Includes Executive Summary (A); Approach, Architecture, and Security Characteristics (B); and How-To Guides (C)

Michael Powell
National Cybersecurity Center of Excellence
National Institute of Standards and Technology

Michael Pease
Keith Stouffer
CheeYee Tang
Timothy Zimmerman
Engineering Laboratory
National Institute of Standards and Technology

Joe Brule
Chelsea Deane
John Hoyt
Mary Raguso
Aslam Sherule
Kangmin Zheng
The MITRE Corporation
McLean, Virginia

Matthew Zopf
Strativia
Largo, Maryland

FINAL

March 2022

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

NIST SPECIAL PUBLICATION 1800-10A

Protecting Information and System Integrity in Industrial Control System Environments:

Cybersecurity for the Manufacturing Sector

Volume A:
Executive Summary

Michael Powell
National Cybersecurity Center of Excellence
National Institute of Standards and Technology

Michael Pease
Keith Stouffer
CheeYee Tang
Timothy Zimmerman
Engineering Laboratory
National Institute of Standards and Technology

Matthew Zopf
Strativia
Largo, Maryland

Joseph Brule
Chelsea Deane
John Hoyt
Mary Raguso
Aslam Sherule
Kangmin Zheng
The MITRE Corporation
McLean, Virginia

FINAL

March 2022

This publication is available free of charge from
https://doi.org/10.6028/NIST.SP.1800-10

The first draft of this publication is available free of charge from
https://www.nccoe.nist.gov/publications/practice-guide/protecting-information-and-system-integrity-industrial-control-system-draft

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

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Executive Summary

Many manufacturing organizations rely on industrial control systems (ICS) to monitor and control their machinery, production lines, and other physical processes that produce goods. To stay competitive, manufacturing organizations are increasingly connecting their operational technology (OT) systems to their information technology (IT) systems to enable and expand enterprise-wide connectivity and remote access for enhanced business processes and capabilities.

Although the integration of IT and OT networks is helping manufacturers boost productivity and gain efficiencies, it has also provided malicious actors, including nation states, common criminals, and insider threats a fertile landscape where they can exploit cybersecurity vulnerabilities to compromise the integrity of ICS and ICS data to reach their end goal. The motivations behind these attacks can range from degrading manufacturing capabilities to financial gain, and causing reputational harm.

Once malicious actors gain access, they can harm an organization by compromising data or system integrity, hold ICS and/or OT systems ransom, damage ICS machinery, or cause physical injury to workers. The statistics bear this out. The X-Force Threat Intelligence Index 2021 (ibm.com) stated that manufacturing was the second-most-attacked industry in 2020, up from eighth place in 2019.

One particular case study illustrates the long-lasting effects and damage a single cyber attack can inflict on an organization. It was reported that a global pharmaceutical manufacturer suffered a cyber attack that caused temporary production delays at a facility making a key vaccination. More than 30,000 laptop and desktop computers, along with 7,500 servers, sat idle. Although the company claimed that its operations were back to normal within six months of the incident; at this writing, news reports stated that the organization is locked in a legal battle with its insurers and is looking to reclaim expenses that include repairing its computer networks and the costs associated with interruptions to its operations. They are seeking more than $1.3 billion in damages.

To address the cybersecurity challenges facing the manufacturing sector, the National Institute of Standards and Technology's (NIST's) National Cybersecurity Center of Excellence (NCCoE) launched this project in collaboration with NIST's Engineering Laboratory (EL) and cybersecurity technology providers. Together, we have built example solutions that manufacturing organizations can use to mitigate ICS integrity risks, strengthen the cybersecurity of OT systems, and protect the data that these systems process.

CHALLENGE

The manufacturing industry is critical to the economic well-being of the nation, and is constantly seeking ways to modernize its systems, boost productivity, and raise efficiency. To meet these goals, manufacturers are modernizing their OT systems by making them more interconnected and integrated with other IT systems and introducing automated methods to strengthen their overall OT asset management capabilities.

As OT and IT systems become increasingly interconnected, manufacturers have become a major target of more widespread and sophisticated cybersecurity attacks, which can disrupt these processes and cause damage to equipment and/or injuries to workers. Furthermore, these incidents could significantly impact productivity and raise operating costs, depending on the extent of a cyber attack.

This practice guide can help your organization:

- detect and prevent unauthorized software installation
- protect ICS networks from potentially harmful applications
- determine changes made to a network using change management tools
- detect unauthorized use of systems
- continuously monitor network traffic
- leverage anti-malware tools

SOLUTION

The NCCoE, in conjunction with the NIST EL, collaborated with cybersecurity technology providers to develop and implement example solutions that demonstrate how manufacturing organizations can protect the integrity of their data from destructive malware, insider threats, and unauthorized software within manufacturing environments that rely on ICS.

The example solutions use technologies and security capabilities from the project collaborators listed in the table below. These technologies were implemented in two distinct manufacturing lab environments that emulate discrete and continuous manufacturing systems. This project takes a modular approach in demonstrating two unique builds in each of the lab environments.

The following is a list of the project's collaborators.

Collaborator	Component
DISPEL	Provides secure remote access with authentication and authorization support.
DRAGOS	Provides network and asset monitoring to detect behavior anomalies and modifications to hardware, firmware, and software capabilities.
FORESCOUT	Provides network and asset monitoring to detect behavior anomalies and modifications to hardware, firmware, and software capabilities.
GreenTec™ www.GreenTec-USA.com	Offers secure data storage on-prem.
Microsoft	Provides network and asset monitoring to detect behavior anomalies and modifications to hardware, firmware, and software capabilities.

Collaborator	Component
OSIsoft is now part of AVEVA	Real-time data management software that enables detection of behavior anomalies and modifications to hardware, firmware, and software capabilities.
tdi technologies	Access control platform that secures connections and provides control mechanisms to enterprise systems for authorized users and devices; monitors activity down to the keystroke
tenable	Provides network and asset monitoring to detect behavior anomalies and modifications to hardware, firmware, and software capabilities.
vmware	Provides host-based application allowlisting (the blocking of unauthorized activities that have the potential to pose a harmful attack) and file integrity monitoring.

While the NCCoE used a suite of commercial products to address this challenge, this guide does not endorse these particular products, nor does it guarantee compliance with any regulatory initiatives. Your organization's information security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of a solution.

HOW TO USE THIS GUIDE

Depending on your role in your organization, you might use this guide in different ways:

Business decision makers, including chief information security and technology officers, can use this part of the guide, *NIST SP 1800-10A: Executive Summary*, to understand the drivers for the guide, the cybersecurity challenge we address, our approach to solving this challenge, and how the solution could benefit your organization.

Technology, security, and privacy program managers who are concerned with how to identify, understand, assess, and mitigate risk can use *NIST SP 1800-10B: Approach, Architecture, and Security Characteristics*. It describes what we built and why, including the risk analysis performed and the security/privacy control mappings.

Technology professionals who want to implement an approach like this can make use of *NIST SP 1800-10C: How-To Guides*. It provides specific product installation, configuration, and integration instructions for building the example implementation, allowing you to replicate all or parts of this project.

SHARE YOUR FEEDBACK

You can view or download the guide at https://www.nccoe.nist.gov/projects/use-cases/manufacturing/integrity-ics.

Once the example implementation is developed, you can adopt this solution for your own organization. If you do, please share your experience and advice with us. We recognize that technical solutions alone will not fully enable the benefits of our solution, so we encourage organizations to share lessons learned and best practices for transforming the processes associated with implementing this guide.

To provide comments, join the community of interest, or to learn more about the project and example implementation, contact the NCCoE at manufacturing_nccoe@nist.gov.

COLLABORATORS

Collaborators participating in this project submitted their capabilities in response to an open call in the Federal Register for all sources of relevant security capabilities from academia and industry (vendors and integrators). Those respondents with relevant capabilities or product components signed a Cooperative Research and Development Agreement (CRADA) to collaborate with NIST in a consortium to build this example solution.

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

NIST SPECIAL PUBLICATION 1800-10B

Protecting Information and System Integrity in Industrial Control System Environments:

Cybersecurity for the Manufacturing Sector

Volume B:
Approach, Architecture, and Security Characteristics

Michael Powell
National Cybersecurity Center of Excellence
National Institute of Standards and Technology

Michael Pease
Keith Stouffer
CheeYee Tang
Timothy Zimmerman
Engineering Laboratory
National Institute of Standards and Technology

Matthew Zopf
Strativia
Largo, Maryland

Joseph Brule
Chelsea Deane
John Hoyt
Mary Raguso
Aslam Sherule
Kangmin Zheng
The MITRE Corporation
McLean, Virginia

FINAL

March 2022

This publication is available free of charge from
https://doi.org/10.6028/NIST.SP.1800-10

The first draft of this publication is available free of charge from
https://www.nccoe.nist.gov/publications/practice-guide/protecting-information-and-system-integrity-industrial-control-system-draft

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

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

DISCLAIMER

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

Domain name and IP addresses shown in this guide represent an example domain and network environment to demonstrate the NCCoE project use case scenarios and the security capabilities.

National Institute of Standards and Technology Special Publication 1800-10B, Natl. Inst. Stand. Technol. Spec. Publ. 1800-10B, 149 pages, March 2022, CODEN: NSPUE2

FEEDBACK

As a private-public partnership, we are always seeking feedback on our practice guides. We are particularly interested in seeing how businesses apply NCCoE reference designs in the real world. If you have implemented the reference design, or have questions about applying it in your environment, please email us at manufacturing_nccoe@nist.gov.

All comments are subject to release under the Freedom of Information Act (FOIA).

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, easily adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST *Cybersecurity Framework* (CSF) and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align more easily with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Today's manufacturing organizations rely on industrial control systems (ICS) to conduct their operations. Increasingly, ICS are facing more frequent, sophisticated cyber attacks—making manufacturing the second-most-targeted industry [1]. Cyber attacks against ICS threaten operations and worker safety, resulting in financial loss and harm to the organization's reputation.

The architecture and solutions presented in this guide are built upon standards-based, commercially available products, and represent some of the possible solutions. The solutions implement standard cybersecurity capabilities such as behavioral anomaly detection (BAD), application allowlisting (AAL), file

integrity-checking, change control management, and user authentication and authorization. The solution was tested in two distinct lab settings: a discrete manufacturing workcell, which represents an assembly line production, and a continuous process control system (PCS), which represents chemical manufacturing industries.

An organization that is interested in protecting the integrity of a manufacturing system and information from destructive malware, insider threats, and unauthorized software should first conduct a risk assessment and determine the appropriate security capabilities required to mitigate those risks. Once the security capabilities are identified, the sample architecture and solution presented in this document may be used.

The security capabilities of the example solution are mapped to the *NIST Cybersecurity Framework*, the *National Initiative for Cybersecurity Education Framework*, and *NIST Special Publication 800-53*.

KEYWORDS

Application allowlisting; behavioral anomaly detection; file integrity checking; firmware modification; industrial control systems; manufacturing; remote access; software modification; user authentication; user authorization.

ACKNOWLEDGEMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Dan Frechette	Microsoft
Ian Schmertzler	Dispel
Ben Burke	Dispel
Chris Jensen	Tenable
Bethany Brower	VMWare
Dennis Hui	OSIsoft (now part of AVEVA)
John Matranga	OSIsoft (now part of AVEVA)
Michael A. Piccalo	Forescout
Tim Jones	Forescout
Yejin Jang	Forescout
Samantha Pelletier	TDI Technologies
Rusty Hale	TDI Technologies
Steve Petruzzo	GreenTec

Name	Organization
Josh Carlson	Dragos
Alex Baretta	Dragos

The Technology Partners/Collaborators who participated in this build submitted their capabilities in response to a notice in the Federal Register. Respondents with relevant capabilities or product components were invited to sign a Cooperative Research and Development Agreement (CRADA) with NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Product
Carbon Black (VMware)	Carbon Black App Control
Microsoft	Azure Defender for the internet of things (IoT) (incorporating technology from the acquisition of CyberX)
Dispel	Dispel Wicket ESI Dispel Enclave Dispel VDI (Virtual Desktop Interface)
Dragos	Dragos Platform
Forescout	eyeInspect (Formerly SilentDefense) ICS Patrol EyeSight
GreenTec	WORMdisk and ForceField
OSIsoft (now part of AVEVA)	PI System (which comprises products such as PI Server, PI Vision and others)
TDi Technologies	ConsoleWorks
Tenable	Tenable.ot

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly to conform to the publication and from which no deviation is permitted. The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited. The terms “may” and “need not” indicate a course of action permissible within the limits of the publication. The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

PATENT DISCLOSURE NOTICE

NOTICE: The Information Technology Laboratory (ITL) has requested that holders of patent claims whose use may be required for compliance with the guidance or requirements of this publication disclose such patent claims to ITL. However, holders of patents are not obligated to respond to ITL calls for patents and ITL has not undertaken a patent search in order to identify which, if any, patents may apply to this publication.

As of the date of publication and following call(s) for the identification of patent claims whose use may be required for compliance with the guidance or requirements of this publication, no such patent claims have been identified to ITL.

No representation is made or implied by ITL that licenses are not required to avoid patent infringement in the use of this publication.

Contents

List of Figures

List of Tables

1 Summary

While availability is always a critical aspect of manufacturing system environments, manufacturers also need to consider maintaining the integrity of their systems and information to ensure continued operations. The integrity of information can be degraded or lost as a result of behaviors by authorized users (e.g., failure to perform backups or record their actions) or malicious actors seeking to disrupt manufacturing operations for illicit profits, political statements, or other reasons.

Manufacturers are unique because of their reliance on industrial control systems (ICS) to monitor and control their manufacturing operations. ICS typically prioritize information availability and integrity over confidentiality. As a result, cybersecurity solutions used in traditional information technology (IT) settings are not optimized to protect ICS from cyber threats.

This guide, prepared by the National Cybersecurity Center of Excellence (NCCoE) and the NIST Engineering Laboratory (EL), contains four examples of practical solutions that organizations can implement in their environments to protect ICS from information and system integrity attacks.

The goal of this NIST Cybersecurity Practice Guide is to help organizations protect the integrity of systems and information by:

- securing historical system data
- preventing execution or installation of unapproved software
- detecting anomalous behavior on the network
- identifying hardware, software, or firmware modifications
- enabling secure remote access
- authenticating and authorizing users

This document provides a detailed description of how each solution was implemented and what technologies were used to achieve each of the above listed goals across four example builds. Scenarios are used to demonstrate the efficacy of the solutions. The results and challenges of each scenario in the four example builds are also presented and discussed.

Ultimately, manufacturing organizations that rely on ICS can use the example solutions described in this guide to safeguard their information and system integrity from:

- destructive malware
- insider threats
- unauthorized software
- unauthorized remote access

- loss of historical data
- anomalous network traffic
- unauthorized modification of systems

This document contains the following sections:

Section 1, Summary, presents the challenges addressed by the NCCoE project, with a look at the solutions demonstrated to address the challenge, as well as benefits of the solutions.

Section 2, How to Use This Guide, explains how readers—business decision makers, program managers, control system engineers, cybersecurity practitioners, and IT professionals (e.g., systems administrators)— might use each volume of this guide.

Section 3, Approach, offers a description of the intended audience and the scope of the project. This section also describes the assumptions on which the security architecture and solution development was based, the risk assessment that informed architecture development, the NIST *Cybersecurity Framework* functions supported by each component of the architecture and reference design, and which industry collaborators contributed support in building, demonstrating, and documenting the solutions. This section also includes a mapping of the NIST *Cybersecurity Framework* Subcategories to other industry guidance, and identifies the products used to address each subcategory.

Section 4, Architecture, summarizes the Cybersecurity for Smart Manufacturing Systems (CSMS) demonstration environment, which emulates real-world manufacturing processes and their ICS by using software simulators and commercial off-the-shelf hardware in a laboratory environment. The implementation of the information and system integrity solutions is also described.

Section 5, Security Characteristic Analysis, summarizes the scenarios and findings that were employed to demonstrate the example implementations' functionality. Each of the scenarios is mapped to the relevant NIST *Cybersecurity Framework* functions and Subcategories and the security capabilities of the products that were implemented. Additionally, it briefly describes how the security capabilities that were used in the solution implementation help detect cyber attacks and protect the integrity of the manufacturing systems and information.

Section 6, Future Build Considerations, identifies additional areas that should be reviewed in future practice guides.

Section Appendix D, Scenario Execution Results, describes, in detail, the test results of the scenarios, including screenshots from the security products captured during the tests.

1.1 Challenge

Manufacturing organizations that rely on ICS to monitor and control physical processes face risks from malicious and non-malicious insiders along with external threats in the form of increasingly sophisticated cyber attacks. A compromise to system or information integrity may very well pose a significant threat to human safety and can adversely impact an organization's operations, resulting in financial loss and harm production for years to come.

Manufacturing organizations may be the targets of malicious cyber actors or may be incidentally impacted by a broader malware event such as ransomware attacks. ICS components remain vulnerable to cyber attacks for numerous reasons, including adoption and integration of enhanced connectivity, remote access, the use of legacy technologies, flat network topologies, lack of network segmentation, and the lack of cybersecurity technologies (e.g., anti-virus, host-based firewalls, encryption) typically found on IT systems.

Organizations are increasingly adopting and integrating IT into the ICS environment to enhance connectivity to business systems and to enable remote access. As a result, ICS are no longer isolated from the outside world, making them more vulnerable to cyber attacks. Security controls designed for the IT environment may impact the performance of ICS when implemented within the operational technology (OT) environment, so special precautions are required when introducing these controls. In some cases, new security techniques tailored to the specific ICS environment are needed.

Another challenge facing manufacturing organizations comes from authorized users who accidentally or intentionally compromise information and system integrity. For example, a user may install an unapproved software utility to perform maintenance activities or update the logic of a programmable logic controller (PLC) to fix a bug. Even if the software or logic changes are not malicious, they may inadvertently disrupt information flows, starve critical software of processing resources, or degrade the operation of the system. In a worst-case scenario, malware may be inadvertently installed on the manufacturing system, causing disruptions to system operations, or opening a backdoor to remote attackers.

1.2 Solution

This NCCoE Cybersecurity Practice Guide demonstrates how manufacturing organizations can use commercially available technologies that are consistent with cybersecurity standards to detect and prevent cyber incidents on their ICS.

Manufacturers use a wide range of ICS equipment and manufacturing processes. This guide contains four different example solutions that are applicable to a range of manufacturing environments, focusing on discrete and continuous manufacturing processes.

This project provides example solutions, composed of the following capabilities, for manufacturing environments:

- application allowlisting (AAL)
- behavior anomaly detection (BAD)

- file integrity
- user authentication and authorization
- remote access

1.2.1 Relevant Standards and Guidance

The solutions presented in this guide are consistent with the practices and guidance provided by the following references:

- NIST Special Publication (SP) 800-167: *Guide to Application Whitelisting* [2]
- Department of Homeland Security, *Critical Manufacturing Sector Cybersecurity Framework Implementation Guidance* [3]
- Executive Order no. 13636: *Improving Critical Infrastructure Cybersecurity* [4]
- NIST, *Framework for Improving Critical Infrastructure Cybersecurity* [5]
- NIST Interagency Report (NISTIR) 8219: *Securing Manufacturing Industrial Control Systems: Behavioral Anomaly Detection* [6]
- NIST Internal Report (NISTIR) 8183: *Cybersecurity Framework Manufacturing Profile* [7]
- NISTIR 8089: *An Industrial Control System Cybersecurity Performance Testbed* [8]
- NIST SP 800-53 Rev. 5: *Security and Privacy Controls for Federal Information Systems and Organizations* [9]
- NIST SP 800-181: *National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework* [10]
- NIST Special Publication 1800-25: *Data Integrity: Identifying and Protecting Assets Against Ransomware and Other Destructive Events* [11]
- NIST Interagency or Internal Report 7298 Rev 3: *Glossary of Key Information Security Terms* [12]
- U.S.-Canada Power System Outage Task Force [13]
- NIST SP 800-82 Rev. 2: *Guide to Industrial Control Systems (ICS) Security* [14]

1.3 Benefits

This NCCoE practice guide can help organizations:

- mitigate cybersecurity risk
- reduce downtime for operations
- provide a reliable environment that can detect cyber anomalies
- respond to security alerts through automated cybersecurity-event products

- develop and execute an OT cybersecurity strategy for which continuous OT cybersecurity monitoring is a foundational building block
- implement current cybersecurity standards and best practices

2 How to Use This Guide

This NIST Cybersecurity Practice Guide demonstrates a modular design and provides users with the information they need to replicate the described manufacturing ICS security solutions, specifically focusing on information and system integrity. This reference design is modular and can be deployed in whole or in part.

This guide contains three volumes:

- NIST SP 1800-10A: *Executive Summary*
- NIST SP 1800-10B: *Approach, Architecture, and Security Characteristics* – what we built and why (**this document**)
- NIST SP 1800-10C: *How-To Guide* – instructions for building the example solution

Depending on your role in your organization, you might use this guide in different ways:

Senior information technology (IT) executives, including chief information security and technology officers, will be interested in the *Executive Summary*, NIST SP 1800-10A, which describes the following topics:

- challenges that enterprises face in ICS environments in the manufacturing sector
- example solution built at the NCCoE
- benefits of adopting the example solution

Technology or security program managers might share the *Executive Summary*, NIST SP 1800-10A, with your leadership to help them understand the importance of adopting a standards-based solution. Doing so can strengthen their information and system integrity practices by leveraging capabilities that may already exist within their operating environment or by implementing new capabilities.

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in NIST SP 1800-10B (this document), which describes what we did and why. The following section will be of particular interest:

- Section 3.4.4, which maps the security characteristics of the example solutions to cybersecurity standards and best practices

IT and OT professionals who want to implement an approach like this will find the whole practice guide useful, particularly the how-to portion, NIST SP 1800-10C, which provides step-by-step details to replicate all, or parts of the example solutions created in our lab. Volume C does not re-create the product manufacturers' documentation, which is generally widely available. Rather, Volume C shows how we integrated the products together to create an example solution.

This guide assumes that IT and OT professionals have experience implementing security products within the enterprise. While we have used a suite of commercial products to address this challenge, this guide does not endorse these particular products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of the manufacturing ICS solution. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and best practices. Section 3.5, Technologies, lists the products we used and maps them to the cybersecurity controls provided by this reference solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution. Every organization is unique in its priorities, risk tolerance, and the cyber ecosystem they operate in. This document presents a possible solution that may be tailored or augmented to meet an organization's own needs.

This document provides initial guidance. We seek feedback on its contents and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to manufacturing_nccoe@nist.gov.

2.1 Typographic Conventions

The following table presents typographic conventions used in this volume.

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For language use and style guidance, see the *NCCoE Style Guide*.
Bold	names of menus, options, command buttons, and fields	Choose **File > Edit**.
`Monospace`	command-line input, onscreen computer output, sample code examples, and status codes	`mkdir`
`Monospace Bold`	command-line user input contrasted with computer output	**`service sshd start`**
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

3 Approach

This practice guide documents the approach the NCCoE used to develop example solutions, called builds, to support information and system integrity objectives. The approach includes a logical design, example build development, testing, security control mapping, and analysis.

Based on our discussions with cybersecurity practitioners in the manufacturing sector, the NCCoE pursued the Information and System Integrity in ICS Environments project to illustrate the broad set of capabilities available to manage and protect OT assets.

The NCCoE collaborated with the NIST Engineering Lab (EL), Community of Interest (COI) members, and the participating vendors to produce an example architecture and its corresponding implementations. Vendors provided technologies that met project requirements and assisted in installation and configuration of those technologies. This practice guide highlights the implementation of example architectures, including supporting elements such as functional tests, security characteristic analysis, and future build considerations.

3.1 Audience

This guide is intended for individuals or entities responsible for cybersecurity of ICS and for those interested in understanding information and system integrity capabilities for OT and how one approaches the implementation of an architecture. It may also be of interest to anyone in industry, academia, or government who seeks general knowledge of an OT information and system integrity solution for manufacturing-sector organizations.

3.2 Scope

This document focuses on information and system integrity in ICS environments typical of manufacturing organizations. It provides real-world guidance on implementing a solution for manufacturing ICS environments.

The scope of this project is to assist organizations in maintaining the integrity of information and systems by:

- Preventing execution or installation of unapproved software
- preventing unauthorized access to systems and information
- detecting anomalous behavior on the network that affects system or information integrity
- detecting hardware, software, or firmware modification

Organizational cybersecurity policies and procedures, as well as response and recovery functions, are out of scope for this document. The scenarios and security capabilities covered in this practice guide should be part of a comprehensive OT/ICS security plan that addresses the NIST *Cybersecurity Framework* Protect and Detect functions.

The security capabilities used in this demonstration for protecting information and system integrity in ICS environments are briefly described below. These capabilities are implemented using commercially available third-party and open source solutions that provide the following capabilities:

- **Application Allowlisting (AAL):** A list of applications and application components (libraries, configuration files, etc.) that are authorized to be present or active on a host according to a well-defined baseline. [2]
- **Behavioral Anomaly Detection (BAD)**: A mechanism providing a multifaceted approach to detecting cybersecurity attacks. [6]
- **Hardware/Software/Firmware Modification Detection**: A mechanism providing the ability to detect changes to hardware, software, and firmware on systems or network connected devices.
- **File Integrity Checking**: A mechanism providing the ability to detect changes to files on systems or network-connected devices.
- **User Authentication and Authorization**: A mechanism for verifying the identity and the access privileges granted to a user, process, or device. [12]
- **Remote Access**: A mechanism supporting access to an organizational information system by a user (or an information system acting on behalf of a user) communicating through an external network (e.g., the Internet). [12]

3.3 Assumptions

This project makes the following assumptions:

- Each solution is comprised of several readily available products. The modularity of the solutions might allow organizations to consider swapping one or more products, depending on their specific requirements.
- A cybersecurity stakeholder might implement all or part of a solution in a manner that is compatible with their existing environment.
- Organizations will test and evaluate the compatibility of the solutions with their ICS devices prior to production implementation and deployment. Response and recovery functions are beyond the scope of this guide.
- Events detected by the security tools are passed on to the security operation team for further action.

3.4 Risk Assessment

NIST SP 800-30 Revision 1, *Guide for Conducting Risk Assessments*, states that risk is "a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence." The guide further defines risk assessment as "the process of identifying, estimating, and prioritizing risks to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, other organizations, and the Nation, resulting from the operation of an information system. Part of risk management incorporates threat and vulnerability analyses, and considers mitigations provided by security controls planned or in place."

The NCCoE recommends that any discussion of risk management, particularly at the enterprise level, begins with a comprehensive review of NIST SP 800-37 Revision 2, *Risk Management Framework for Information Systems and Organizations*, material that is available to the public. The Risk Management Framework (RMF) guidance, as a whole, proved to be invaluable in giving us a baseline to assess risks, from which we developed the project, the security characteristics of the build, and this guide.

3.4.1 Threats

A threat is "any circumstance or event with the potential to adversely impact organizational operations" [11]. Within an IT environment, threats are typically thought of in terms of threats to confidentiality, integrity, or availability.

The realization of a threat to confidentiality, integrity, and availability may have different impacts to the OT versus the IT environments. OT environments are sensitive to loss of safety, availability, and integrity, while traditional IT environments tend to direct more resources toward confidentiality. Organizations that combine IT and OT operations are advised to evaluate the threats from both perspectives.

In a cyber-physical system, cybersecurity stakeholders are advised to consider events that occur in the OT environment may have impact to physical assets and events that occur in the physical world may impact the OT environment. For example, in 2021 a ransomware attack against an American oil pipeline system led to a disruption of operations and ultimately resulted in fuel shortages at airports and filling stations on the United States east coast. At the time of this writing, a full assessment has not been completed, but the economic impact to the pipeline was substantial.

An integrity loss need not be malicious to cause a significant impact. For example, a race condition in a supervisory control and data acquisition (SCADA) program caused a loss of information integrity. This led to alarm and notification failures and ultimately caused the Northeast Blackout of 2003. In excess of 55 million people were affected by this blackout and more than 100 people died. [13] Similarly, a sensor or metrology malfunction can lead to corrupted values in databases, logs, or other repositories.

Examples of integrity loss that may have an impact on the physical system include:

- Data corruption of alarm thresholds or control setpoints may lead to poor production quality in products or, in the extreme case, damage and destruction to physical manufacturing equipment.
- A loss of integrity of telemetry data may cause control algorithms to produce erroneous or even detrimental commands to manufacturing or control equipment.
- Corrupted routing tables or a denial-of-service attack on the communications infrastructure may cause the manufacturing processes to enter into a fail-safe state, thus inhibiting production. If the process is not designed to be fail-safe, an attack could result in equipment damage and lead to a greater disaster.
- Unauthorized remote access to the plant network could enable an attacker to stop production or operate the plant and equipment beyond its intended operating range. An attacker succeeding in disabling the safety instrument systems or changing its threshold parameters—operating the plant beyond its intended range—could lead to severe equipment damage.

3.4.2 Vulnerabilities

A vulnerability as defined in NISTIR 7298, Glossary of Key Information Security Terms [12] is a "weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source."

As indicated in Section 1 of this document, when IT and OT environments are integrated, each domain inherits the vulnerabilities of the other. Increasing complexity of the interfaces typically results in the vulnerability of the overall system being much greater than the sum of the vulnerabilities of the subsystems.

NIST SP 800-82 categorizes ICS vulnerabilities into the following categories with examples [14]:

- **Policy and Procedure**: incomplete, inappropriate, or nonexistent security policy, including its documentation, implementation guides (e.g., procedures), and enforcement
- **Architecture and Design**: design flaws, development flaws, poor administration, and connections with other systems and networks
- **Configuration and Maintenance**: misconfiguration and poor maintenance
- **Physical**: lack of or improper access control, malfunctioning equipment
- **Software Development**: improper data validation, security capabilities not enabled, inadequate authentication privileges
- **Communication and Network**: nonexistent authentication, insecure protocols, improper firewall configuration

The first step in understanding the vulnerabilities and securing an organization's ICS infrastructure is knowledge of deployed assets and their interfaces. The knowledge of an asset's location and baselining its behavior enable detection of anomalous behavior, via network monitoring, that may be the result of a successfully exploited vulnerability. The ability to reliably detect changes in asset behavior and knowing an asset's attributes are key in responding to potential cybersecurity incidents.

3.4.3 Risk

The risk to an organization is the intersection of:

- the vulnerabilities and threats to the organization
- the likelihood that the vulnerability and threat event will be realized
- the impact to the organization should the event be realized

A meaningful risk assessment must be performed in the context of the cyber-ecosystem and the impact to an organization should a loss or degradation occur. The usefulness of the risk assessment is limited by how well the organization identifies and prioritizes the criticality of its assets, identifies the threats, and estimates the likelihood of the threats being realized.

Though risk analysis is a mature discipline, careful deliberations and analyses are necessary to determine the effect integrating IT and OT assets has on the threats, vulnerabilities, and impact to the organization. Once a baseline risk assessment has been completed, information assurance controls, such as the integrity protection measures investigated in this project, can be evaluated on how well they reduce the likelihood of the threat and subsequent reduction of risk. Cybersecurity stakeholders are strongly encouraged to leverage the NIST *Cybersecurity Framework* and manufacturing overlays to identify the components, elements, or items for which a risk assessment must be conducted. In addition, NIST SP 800-82 [14] mentions special considerations for performing an ICS risk assessment.

3.4.4 Security Control Map

Implementation of cybersecurity architectures is most effective when executed in the context of an overall cybersecurity framework. Frameworks include a holistic set of activities or functions (i.e., what needs to be done) and a selection of controls (i.e., how these are done) that are appropriate for a given cyber-ecosystem. For this project, the NIST *Cybersecurity Framework* provided the overarching framework.

The subset of NIST *Cybersecurity Framework* Functions, Categories, and Subcategories that are supported by this example solution are listed below in Table 3-1, along with the subset of mappings to *NIST SP 800-53 Rev. 5* and to the *National Initiative for Cybersecurity Education (NICE) Workforce Framework*. *NIST SP 800-53 Rev 5: Security and Privacy Controls for Information Systems and Organizations* provides a list of controls for protecting operations, assets, and individuals. The controls detail requirements necessary to meet organizational needs. The *NICE Cybersecurity Workforce Framework* identifies knowledge, skills, and abilities needed to perform cybersecurity tasks. It is a reference guide on how to recruit and retain talent for various cybersecurity roles.

For more information on the security controls, the *NIST SP 800-53 Rev.5, Security and Privacy Controls for Information Systems and Organizations* is available at https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final.

For more information about NICE and resources that are available to employers, education and training providers, students, and job seekers, the *NIST SP-181 Rev. 1, NICE Cybersecurity Workforce Framework*, and other NICE resources are available at https://nist.gov/itl/applied-cybersecurity/nice/nice-framework-resource-center.

Table 3-1 Security Control Map

Function	Category	Subcategory	NIST SP 800-53 Rev. 5	NIST SP 800-181 Rev. 1 (NICE Framework) Work Roles
PROTECT (PR)	Identity Management, Authentication, and Access Control (PR.AC): Access to physical and logical assets and associated facilities is limited to authorized users, processes, and devices, and is managed consistent with the assessed risk of unauthorized access to authorized activities and transactions.	PR.AC-1: Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes	IA-2, IA-4, IA-5, IA-7, IA-9, IA-10, IA-12	SP-DEV-001, OM-ADM-001, OV-PMA-003
		PR.AC-3: Remote access is managed	AC-17, AC-19	SP-SYS-001, OM-ADM-001, PR-INF-001
		PR.AC-4: Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties	AC-2, AC-3, AC-14, AC-24	OM-STS-001, OM-ADM-001
		PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks)	AC-14, IA-2, IA-4, IA-5	OM-STS-001, OM-ADM-001
	Data Security (PR.DS): Information and records	PR.DS-1: Data-at-rest is protected	MP-7, SC-28	SP-DEV-002, SP-SYS-002, OM-DTA-001

Function	Category	Subcategory	NIST SP 800-53 Rev. 5	NIST SP 800-181 Rev. 1 (NICE Framework) Work Roles
	(data) are managed consistent with the organization's risk strategy to protect the confidentiality, integrity, and availability of information.	PR.DS-6: Integrity checking mechanisms are used to verify software, firmware, and information integrity	SI-7	OM-DTA-001
	Information Protection Processes and Procedures (PR.IP): Security policies (that address purpose, scope, roles, responsibilities, management commitment, and coordination among organizational entities), processes, and procedures are maintained and used to manage protection of information systems and assets.	PR.IP-4: Backups of information are conducted, maintained, and tested	CP-9	SP-SYS-001, SP-SYS-002, OM-DTA-001
	Maintenance (PR.MA): Maintenance and repairs of industrial control and information system components is performed consistent with policies and procedures.	PR.MA-1: Maintenance and repair of organizational assets are performed and logged, with approved and controlled tools	MA-3	SP-SYS-001, OM-ANA-001
		PR.MA-2: Remote maintenance of organizational assets is approved, logged, and performed in a manner that prevents unauthorized access.	MA-4	SP-SYS-001, OM-ANA-001
DETECT (DE)	Anomalies and Events (DE.AE): Anomalous activity is detected in a timely manner and the potential impact of events is understood.	DE.AE-1: A baseline of network operations and expected data flows for users and systems is established and managed	CM-2, SI-4	SP-ARC-001, PR-CDA-001
		DE.AE-2: Detected events are analyzed to understand attack targets and methods	CA-7, SI-4 RA-5	OM-DTA-002, PR-CDA-001, CO-OPS-001
		DE.AE-3: Event data are collected and correlated from multiple sources and sensors	CA-7, SI-4	OM-DTA-002, PR-CDA-001,

Function	Category	Subcategory	NIST SP 800-53 Rev. 5	NIST SP 800-181 Rev. 1 (NICE Framework) Work Roles
				PR-CIR-001, CO-OPS-001
	Security Continuous Monitoring (DE.CM): The information system and assets are monitored at discrete intervals to identify cybersecurity events and verify the effectiveness of protective measures.	DE.CM-1: The network is monitored to detect potential cybersecurity events	AU-12, CA-7, CM-3, SC-7, SI-4	OM-NET-001, PR-CDA-001, PR-CIR-001
		DE.CM-3: Personnel activity is monitored to detect potential cybersecurity events	AU-12, CA-7, CM-11	PR-CDA-001, AN-TWA-001
		DE.CM-7: Monitoring for unauthorized personnel, connections, devices, and software is performed	AU-12, CA-7, CM-3, SI-4	PR-CDA-001, PR-CIR-001, AN-TWA-001, CO-OPS-001

3.5 Technologies

Table 3-2 lists the capabilities demonstrated in this project, the products, and their functions, along with a mapping of the capabilities to the NIST *Cybersecurity Framework*. Refer to Table 3-1 for an explanation of the NIST *Cybersecurity Framework* subcategory codes.

Table 3-2 Products and Technologies

Capability	Product	Function	NIST *Cybersecurity Framework* Subcategories Mapping
AAL	VMWare Carbon Black		

Capability	Product	Function	NIST *Cybersecurity Framework* Subcategories Mapping
	Windows Software Restriction Policies (SRP) (Note: This component was not provided by collaborator. It is a feature of the Windows operating system product.)	Allow approved ICS applications to execute.	DE.AE-2, DE.AE-3, DE.CM-3, DE.CM-7
File Integrity Checking	GreenTec WORMdisk and ForceField	Provides immutable storage for data, system, and configuration files.	PR.DS-1, PR.IP-4, PR.MA-1
	VMWare Carbon Black	Provides integrity checks for files and software.	PR.DS-6, PR.MA-1, DE.AE-2, DE.CM-3
	Wazuh Security Onion (Note: This component was not provided by collaborator. It is an open source product.)		
BAD, Hardware/ Software/ Firmware Modification Detection	Microsoft Azure Defender for IoT	Passively scans the OT network to create a baseline of devices and network traffic. Alerts when activity deviates from the baseline.	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
	Dragos Platform		
	Forescout eyeInspect (formerly SilentDefense)		
	Tenable Tenable.ot		
	PI System	Collects, analyzes, and visualizes time-series data from multiple sources. Alerts when activity deviates from the baseline.	PR.IP-4, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3

Capability	Product	Function	NIST *Cybersecurity Framework* Subcategories Mapping
User Authentication and User Authorization	TDi ConsoleWorks	Provides a central location for managing password changes.	PR.AC-1, PR.AC-3, PR.AC-4, PR.MA-1, PR.MA-2, DE.AE-2, DE.AE-3, DE.CM-3, DE.CM-7
	Dispel	Provides a security perimeter for all devices within the OT environment.	
Remote Access	Dispel	Provides secure remote access. Records and logs user activity for each session.	PR.AC-3, PR.MA-2, DE.AE-2, DE.CM-7
	Cisco AnyConnect (Note: This component was not provided by collaborator. It was a component of the existing lab infrastructure.)		

4 Architecture

These mechanisms and technologies were integrated into the existing NIST CSMS lab environment [8]. This cybersecurity performance testbed for ICS is comprised of the Process Control System (PCS) and the Collaborative Robotic System (CRS) ICS environments along with additional networking capabilities to emulate common manufacturing environments.

Typically, manufacturing organizations have unique cyber-ecosystems and specific needs for their operation. To demonstrate the modularity and interoperability of the provided solutions, this project used available CRADA partner technologies to assemble four "builds" deployed across both the PCS and CRS. Additionally, to increase the diversity of technologies between builds, two of the builds also utilized open source solutions (Security Onion Wazuh), native operating system features (Windows SRP), and a Cisco Adaptive Security Appliance (ASA) device configured with the AnyConnect virtual private network (VPN) client.

This modular approach, focusing on specific products and outcomes, demonstrates how solutions might be tailored to the operating environment. Table 4-1 provides a summary of the four builds and how the products were distributed across them. Detailed descriptions of the installation, configuration, and integration of these builds are included in Volume C of this guide.

Table 4-1 Summary of What Products Were Used in Each Build

Capability	Build 1	Build 2	Build 3	Build 4
	PCS		**CRS**	
AAL	Carbon Black	Windows SRP	Windows SRP	Carbon Black
BAD, Hardware/Software/Firmware Modification Detection	PI Server	PI Server	PI Server	PI Server
	Tenable.ot	eyeInspect	Dragos	Azure Defender for IoT
File Integrity Checking	Carbon Black	Wazuh	Wazuh	Carbon Black
	ForceField, WORMdisk	ForceField, WORMdisk	ForceField, WORMdisk	ForceField, WORMdisk
User Authentication and Authorization	ConsoleWorks	Dispel	ConsoleWorks	Dispel
Remote Access	AnyConnect	Dispel	AnyConnect	Dispel

Sections 4.1, 4.2, 4.3, and 4.4, present descriptions of the manufacturing processes and control systems of the testbed that are used for demonstrating the security capabilities required for protecting information and system integrity in ICS environments. Section 4.5 describes the network and security architectures that are used to implement the above security capabilities.

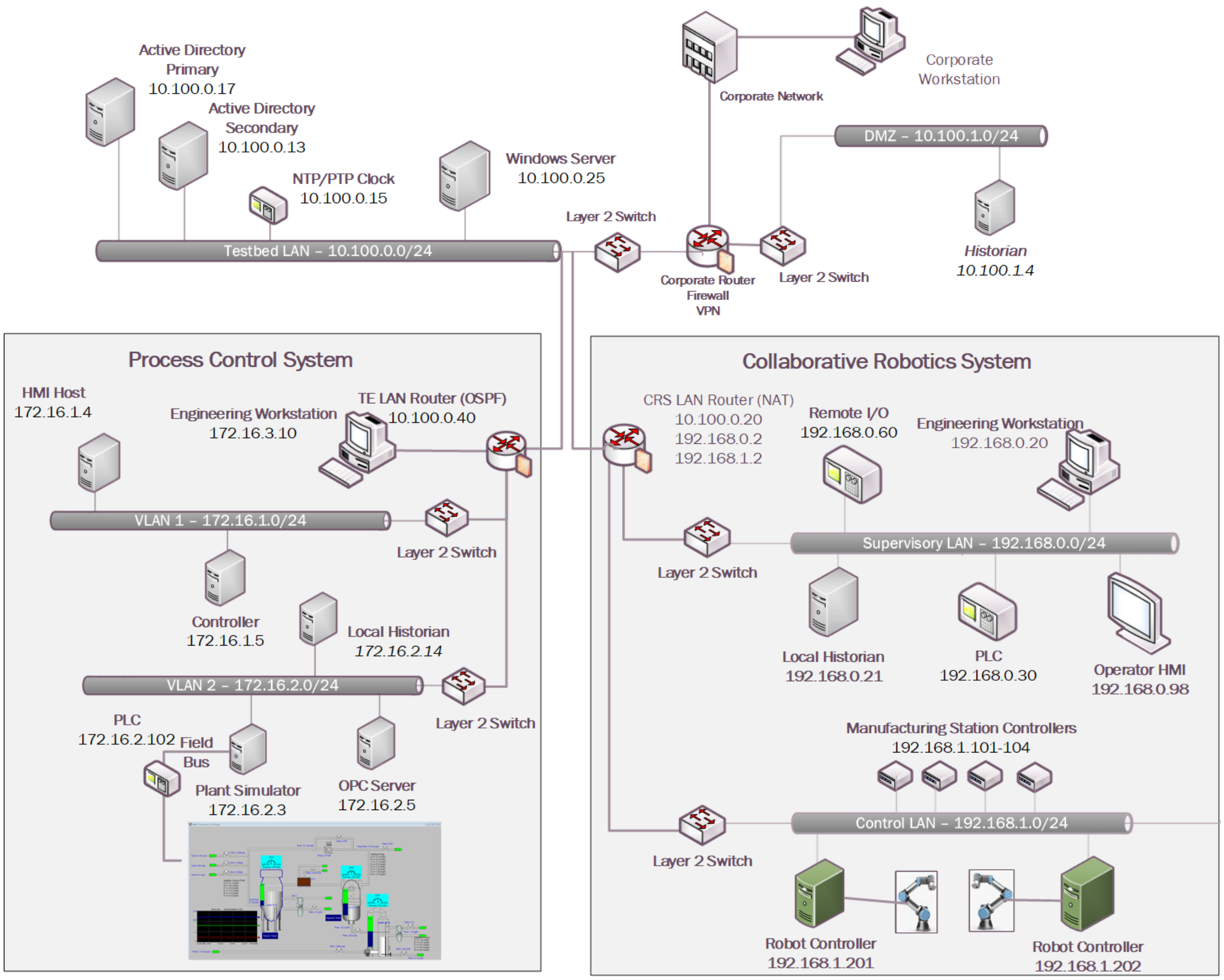
4.1 Manufacturing Process and Control System Description

The CSMS demonstration environment emulates real-world manufacturing processes and their ICS by using software simulators and commercial off-the-shelf (COTS) hardware in a laboratory environment [8]. The CSMS environment was designed to measure the performance impact on ICS that is induced by cybersecurity technologies. For this effort, the CSMS and the integrated PCS and CRS are used to demonstrate the information and system integrity capabilities and are described in Sections 4.3 and 4.4.

4.2 Cybersecurity for Smart Manufacturing Systems Architecture

Figure 4-1 depicts a high-level architecture for the demonstration environment consisting of a testbed local area network (LAN), a demilitarized zone (DMZ), the PCS, and the CRS. The environment utilizes a combination of physical and virtual systems and maintains a local network time protocol server for time synchronization. Additionally, the environment utilizes virtualized Active Directory servers for domain services. The tools used to support information and system integrity are deployed and integrated in the DMZ, Testbed LAN, PCS, and CRS according to vendor recommendations and standard practices as described in the detailed sections for each build.

Figure 4-1: CSMS Network Architecture



4.3 Process Control System

A continuous manufacturing process is a type of manufacturing process that produces or processes materials continuously and in which the materials are continuously moving, going through chemical reactions, or undergoing mechanical or thermal treatment. Continuous manufacturing usually implies a 24-hours a day, seven days a week (24/7) operation with infrequent maintenance shutdowns. Examples of continuous manufacturing systems are chemical production, oil refining, natural gas processing, and wastewater treatment.

The PCS emulates the Tennessee-Eastman (TE) chemical reaction process. The TE problem, presented by Downs and Vogel [15], is a well-known process-control problem in continuous chemical manufacturing. A control loop is required in the PCS to maintain a steady and stable chemical production. The PCS presents a real-world scenario in which a cybersecurity attack could represent a real risk to human safety, environmental safety, and economic viability. This allows the PCS to be used to assess the impact of cybersecurity attacks on the continuous process manufacturing environment.

The PCS includes a software simulator to emulate the TE chemical reaction process. The simulator is written in C code and is executed on a workstation-class computer. In addition, the system includes a series of COTS hardware, including an Allen-Bradley ControlLogix 5571 PLC, a software controller implemented in MATLAB for process control, a Rockwell FactoryTalk Human Machine Interface (HMI), an object linking and embedding for process control (OPC) data access (DA) server, a data historian, an engineering workstation, and several virtual LAN (VLAN) switches and network routers. Figure 4-2 and Figure 4-3 outline the process flow of the TE manufacturing process. The simulated TE process includes five major units with multiple input feeds, products, and byproducts that has 41 measured variables (sensors) and 12 manipulated variables (actuators). The PCS consists of a software simulated chemical manufacturing process (TE process), integrated with a series of COTS hardware, including PLCs, industrial network switches, protocol converters, and hardware modules to connect the simulated process and the control loop.

Figure 4-2 Simplified Tennessee Eastman Process Model

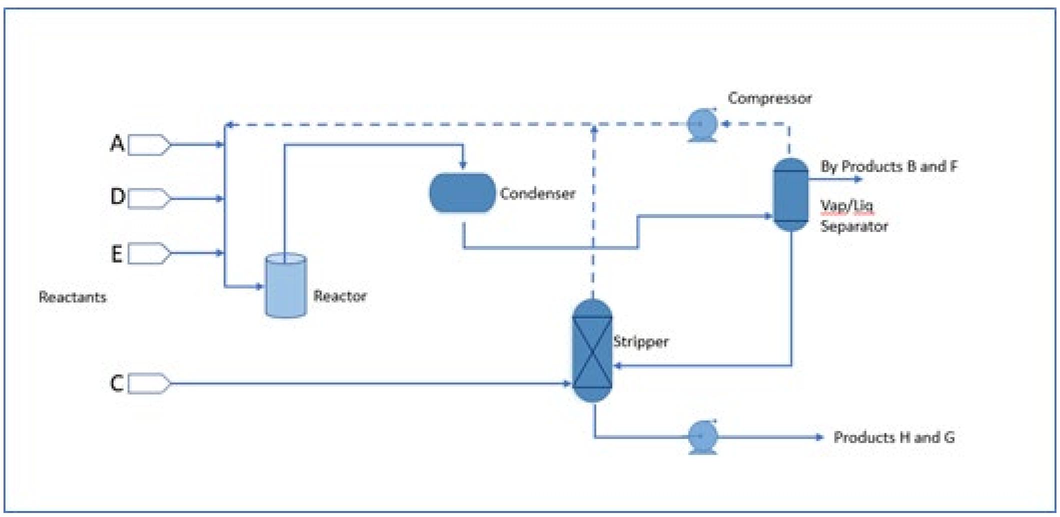
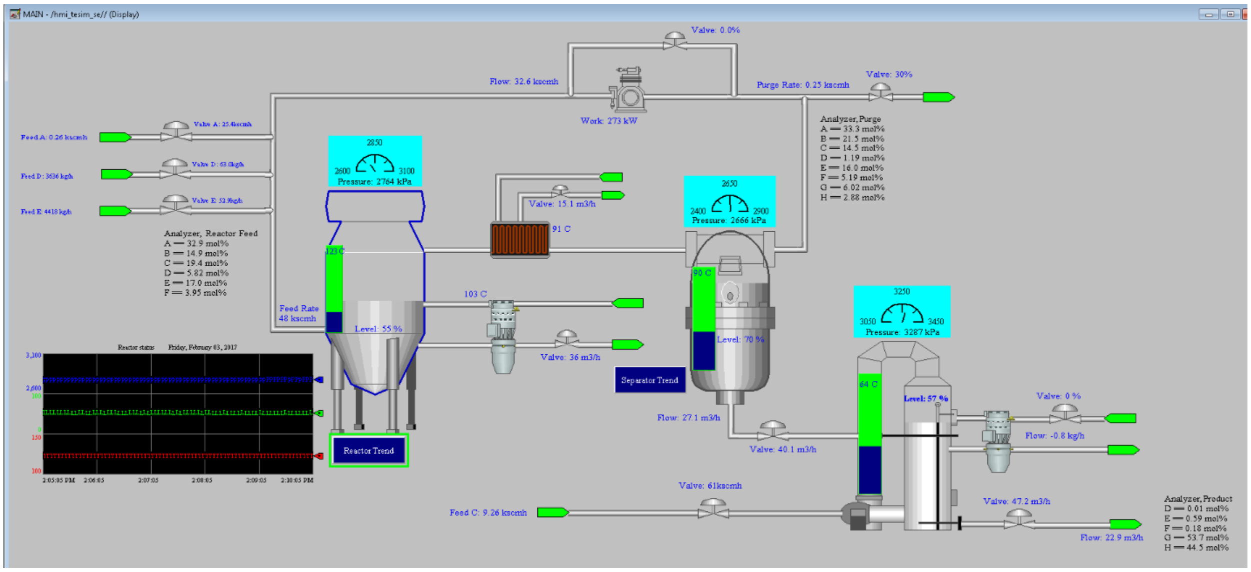


Figure 4-3 HMI Screenshot for the PCS Showing the Main Components in the Process

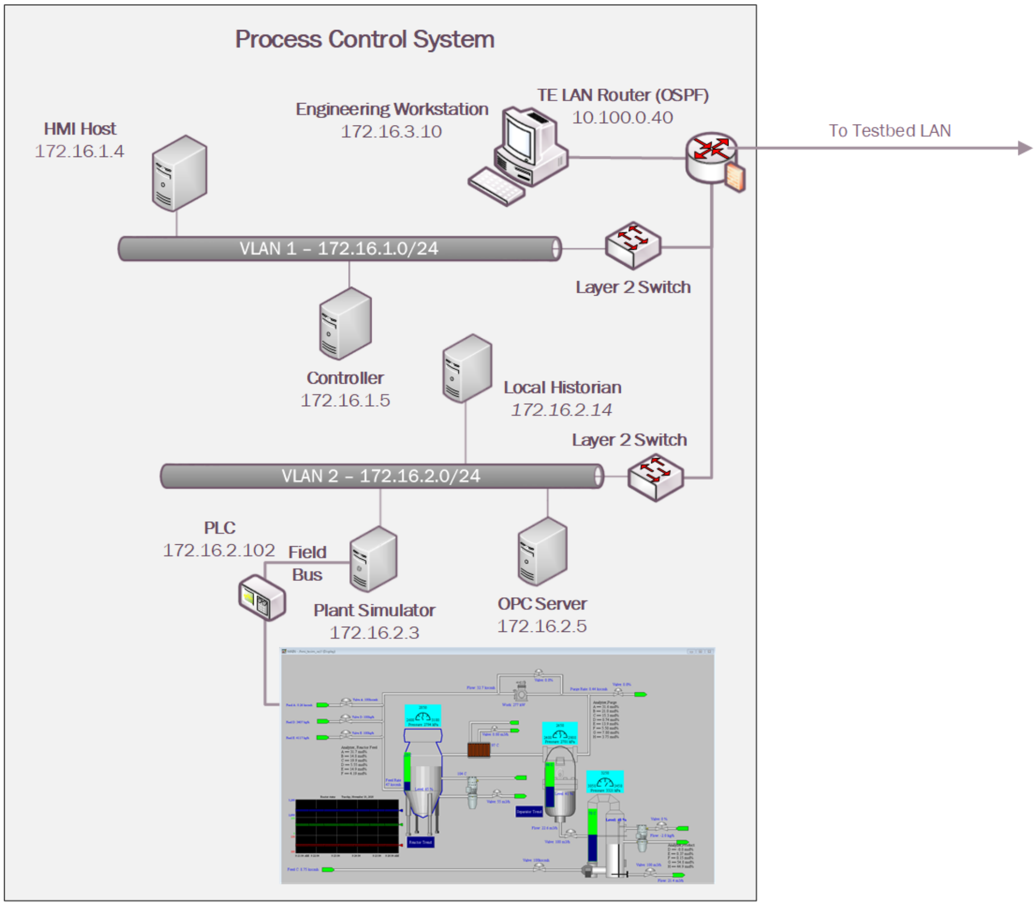


The PCS network architecture is shown in Figure 4-4. The PCS network is connected to the Testbed LAN via a boundary router. The boundary router is an Allen-Bradley Stratix 8300. All network traffic is going through the boundary router to access the Testbed LAN and the DMZ. The PCS environment is segmented into three local networks, namely the engineering LAN, Operations LAN (VLAN1), and the Supervisory LAN (VLAN2). Each of these local networks is connected using an industrial network switch, an Allen-Bradley Stratix 5700. The engineering workstation is hosted in the engineering LAN. The HMI and the Plant Controller are hosted in the operations LAN. The Plant Simulator is hosted in the supervisory LAN along with the Local Historian, OPC Server, and the Supervisory PLC.

The Operations LAN (VLAN1) simulates a central control room environment. The supervisory LAN (VLAN2) simulates the process operation/ manufacturing environment, which typically consists of the operating plant, PLCs, OPC server, and data historian.

An OPC DA server is the main data gateway for the PLC and the simulated controller. The PLC reads in the manufacturing process sensor data from the Plant Simulator using the DeviceNet connection and communicates the data to the OPC DA server. The PLC also retrieves actuator information from the controller through the OPC DA and transmits to the Plant Simulator. The controller uses a MATLAB Simulink interface to communicate with the OPC DA server directly.

Figure 4-4 PCS Network



4.4 Collaborative Robotics System (CRS)

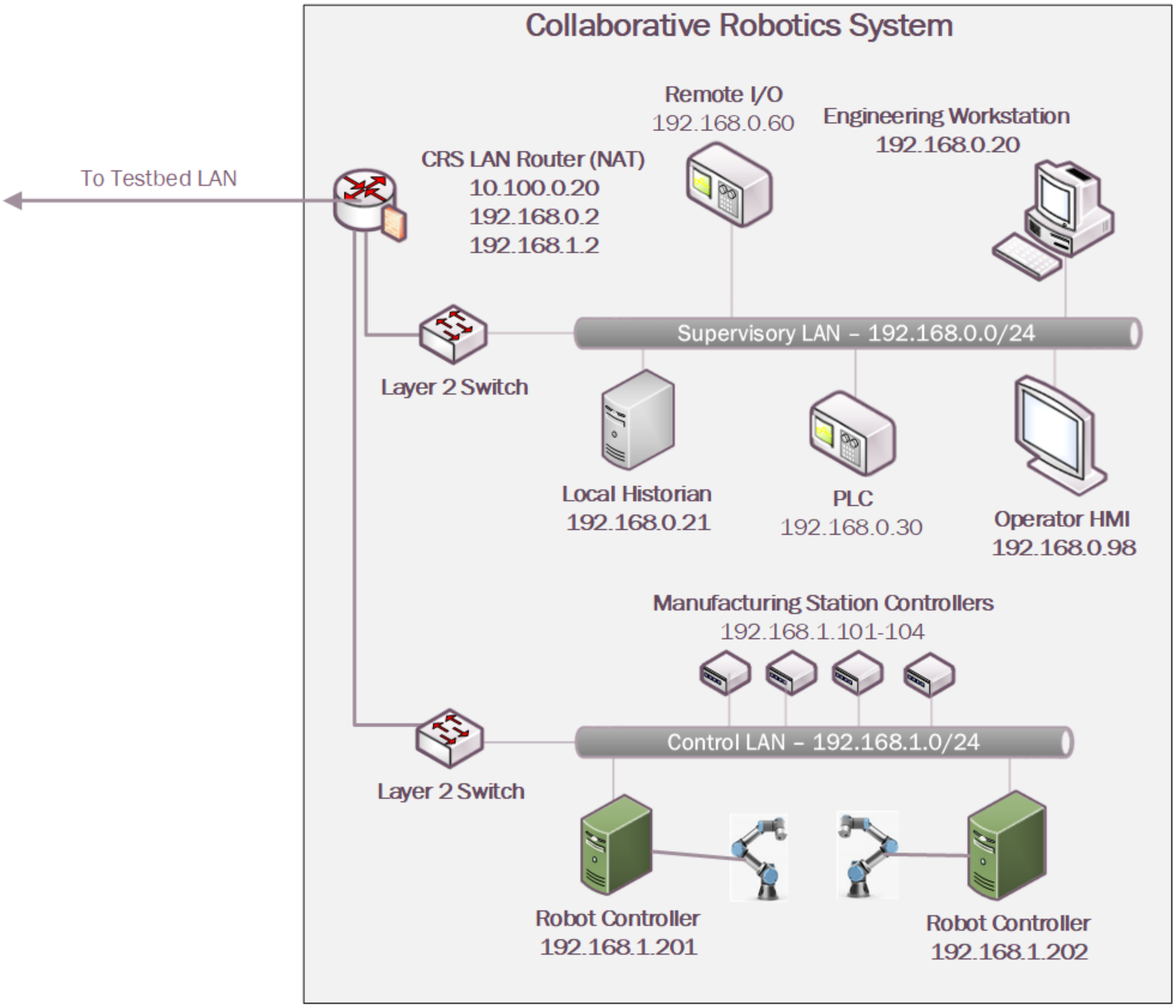
The CRS workcell, shown in Figure 4-5, contains two robotic arms that perform a material handling process called machine tending [8]. Robotic machine tending utilizes robots to interact with machinery, performing physical operations a human operator would normally perform (e.g., loading and unloading of parts in a machine, opening and closing machine doors, activating operator control panel buttons, etc.).

Parts are transported by two Universal Robots UR3e robotic arms through four simulated machining stations. Each station communicates with the Supervisory PLC (a Beckhoff CX9020) over the workcell network, which monitors and controls all aspects of the manufacturing process. An HMI (Red Lion G310) allows the workcell operator to monitor and control process parameters.

Figure 4-5 The CRS Workcell

The CRS network, shown in Figure 4-6, is hierarchically architected, separating the supervisory devices from the low-level OT that control the manufacturing process. The top-level router is a Siemens RUGGEDCOM RX1510, which provides firewall capabilities, logical access to the Testbed LAN network, network address translation (NAT), and other cybersecurity capabilities. The router is connected to the Testbed LAN (identified in Figure 4-1 as the Testbed LAN) using NAT. Layer 2 network traffic for the Supervisory LAN is handled by a Netgear GS724T-managed Ethernet switch, and network traffic for the Control LAN is handled by a Siemens i800-managed Ethernet switch.

Figure 4-6 CRS Network



4.5 Logical Network and Security Architectures

The following sections provide a high-level overview of the technology integration into the ICS environments for each solution, also referred to as a build. Additional details related to the installation and configuration of these tools are provided in Volume C of this guide.

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

4.5.1 Build 1

For Build 1, the technologies in Table 4-2 were integrated into the PCS environment, Testbed LAN, and DMZ segments of the testbed environment to enhance system and information integrity capabilities.

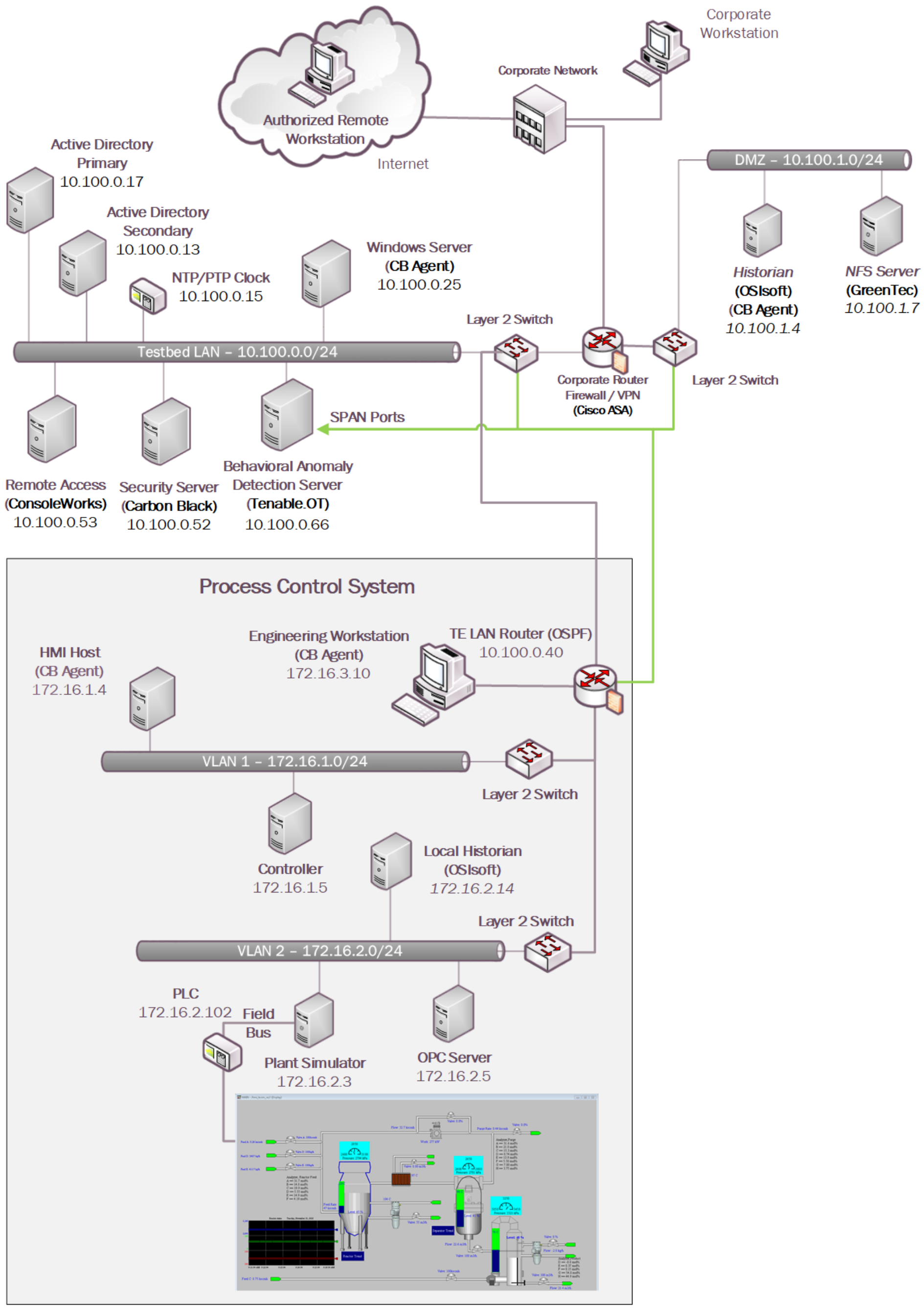
Table 4-2 Build 1 Technology Stack to Capabilities Map

Capability	Products	Description
AAL	Carbon Black	Carbon Black Server is deployed within the Testbed LAN with the Carbon Black Agents installed on key workstations and servers in the Testbed LAN, PCS environment, and DMZ to control application execution.
BAD, Hardware/Software/Firmware Modification Detection	PI Server	Deployed in the DMZ and PCS environments, the PI Server provides the historian repository for process data through its Data Archive and generates Event Frames upon detection of abnormal manufacturing system behavior.
	Tenable.ot	Passively monitors the PCS network, Testbed LAN, and DMZ for abnormal network activity via SPAN ports, and is also configured to capture detailed asset information for supporting inventory, change via both passive and active scanning.
File Integrity Checking	Carbon Black	Deployed within the Testbed LAN environment with the Carbon Black Agents installed on key workstations and servers to monitor the integrity of local files.
	ForceField, WORMdisk	A GreenTec fileserver is added to the DMZ environment and configured with both a ForceField and WORM drive to provide a protected archive for the historian data and the approved versions of configuration, source (PLC Programs), and executable files for the ICS environment.

Capability	Products	Description
User Authentication and Authorization	ConsoleWorks	Deployed to centralize the access and management of the systems and credentials. ConsoleWorks is deployed to the Testbed LAN to allow connections to the PCS environment.
Remote Access	AnyConnect	Supports authenticated VPN connections to the environment with limited access to only the TDI ConsoleWorks web interface.

The technology was integrated into the lab environment as shown in Figure 4-7.

Figure 4-7 Build 1, PCS Complete Architecture with Security Components



4.5.2 Build 2

For Build 2, the technologies in Table 4-3 were integrated into the PCS, Testbed LAN, and DMZ segments of the testbed environment to enhance system and information integrity capabilities.

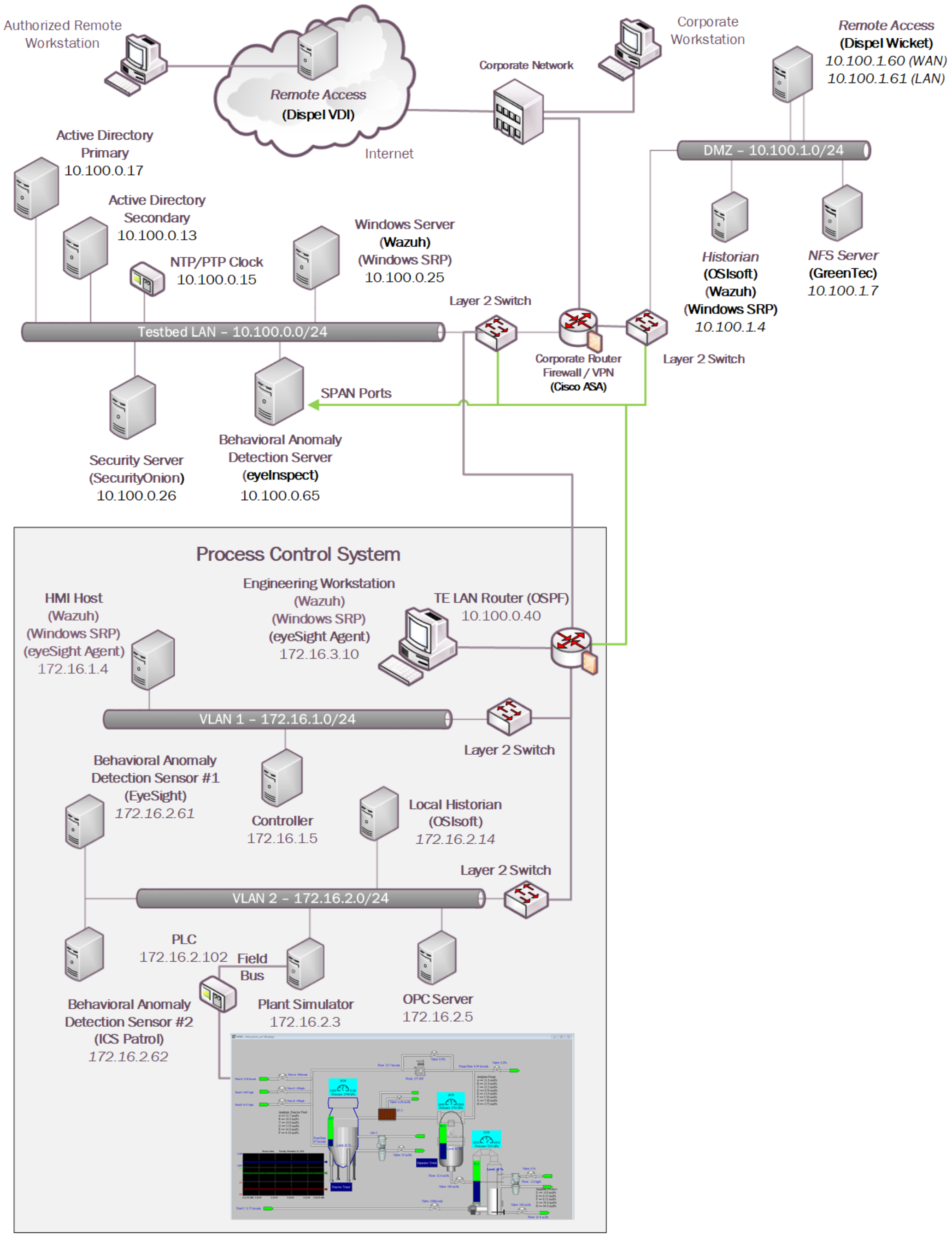
Table 4-3 Build 2 Technology Stack to Capabilities Map

Capability	Product	Description
AAL	Windows SRP	AD Group Policy Objects (GPOs) are used to configure and administer the Windows Software Restriction Policy (SRP) capabilities within the Testbed LAN environment and PCS environments. For non-domain systems (e.g., Dispel VDI and DMZ systems), the GPO was applied as local settings on the systems.
BAD, Hardware/Software/Firmware Modification Detection	PI Server	Deployed in the DMZ and PCS environments, the PI Server provides the historian repository for process data through its Data Archive and generates Event Frames upon detection of abnormal manufacturing system behavior.
	eyeInspect ICSPatrol	Passively monitors the PCS network, Testbed LAN, and DMZ for abnormal network activity via SPAN ports, and is also configured to capture detailed asset information for supporting inventory and change management capabilities using the ICSPatrol server, which can perform scans on ICS components.
File Integrity Checking	Wazuh	The Security Onion server is used to manage and monitor the integrity of local files using the Wazuh agents deployed on the Dispel VDI, DMZ, Testbed LAN, and PCS.
	ForceField, WORMdisk	A GreenTec fileserver is added to the DMZ environment and configured with both a ForceField and WORM drive to provide a protected archive for the historian data and the approved versions of configuration, source, and executable files for the ICS environment.

Capability	Product	Description
User Authentication and Authorization	Dispel	The Dispel Wicket is deployed to the DMZ environment and integrated with the Dispel cloud-based environment to provide a virtual desktop interface (VDI) with a secure remote connection to the testbed environment. Through this connection, authorized users are permitted to access resources in both the Testbed LAN and PCS environment.
Remote Access		

The technology was integrated into the lab environment as shown in Figure 4-8.

Figure 4-8 Build 2, PCS Complete Architecture with Security Components



This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

4.5.3 Build 3

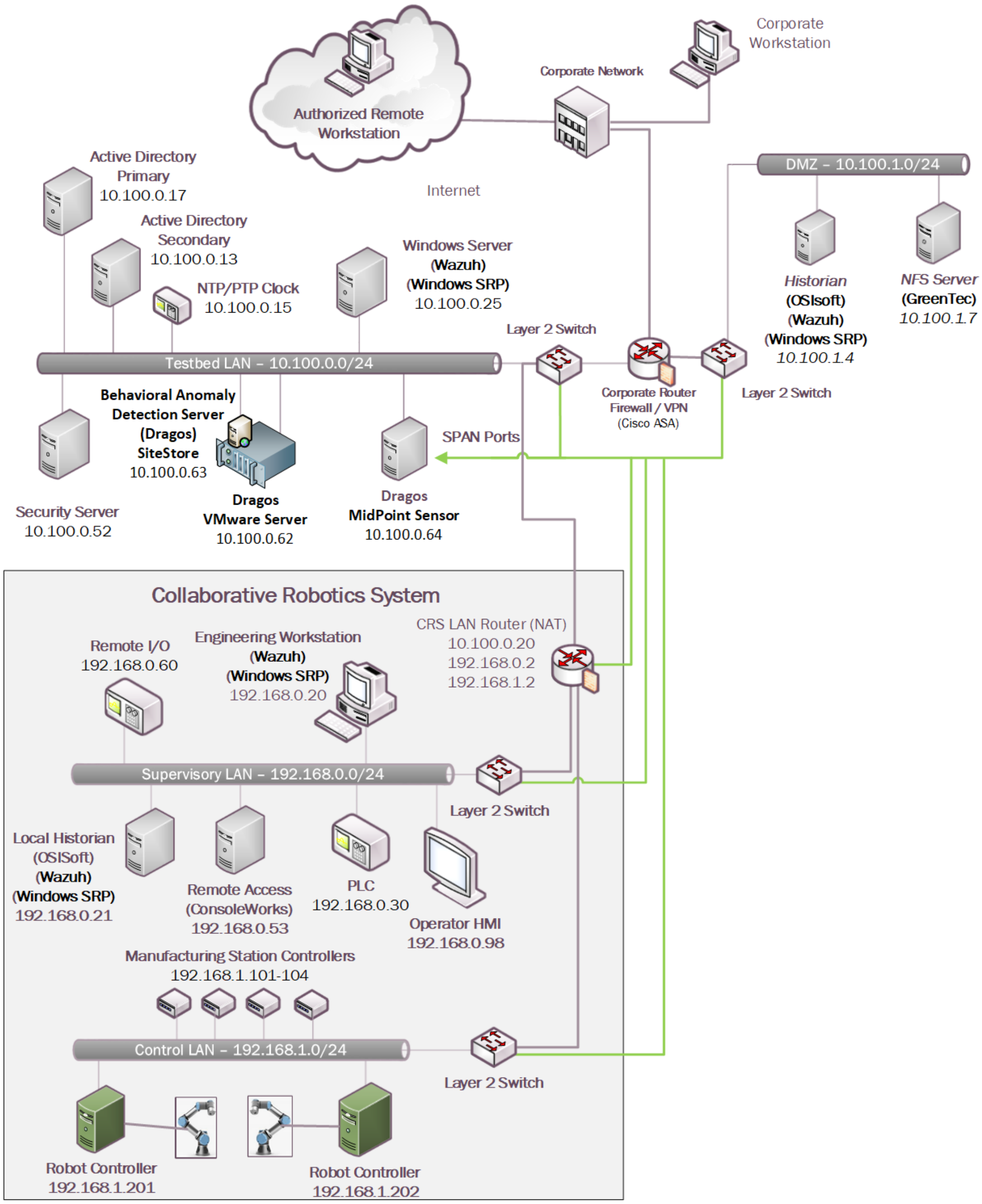
The technologies in Table 4-4 were integrated into the CRS for Build 3 to enhance system and data integrity capabilities.

Table 4-4 Build 3 Technology Stack to Capabilities Map

Capability	Products	Description
AAL	Windows SRP	AD Group Policy Objects (GPOs) are used to configure and administer the Windows Software Restriction Policy (SRP) capabilities within the Testbed LAN environment and CRS environments.
BAD, Hardware/Software/Firmware Modification Detection	PI Server	Deployed in the DMZ and CRS environments, the PI Server provides the historian repository for process data through its Data Archive and generates Event Frames upon detection of abnormal manufacturing system behavior
	Dragos	Passively monitors the CRS network, Testbed LAN, and DMZ for abnormal network activity via SPAN ports and receives Event Frames from the DMZ PI system through the PI Web API interface.
File Integrity Checking	Wazuh	The Security Onion server is used to manage and monitor the integrity of local files using the Wazuh agents deployed on the DMZ, Testbed LAN, and CRS.
	ForceField, WORMdisk	A GreenTec fileserver is added to the DMZ environment and configured with both a ForceField and WORM drive to provide a protected archive for the historian data and the approved versions of configuration and coding files for the ICS environment.
User Authentication and Authorization	ConsoleWorks	Deployed to centralize the access and management of the systems and credentials. ConsoleWorks is deployed to allow connections within the CRS environment.
Remote Access	AnyConnect	Supports authenticated VPN connections to the environment with limited access to only the TDI ConsoleWorks web interface.

The technology was integrated into the lab environment as shown in Figure 4-9.

Figure 4-9 Build 3, CRS Complete Architecture with Security Components



4.5.4 Build 4

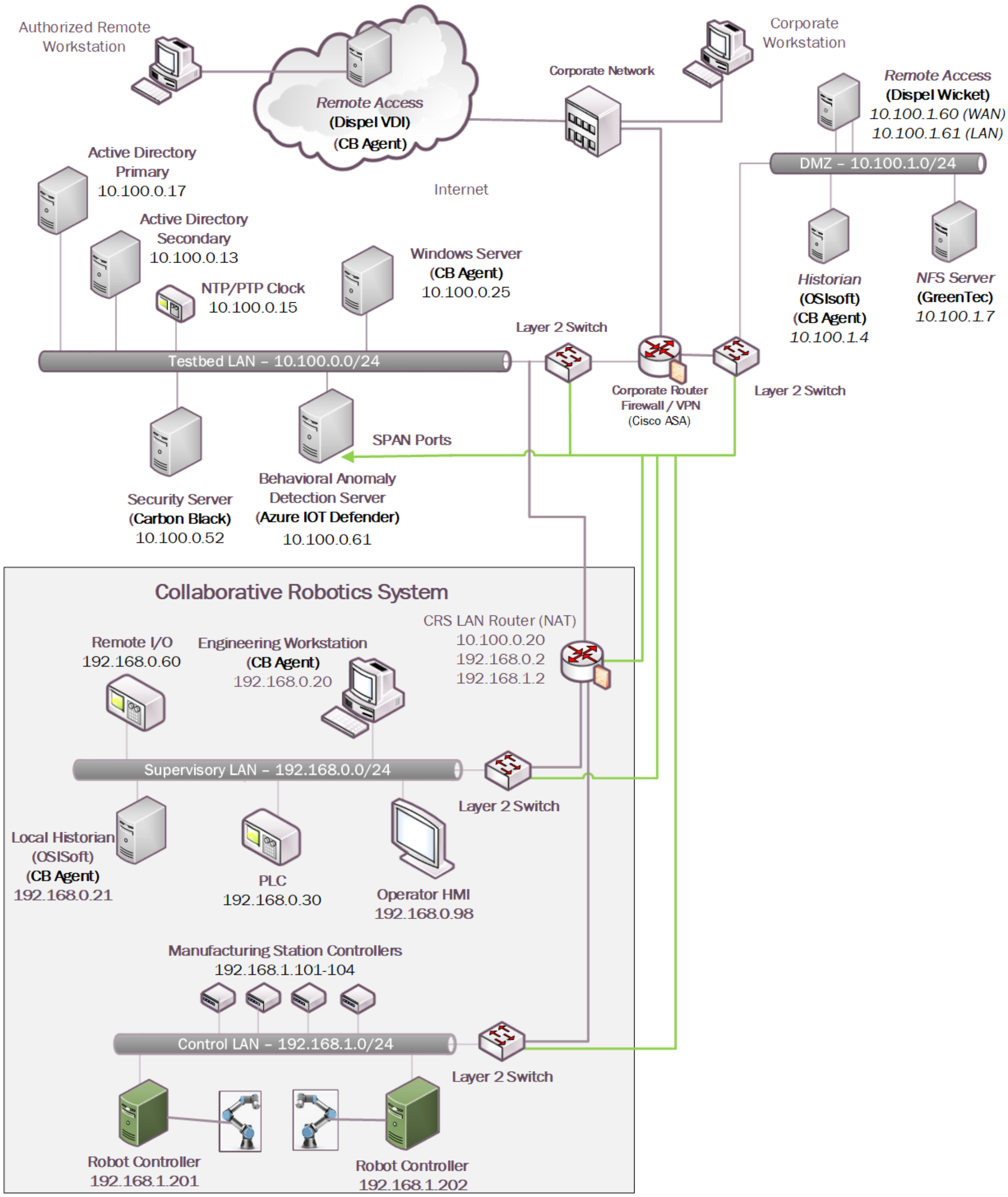
For Build 4, the technologies in Table 4-5 were integrated into the CRS, Testbed LAN, and DMZ segments of the testbed environment to enhance system and data integrity capabilities.

Table 4-5 Build 4 Technology Stack to Capabilities Map

Capability	Products	Description
AAL	Carbon Black	Deployed within the Testbed LAN environment with the Carbon Black agents installed on key workstations and servers to control application execution.
BAD, Hardware/Software/Firmware Modification Detection	PI Server	Deployed in the DMZ and CRS environments, the PI Server provides the historian repository for process data through its Data Archive and generates Event Frames upon detection of abnormal manufacturing system behavior.
	Azure Defender for IoT	Passively monitors the CRS network, Testbed LAN, and DMZ for abnormal network activity via SPAN ports and is also configured to capture detailed asset information for supporting inventory and change management capabilities.
File Integrity Checking	Carbon Black	Deployed within the Testbed LAN environment with the Carbon Black agents installed on key workstations and servers to monitor the integrity of local files.
	ForceField, WORMdisk	A GreenTec fileserver is added to the DMZ environment and configured with both a ForceField and WORM drive to provide a protected archive for the historian data and the approved versions of configuration and coding files for the ICS environment.
User Authentication and Authorization **Remote Access**	Dispel	The Dispel Wicket is deployed to the DMZ environment and integrated with the Dispel cloud-based environment to provide a VDI with a secure remote connection to the testbed environment. Through this connection, authorized users are permitted to access resources in both the Testbed LAN and CRS environment.

The technology was integrated into the lab environment as shown in Figure 4-10.

Figure 4-10 Build 4, CRS Complete Architecture with Security Components



5 Security Characteristic Analysis

The purpose of the security characteristic analysis is to understand the extent to which the project meets its objective to demonstrate protecting information and system integrity in ICS environments. In addition, it seeks to understand the security benefits and drawbacks of the example solution.

5.1 Assumptions and Limitations

The security characteristic analysis has the following limitations:

- It is neither a comprehensive test of all security components nor a red-team exercise.
- It cannot identify all weaknesses.
- It does not include the lab infrastructure.

5.2 Example Solution Testing

This section presents a summary of the solution testing and results. A total of eleven tests were developed for the builds. The following information is provided for each scenario:

- **Objective**: Purpose of the scenario and what it will demonstrate
- **Description**: Brief description of the scenario and the actions performed
- **Relevant NIST Cybersecurity Framework Subcategories**: Mapping of NIST *Cybersecurity Framework* subcategories relevant to the scenario
- **Assumptions**: Assumptions about the cyber-environment
- **Security Capabilities and Products**: Capabilities and products demonstrated during the scenario
- **Test Procedures**: Steps performed to execute the scenario
- **Expected Results**: Expected results from each capability and product demonstrated during the scenario, and for each build
- **Actual Test Results**: Confirm the expected results
- **Overall Result**: Were the security capabilities and products able to meet the objective when the scenario was executed (PASS/FAIL rating).

Additional information for each scenario such as screenshots captured during the execution of the test procedures and detailed results from the security capabilities are presented in Appendix D.

5.2.1 Scenario 1: Protect Host from Malware Infection via USB

Objective	This test demonstrates blocking the introduction of malware through physical access to a workstation within the manufacturing environment.
Description	An authorized user transports executable files into the manufacturing system via a USB flash drive that contains malware.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-2, DE.AE-2
Assumptions	▪ User does not have administrative privileges on the target machine. ▪ User has physical access to the target machine.
Security Capabilities and Products	Build 1: ▪ Carbon Black: AAL Build 2: ▪ Windows SRP: AAL Build 3: ▪ Windows SRP: AAL Build 4: ▪ Carbon Black: AAL
Test Procedures	1. Attempt to execute malware on the target machine.
Expected Results	▪ The AAL tool will detect and stop the malware upon execution.
Actual Test Results	▪ The AAL technology successfully blocks and alerts on the execution of the application on the workstation in all builds.
Overall Result	PASS

5.2.2 Scenario 2: Protect Host from Malware Infection via Network Vector

Objective	This test demonstrates the detection of malware introduced from the network.

Description	An attacker pivoting from the corporate network into the manufacturing environment attempts to insert malware to establish persistence in the manufacturing environment.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	▪ The attacker has completed reconnaissance and initial access, gaining the ability to pivot into the manufacturing environment.
Security Capabilities and Products	Build 1: ▪ Carbon Black: AAL ▪ Tenable.ot: BAD Build 2: ▪ Windows SRP: AAL ▪ Forescout eyeInspect: BAD Build 3: ▪ Windows SRP: AAL ▪ Dragos: BAD Build 4: ▪ Carbon Black: AAL ▪ Azure Defender for IoT: BAD
Test Procedures	1. Attacker pivots into the manufacturing environment. 2. Attacker copies malware to the server in Testbed LAN. 3. Attacker attempts to execute malware on server in Testbed LAN.
Expected Results	▪ The AAL capabilities installed on target systems will block execution of the malicious code. ▪ The BAD tool will capture the suspicious traffic and generate an alert.

Actual Test Results	▪ The AAL technology successfully blocks and alerts on the execution of the application on the workstation in all builds. ▪ The BAD tool is able to detect and alert on activity pivoting into manufacturing systems.
Overall Result	PASS

5.2.3 Scenario 3: Protect Host from Malware via Remote Access Connections

Objective	This test demonstrates blocking malware that is attempting to infect the manufacturing system through authorized remote access connections.
Description	A remote workstation authorized to use a remote access connection has been infected with malware. When the workstation is connected to the manufacturing environment through the remote access connection, the malware attempts to pivot and spread to vulnerable host(s).
Relevant NIST *Cybersecurity Framework* Subcategories	PR.AC-1, PR.AC-3, PR.AC-4, PR.AC-7, PR.MA-1, PR.MA-2, DE.CM-3, DE.CM-7
Assumptions	▪ Infection of the remote workstation occurs prior to remote access session.

Security Capabilities and Products	Build 1: ▪ Cisco VPN: Remote Access ▪ ConsoleWorks: User Authentication and User Authorization Build 2: ▪ Dispel: User Authentication and User Authorization, and Remote Access Build 3: ▪ Cisco VPN: Remote Access ▪ ConsoleWorks: User Authentication and User Authorization Build 4: ▪ Dispel: User Authentication and User Authorization, and Remote Access
Test Procedures	1. Authorized remote user connects to the manufacturing environment. 2. Malware on remote host attempts to pivot into the manufacturing environment.
Expected Results	▪ Malware will be blocked from propagation by the remote access capabilities.
Actual Test Results	▪ Remote access connection blocks malware attempts to pivot into the manufacturing environment.
Overall Result	PASS

5.2.4 Scenario 4: Protect Host from Unauthorized Application Installation

Objective	This test demonstrates blocking installation and execution of unauthorized applications on a workstation in the manufacturing system.
Description	An authorized user copies downloaded software installation files from a shared network drive accessible from the workstation in the manufacturing system. The user then attempts to install the unauthorized software on the workstation.

Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	▪ User does not have administrative privileges on the target machine. ▪ Applications to be installed are unapproved applications.
Security Capabilities and Products	Build 1: ▪ Carbon Black: AAL ▪ Tenable.ot: BAD Build 2: ▪ Windows SRP: AAL ▪ eyeInspect: BAD Build 3: ▪ Windows SRP: AAL ▪ Dragos: BAD Build 4: ▪ Carbon Black: AAL ▪ Azure Defender for IoT: BAD
Test Procedures	1. The user copies software to a host in the manufacturing environment. 2. The user attempts to install the software on the host. 3. The user attempts to execute software that does not require installation.
Expected Results	▪ The AAL tool will detect and stop the execution of the software installation or executable file. ▪ The BAD tool will capture the suspicious traffic and generate an alert.
Actual Test Results	▪ The AAL technology successfully blocks and alerts on the execution of the application on the workstation in all builds. ▪ The BAD tool is able to detect and alert on activity in the manufacturing system.

Overall Result	PASS

5.2.5 Scenario 5: Protect from Unauthorized Addition of a Device

Objective	This test demonstrates detection of an unauthorized device connecting to the manufacturing system.
Description	An individual authorized to access the physical premises connects and uses an unauthorized device on the manufacturing network.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	▪ Ports on switch are active and available.
Security Capabilities and Products	Build 1: ▪ Tenable.ot: BAD Build 2: ▪ eyeInspect: BAD Build 3: ▪ Dragos: BAD Build 4: ▪ Azure Defender for IoT: BAD
Test Procedures	1. The individual connects the unauthorized device to the manufacturing network. 2. The individual uses an unauthorized device to access other devices on the manufacturing network.
Expected Results	▪ The BAD detection tool will capture the suspicious traffic and generate an alert.
Actual Test Results	▪ The BAD detection tool is able to detect and alert on activity in the manufacturing system.
Overall Result	PASS

5.2.6 Scenario 6: Detect Unauthorized Device-to-Device Communications

Objective	This test demonstrates detection of unauthorized communications between devices.
Description	A device authorized to be on the network attempts to establish an unapproved connection.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	▪ The environment has a predictable communications pattern.
Security Capabilities and Products	Build 1: ▪ Tenable.ot: BAD. Build 2: ▪ eyeInspect: BAD. Build 3: ▪ Dragos: BAD Build 4: ▪ Azure Defender for IoT: BAD
Test Procedures	1. The device attempts to establish an unapproved connection.
Expected Results	▪ The BAD tool will capture the suspicious traffic and generate an alert.
Actual Test Results	▪ The BAD tool is able to detect and alert on activity in manufacturing systems.
Overall Result	PASS

5.2.7 Scenario 7: Protect from Unauthorized Deletion of Files

Objective	This test demonstrates protection of files from unauthorized deletion both locally and on network file share.

Description	An authorized user attempts to delete files on an engineering workstation and a shared network drive within the manufacturing system.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-1, PR.DS-6, PR.IP-4, PR.MA-1, DE.AE-2
Assumptions	▪ User does not have administrative privileges on the target machine.
Security Capabilities and Products	Build 1: ▪ Carbon Black: File Integrity Checking. ▪ WORMdisk: File Integrity Protection. Build 2: ▪ Security Onion: File Integrity Checking. ▪ WORMdisk: File Integrity Protection. Build 3: ▪ Security Onion: File Integrity Checking. ▪ WORMdisk: File Integrity Protection. Build 4: ▪ Carbon Black: File Integrity Checking. ▪ WORMdisk: File Integrity Protection.
Test Procedures	1. User attempts to delete files located on a workstation in the manufacturing system. 2. User attempts to delete files from the network file share containing the golden images for the manufacturing system.
Expected Results	▪ Deletion of files on the workstation will be detected and alerted on by the file integrity checking tool. ▪ Deletion of files on the network file share will be prevented by the file integrity checking tool.
Actual Test Results	▪ Host-based file integrity checking is able to detect and alert on deletion of files.

	▪ Protected network file share is able to prevent deletion of files on the network file share.
Overall Result	PASS

5.2.8 Scenario 8: Detect Unauthorized Modification of PLC Logic

Objective	This test demonstrates detection of PLC logic modification.
Description	An authorized user performs an unapproved or unauthorized modification of the PLC logic from an engineering workstation.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.AC-3,PR.AC-7, PR.DS-6, PR.MA-1, PR.MA-2, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	None
Security Capabilities and Products	Build 1: ▪ Tenable.ot: BAD and Software Modification ▪ Cisco VPN: Remote Access ▪ ConsoleWorks: User Authentication, User Authorization, and Remote Access Build 2: ▪ eyeInspect: BAD and Software Modification ▪ Dispel: User Authentication and User Authorization, and Remote Access Build 3: ▪ Dragos: BADand Software Modification ▪ Cisco VPN: Remote Access ▪ ConsoleWorks: User Authentication, User Authorization, and Remote Access Build 4: ▪ Azure Defender for IoT: BAD and Software Modification ▪ Dispel: User Authentication and User Authorization, and Remote Access

Test Procedures	1. The authorized user remotely connects to a manufacturing environment. 2. The user modifies and downloads a logic file to the PLC.
Expected Results	▪ The BAD tool will capture the suspicious traffic and generate an alert. ▪ The user authentication/authorization/remote access is able to remotely access the engineering systems as intended.
Actual Test Results	▪ The BAD is able to detect and alert on activity accessing the PLC.
Overall Result	PASS

5.2.9 Scenario 9: Protect from Modification of Historian Data

Objective	This test demonstrates blocking of modification of historian archive data.
Description	An attacker coming from the corporate network pivots into the manufacturing environment and attempts to modify historian archive data.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-2
Assumptions	▪ The attacker has completed reconnaissance and initial access, gaining the ability to pivot into the manufacturing environment.
Security Capabilities and Products	Build 1: ▪ Tenable.ot: BAD ▪ ForceField WFS: File Integrity Protection. Build 2: ▪ eyeInspect: BAD ▪ ForceField WFS: File Integrity Protection.

	Build 3: ▪ Dragos: BAD ▪ ForceField WFS: File Integrity Protection. Build 4: ▪ Azure Defender for IoT: BAD ▪ ForceField WFS: File Integrity Protection.
Test Procedures	1. Attacker pivots into the manufacturing environment from the corporate network. 2. Attacker attempts to delete historian archive data file. 3. Attacker attempts to replace historian archive data file.
Expected Results	▪ The file operations will be blocked by the file integrity checking tool.
Actual Test Results	▪ File integrity checking tool is able to prevent file operations on the protected files.
Overall Result	PASS

5.2.9.1 Scenario 10: Detect Sensor Data Manipulation

Objective	This test demonstrates detection of atypical data reported to the historian.
Description	A sensor in the manufacturing system begins sending atypical data values to the historian.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.IP-4, PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	▪ Devices in the manufacturing system (HMI and PLCs) are not validating sensor data.
Security Capabilities and Products	▪ PI Server: BAD
Test Procedures	1. A sensor sends invalid data to the historian.

Expected Results	▪ The BAD capability will detect atypical sensor data and generate alerts.
Actual Test Results	▪ The BAD tool is able to detect atypical data and create an event frame.
Overall Result	PASS

5.2.9.2 Scenario 11: Detect Unauthorized Firmware Modification

Objective	This test demonstrates detection of device firmware modification.
Description	An authorized user performs a change of the firmware on a PLC.
Relevant NIST *Cybersecurity Framework* Subcategories	PR.DS-6, PR.MA-1, DE.AE-1, DE.AE-2, DE.AE-3, DE.CM-1, DE.CM-3, DE.CM-7
Assumptions	None

Security Capabilities and Products	Build 1: ▪ Cisco VPN: Remote Access. ▪ ConsoleWorks: Remote Access, User Authentication, and User Authorization. ▪ Tenable.ot: BAD and Firmware Modification. Build 2: ▪ Dispel: Remote Access, User Authentication, and User Authorization. ▪ eyeInspect and ICSPatrol: BAD and Firmware Modification. Build 3: ▪ Cisco VPN: Remote Access. ▪ ConsoleWorks: Remote Access, User Authentication, and User Authorization. ▪ Dragos: BAD and Firmware Modification. Build 4: ▪ Dispel: Remote Access, User Authentication, and User Authorization. ▪ Azure Defender for IoT: BAD and Firmware Modification.
Test Procedures	1. Authorized remote user connects to manufacturing environment. 2. The user changes firmware on the PLC component.
Expected Results	▪ The behavioral anomaly detection tool will identify the change to the PLC and generate an alert for review.
Actual Test Results	▪ The BAD is able to detect and generate alerts for updates to PLC component firmware.
Overall Result	PASS

5.3 Scenarios and Findings

One aspect of our security evaluation involved assessing how well the reference design addresses the security characteristics that it was intended to support. The NIST *Cybersecurity Framework* Subcategories were used to provide structure to the security assessment by consulting the specific sections of each standard that are cited in reference to a Subcategory. The cited sections provide validation points that the example solution would be expected to exhibit. Using the NIST *Cybersecurity Framework* Subcategories as a basis for organizing our analysis allowed us to systematically consider how well the reference design supports the necessary security characteristics.

5.3.1 PR.AC-1: Identities and Credentials are Issued, Managed, Verified, Revoked, and Audited for Authorized Devices, Users, and Processes

This NIST *Cybersecurity Framework* Subcategory is supported through user authentication and user authorization capabilities in addition to the native credential management capabilities associated with the tools. In each of the systems, user accounts were issued, managed, verified, revoked, and audited.

5.3.2 PR.AC-3: Remote Access is Managed

This NIST *Cybersecurity Framework* Subcategory is supported by remote access tools integrated with the user authentication and authorization systems. Together, these tools provide a secure channel for an authorized user to access the manufacturing environment from a remote location. These tools are configurable to allow organizations to control who can remotely access the system, what the user can access, and when access is allowed by a user.

5.3.3 PR.AC-4: Access Permissions and Authorizations are Managed, Incorporating the Principles of Least Privilege and Separation of Duties

This NIST *Cybersecurity Framework* Subcategory is supported by the user authentication and user authorization capabilities. These tools are used to grant access rights to each user and notify if suspicious activity is detected. This includes granting access to maintenance personnel responsible for certain sub-systems or components of ICS environments while preventing them from accessing other sub-systems or components. Suspicious activities include operations attempted by an unauthorized user, restricted operations performed by an authenticated user who is not authorized to perform those operations, and operations that are performed outside of the designated time frame.

5.3.4 PR.AC-7: Users, Devices, and Other Assets are Authenticated (e.g., single-factor, multi-factor) Commensurate with the Risk of the Transaction (e.g., Individual Security and Privacy Risks and Other Organizational Risks)

This NIST *Cybersecurity Framework* Subcategory is supported through user authentication and user authorization capabilities in addition to the native credential management capabilities associated with the tools. Based on the lab’s risk assessment, the authentication and authorization systems used user passwords as one factor to verify identity and grant access to the environment. To bolster security in the environment, IP addresses were used as a secondary factor for remote access.

5.3.5 PR.DS-1: Data-at-Rest is Protected

This NIST *Cybersecurity Framework* Subcategory is supported using file integrity checking. For end points, the file integrity tools alert when changes to local files are detected. For historian backups and system program and configuration backups, data was stored on read only or write-once drives to prevent data manipulation.

5.3.6 PR.DS-6: Integrity Checking Mechanisms are Used to Verify Software, Firmware, and Information Integrity

This NIST *Cybersecurity Framework* Subcategory is supported through file integrity checking tools and the BAD tools. The file integrity checking tools monitor the information on the manufacturing end points for changes. The BAD tools monitor the environments for changes made to software, firmware, and validate sensor and actuator information.

5.3.7 PR.IP-4: Backups of Information are Conducted, Maintained, and Tested

This NIST *Cybersecurity Framework* Subcategory is supported by file integrity checking using secure storage to protect backup data. System configuration settings, PLC logic files, and historian databases all have backups stored on secure storage disks. The secure storage is constructed in a way that prohibits modifying or deleting data that is on the disk.

5.3.8 PR.MA-1: Maintenance and Repair of Organizational Assets are Performed and Logged, with Approved and Controlled Tools

This NIST *Cybersecurity Framework* Subcategory is supported by a combination of tools including AAL, the user authentication and user authorization tools, and the behavior anomaly detection tools. User authentication and user authorization tools provide a controlled environment for authorized users to interact with the manufacturing environment. Behavior anomaly detection tools provide a means to detect maintenance activities in the environment such as PLC logic modification or PLC firmware updates via the network. This information can be combined with data from a computerized maintenance management system to ensure that all maintenance activities are appropriately approved and logged. Also, AAL prevents unapproved software from running on systems to ensure that only approved tools are used for maintenance activities.

5.3.9 PR.MA-2: Remote Maintenance of Organizational Assets is Approved, Logged, and Performed in a Manner that Prevents Unauthorized Access

This NIST *Cybersecurity Framework* Subcategory is supported by the remote access capability integrated with the user authentication and user authorization system. The tools in the solution were used to grant access for performing remote maintenance on specific assets. The tools prevent unauthorized users from gaining access to the manufacturing environment.

5.3.10 DE.AE-1: A Baseline of Network Operations and Expected Data Flows for Users and Systems is Established and Managed

This NIST *Cybersecurity Framework* Subcategory is supported by behavior anomaly detection tools. Network baselines were established and approved based on an understanding of normal operations and data flows identified by the behavior anomaly detection tools.

5.3.11 DE.AE-2: Detected Events are Analyzed to Understand Attack Targets And Methods

This NIST *Cybersecurity Framework* Subcategory is supported by all the capabilities included in the solutions. Logs of suspicious activities from the tools can be used by security managers and engineers to understand what unusual activity has occurred in the manufacturing system. Analyzing these logs provides a mechanism to determine what systems were accessed and what actions may have been performed on them. Although not demonstrated in these solutions, an analytic engine would enhance the detection capability of the solution.

5.3.12 DE.AE-3: Event Data are Collected and Correlated from Multiple Sources and Sensors

This NIST *Cybersecurity Framework* Subcategory is supported by all the capabilities included in the solutions. Each tool detects different aspects of the scenarios from diverse perspectives. Although not demonstrated in these solutions, a data aggregation and correlation tool such as a security information and event management tool would enhance the detection capability of the solution.

5.3.13 DE.CM-1: The Network is Monitored to Detect Potential Cybersecurity Events

This NIST *Cybersecurity Framework* Subcategory is supported by the BAD and remote access capabilities used in the example solutions to monitor the manufacturing network to detect potential cybersecurity events. The BAD tools monitor network communications at the external boundary of the system and at key internal points within the network, along with user activities and traffic patterns, and compare it to the established baseline. The remote access capabilities monitor the network communications at the external boundary of the system. This helps detect unauthorized local, network, and remote connections and identify unauthorized use of the manufacturing system.

5.3.14 DE.CM-3: Personnel Activity is Monitored to Detect Potential Cybersecurity Events

This NIST *Cybersecurity Framework* Subcategory is supported by the authentication and authorization tools that allow for monitoring personnel activity while connected through these tools. Further, AAL and file integrity checking tools provide the ability to monitor user actions on hosts. Additionally, BAD tools monitor and record events associated with personnel actions traversing network traffic. Each tool provides a different perspective in monitoring personnel activity within the environment. The resulting alerts and logs from these tools can be monitored individually or collectively to support investigations for potential malicious or unauthorized activity within the environment.

5.3.15 DE.CM-7: Monitoring for Unauthorized Personnel, Connections, Devices, and Software is Performed

This NIST *Cybersecurity Framework* Subcategory is supported by BAD, AAL, user authentication and user authorization, and remote access capabilities of the solutions. The BAD tools established an information baseline for approved assets and connections. Then the manufacturing network is monitored using the BAD capability for any deviation by the assets and connections from the established baseline. If any deviation is detected, an alert is generated. Additionally, the AAL tool blocks any unauthorized application installation or execution and generates an alert on these events. User authentication and user authorization tools monitor for unauthorized personnel connecting to the environment. Remote access capabilities monitor for unauthorized connections to the environment.

6 Future Build Considerations

This guide has presented technical solutions for maintaining and monitoring system and information integrity, which will help detect and prevent incidents in a manufacturing environment. Future builds should demonstrate methods and techniques for fusing event and log data from multiple platforms into a security operations center to improve monitoring and detection capabilities for an organization. Future builds should also demonstrate how to recover from a loss of system or information integrity such as a ransomware attack for ICS environments.

Additionally, trends in manufacturing such as Industry 4.0 and the industrial IoT are increasing connectivity, increasing the attack surface, and increasing the potential for vulnerabilities. Future builds should consider how these advances can be securely integrated into manufacturing environments.

Appendix A List of Acronyms

AAL	Application Allowlisting
BAD	Behavioral Anomaly Detection
CRS	Collaborative Robotic System
CRADA	Cooperative Research and Development Agreement
CSF	NIST *Cybersecurity Framework*
CSMS	Cybersecurity for Smart Manufacturing Systems
DMZ	Demilitarized Zone
EL	Engineering Laboratory
FOIA	Freedom of Information Act
ICS	Industrial Control System
IoT	Internet of Things
IT	Information Technology
LAN	Local Area Network
NCCoE	National Cybersecurity Center of Excellence
NFS	Network File Share
NIST	National Institute of Standards and Technology
NISTIR	NIST Interagency or Internal Report
OT	Operational Technology
PCS	Process Control System
PLC	Programmable Logic Controller
SCADA	Supervisory Control and Data Acquisition
SMB	Server Message Block
SP	Special Publication
SPAN	Switched Port Analyzer

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

SRP	Software Restriction Policies
SSH	Secure Shell
TE	Tennessee-Eastman
VDI	Virtual Desktop Interface
VLAN	Virtual Local Area Network
VPN	Virtual Private Network

Appendix B Glossary

Access Control	The process of granting or denying specific requests to: 1) obtain and use information and related information processing services; and 2) enter specific physical facilities (e.g., federal buildings, military establishments, border crossing entrances). SOURCE: Federal Information Processing Standard (FIPS) 201; CNSSI-4009
Architecture	A highly structured specification of an acceptable approach within a framework for solving a specific problem. An architecture contains descriptions of all the components of a selected, acceptable solution while allowing certain details of specific components to be variable to satisfy related constraints (e.g., costs, local environment, user acceptability). SOURCE: FIPS 201-2
Authentication	Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in an information system. SOURCE: FIPS 200
Authorization	The right or a permission that is granted to a system entity to access a system resource. SOURCE: NIST SP 800-82 Rev. 2
Backup	A copy of files and programs made to facilitate recovery if necessary. SOURCE: NIST SP 800-34 Rev. 1
Continuous Monitoring	Maintaining ongoing awareness to support organizational risk decisions. SOURCE: NIST SP 800-137
CRADA	Collaborative Research and Development Agreement SOURCE: NIST SP 1800-5b, NIST SP 1800-5c
Cybersecurity	Prevention of damage to, protection of, and restoration of computers, electronic communications systems, electronic communications services, wire communication, and electronic communication, including information contained therein, to ensure its availability, integrity, authentication, confidentiality, and nonrepudiation. SOURCE: CNSSI 4009-2015 (NSPD-54/HSPD-23)
Cyber Attack	An attack, via cyberspace, targeting an enterprise's use of cyberspace for the purpose of disrupting, disabling, destroying, or maliciously controlling a computing environment/infrastructure; or destroying the integrity of the data or stealing controlled information. SOURCE: NIST SP 800-30 Rev. 1

Data	A subset of information in an electronic format that allows it to be retrieved or transmitted. SOURCE: CNSSI-4009
Data Integrity	The property that data has not been changed, destroyed, or lost in an unauthorized or accidental manner. SOURCE: CNSSI-4009
File Integrity Checking	Software that generates, stores, and compares message digests for files to detect changes made to the files. SOURCE: NIST SP 800-115
Firmware	Computer programs and data stored in hardware – typically in read-only memory (ROM) or programmable read-only memory (PROM) – such that the programs and data cannot be dynamically written or modified during execution of the programs. SOURCE: CNSSI 4009-2015
Industrial Control Systems	An information system used to control industrial processes such as manufacturing, product handling, production, and distribution. SOURCE: NIST SP 800-30 Rev. 1
Information Security	The protection of information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability. SOURCE: FIPS 199 (44 U.S.C., Sec. 3542)
Information System	A discrete set of information resources organized for the collection, processing, maintenance, use, sharing, dissemination, or disposition of information. SOURCE: FIPS 200 (44 U.S.C., Sec. 3502)
Information Technology	Any equipment or interconnected system or subsystem of equipment that is used in the automatic acquisition, storage, manipulation, management, movement, control, display, switching, interchange, transmission, or reception of data or information by the executive agency. SOURCE: FIPS 200
Log	A record of the events occurring within an organization's systems and networks. SOURCE: NIST SP 800-92
Malware	A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system. SOURCE: NIST SP 800-111

Network Traffic	Computer network communications that are carried over wired or wireless networks between hosts. SOURCE: NIST SP 800-86
Operational Technology	Programmable systems or devices that interact with the physical environment (or manage devices that interact with the physical environment). SOURCE: NIST SP 800-37 Rev. 2
Privacy	Assurance that the confidentiality of, and access to, certain information about an entity is protected. SOURCE: NIST SP 800-130
Remote Access	Access to an organizational information system by a user (or an information system) communicating through an external, non-organization-controlled network (e.g., the Internet). SOURCE: NIST SP 800-128 under Remote Access from NIST SP 800-53
Risk	The level of impact on organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals resulting from the operation of an information system given the potential impact of a threat and the likelihood of that threat occurring. SOURCE: FIPS 200
Risk Assessment	The process of identifying the risks to system security and determining the probability of occurrence, the resulting impact, and additional safeguards that would mitigate this impact. Part of Risk Management and synonymous with Risk Analysis. SOURCE: NIST SP 800-63-2
Risk Management Framework	The Risk Management Framework (RMF), presented in NIST SP 800-37, provides a disciplined and structured process that integrates information security and risk management activities into the system development life cycle. SOURCE: NIST SP 800-82 Rev. 2 (NIST SP 800-37)
Security Control	A protection measure for a system SOURCE: NIST SP 800-123
Virtual Machine	Software that allows a single host to run one or more guest operating systems SOURCE: NIST SP 800-115

Appendix C References

[1] C. Singleton et al., X-Force Threat Intelligence Index 2021, IBM, February 2021, https://www.ibm.com/security/data-breach/threat-intelligence

[2] A Sedgewick et al., *Guide to Application Whitelisting*, NIST SP 800-167, NIST, Oct. 2015. Available: http://dx.doi.org/10.6028/NIST.SP.800-167.

[3] Department of Homeland Security, Critical Manufacturing Sector Cybersecurity Framework Implementation Guidance, 2015. Available: https://www.cisa.gov/uscert/sites/default/files/c3vp/framework_guidance/critical-manufacturing-framework-implementation-guide-2015-508.pdf.

[4] Executive Order no. 13636, *Improving Critical Infrastructure Cybersecurity, DCPD201300091*, Feb. 12, 2013. Available: https://obamawhitehouse.archives.gov/the-press-office/2013/02/12/executive-order-improving-critical-infrastructure-cybersecurity.

[5] NIST, *Framework for Improving Critical Infrastructure Cybersecurity*, V1.1 April 16, 2018. Available: https://doi.org/10.6028/NIST.CSWP.04162018.

[6] J. McCarthy et al., Securing Manufacturing Industrial Control Systems: Behavioral Anomaly Detection, NIST Interagency Report (NISTIR) 8219, NIST, Nov. 2018. Available: https://www.nccoe.nist.gov/sites/default/files/library/mf-ics-nistir-8219.pdf.

[7] K. Stouffer et al., Cybersecurity Framework Manufacturing Profile, NIST Internal Report 8183, NIST, May 2017. Available: https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8183.pdf.

[8] R. Candell et al., An Industrial Control System Cybersecurity Performance Testbed, NISTIR 8089, NIST, Nov. 2015. Available: http://nvlpubs.nist.gov/nistpubs/ir/2015/NIST.IR.8089.pdf.

[9] Security and Privacy Controls for Federal Information Systems and Organizations, NIST SP 800-53 Revision 5, NIST, Apr. 2013. Available: https://doi.org/10.6028/NIST.SP.800-53r5.

[10] W. Newhouse et al., National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework, NIST SP 800-181, Aug. 2017. Available: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-181.pdf.

[11] J. Cawthra et al*., Data Integrity: Identifying and Protecting Assets Against Ransomware and Other Destructive Events*, NIST Special Publication 1800-25 Dec. 2020, https://doi.org/10.6028/NIST.SP.1800-25.

[12] Celia Paulsen, Robert Byers*,* Glossary of Key Information Security Terms NISTIR 7298, https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.7298r3.pdf.

[13] U.S.-Canada Power Systems Outage Task Force, Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations. Available: https://www.energy.gov/sites/default/files/oeprod/DocumentsandMedia/Outage_Task_Force_-_DRAFT_Report_on_Implementation.pdf

[14] K. Stouffer et al., *Guide to Industrial Control Systems (ICS) Security*, NIST SP 800-82 Revision 2, NIST, June 2015, Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r2.pdf

[15] J. J. Downs and E. F. Vogel, "A Plant-wide Industrial Problem Process," Comput. Chem. Eng., vol. 17, no. 3, 1993, pp. 245–255

Appendix D Scenario Execution Results

The following section provides details regarding the execution and results from each scenario. Details such as usernames, filenames, IP addresses, etc. are specific to the NCCoE lab environment and are provided for reference only.

D.1 Executing Scenario 1: Protect Host from Malware via USB

An authorized user inserts a USB storage device containing a malware file (1.exe) into a system in the manufacturing environment (e.g., an engineering workstation). After insertion, the malware file (1.exe) attempts to execute. The expected outcome is that the application allowlisting technology blocks the execution of the file.

D.1.1 Build 1

D.1.1.1 Configuration

- Application Allowlisting: Carbon Black
    - Agent installed on an HMI Workstation and configured to communicate to the Carbon Black Server.

D.1.1.2 Test Results

Carbon Black successfully detects and blocks the malware (1.exe) from running as shown in Figure D-1. Figure D-2 shows Carbon Black's server log. The log provides more detail on the activity detected by Carbon Black.

Figure D-1 An Alert from Carbon Black Showing that Malware (1.exe) was Blocked from Executing

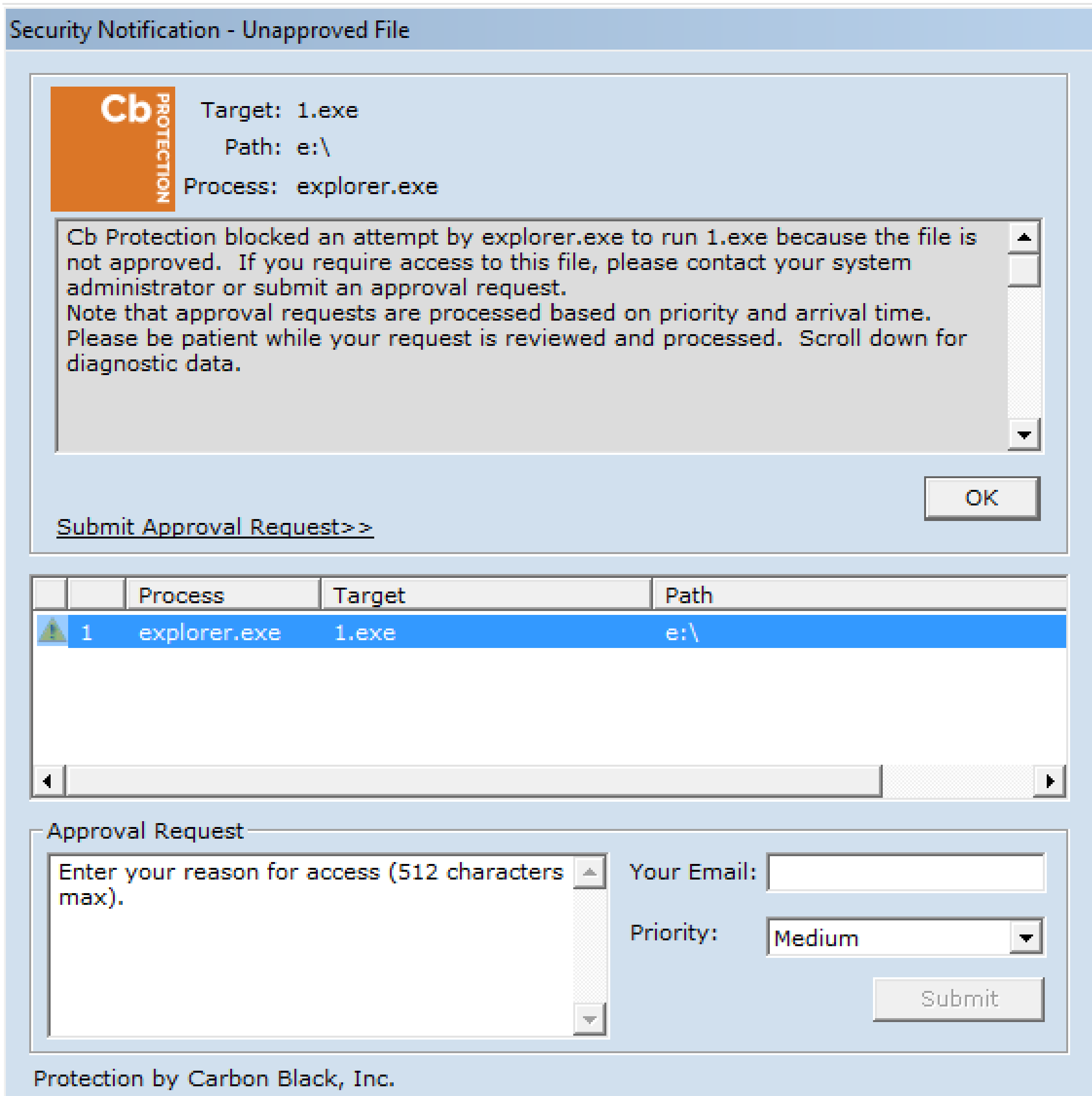


Figure D-2: Carbon Black's Server Provides Additional Details and Logs of the Event

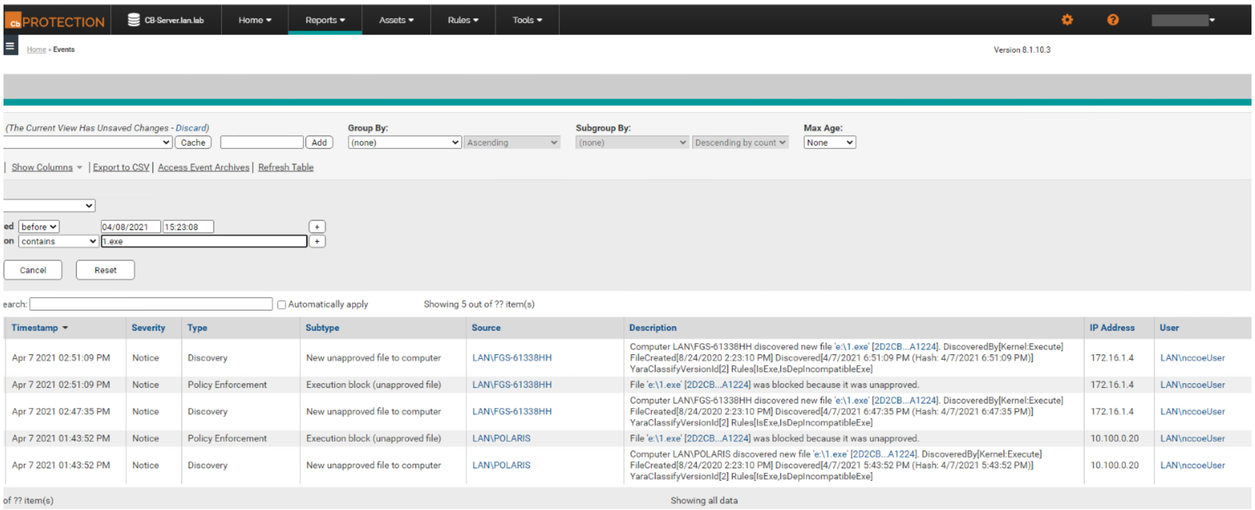
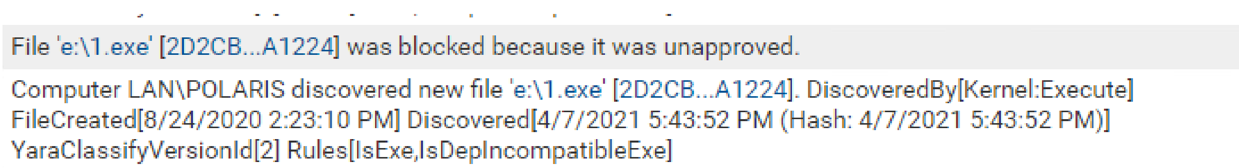


Figure D-3 Carbon Black's Server Log of the Event



D.1.2 Build 2

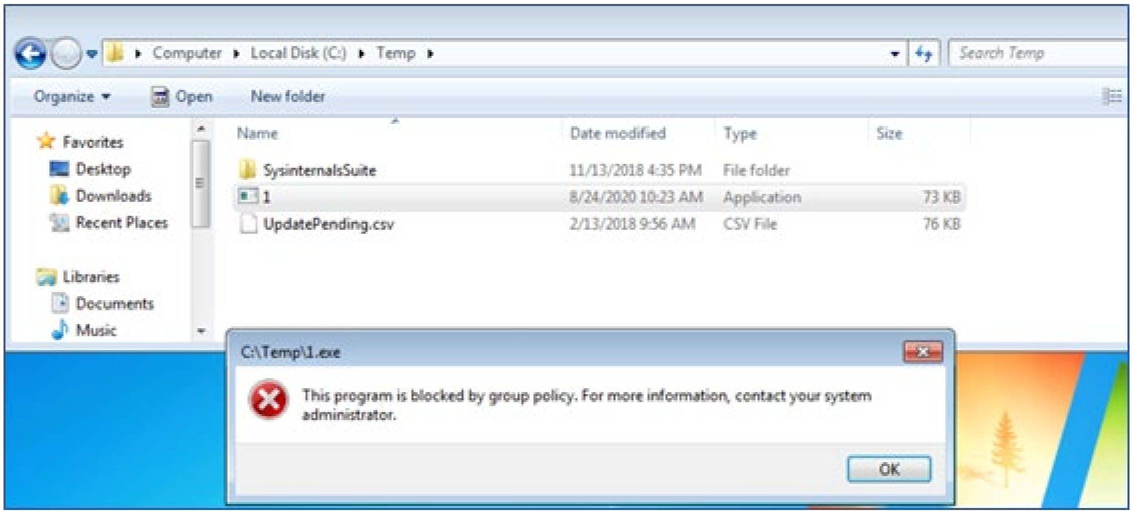
D.1.2.1 Configuration

- Application Allowlisting: Windows SRP
    - Allowlisting policies are applied to HMI Workstation.

D.1.2.2 Test Results

The execution of *1.exe* is blocked successfully when Windows SRP is enforced as shown in Figure D-4.

Figure D-4 Windows 7 Alert as a Result of Windows SRP Blocking the Execution of 1.exe



D.1.3 Build 3

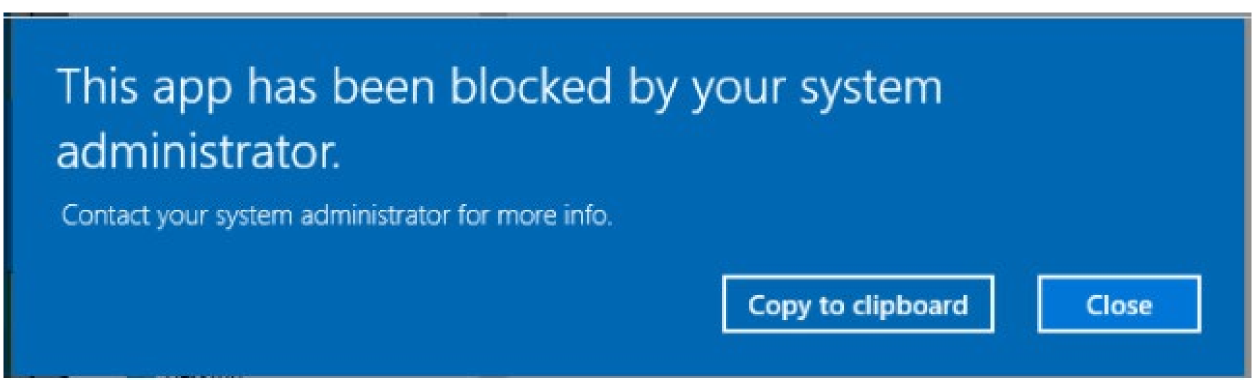
D.1.3.1 Configuration

- Application Allowlisting: Windows SRP
    - Allowlisting policies are applied to Engineering Workstation.

D.1.3.2 Test Results

For Build 3, Windows SRP application allowlisting is enabled in the Collaborative Robotics environment. Figure D-5 shows that the executable is blocked on the CRS workstation.

Figure D-5 Windows 10 Alert as a Result of Windows SRP Blocking the Execution of 1.exe



D.1.4 Build 4

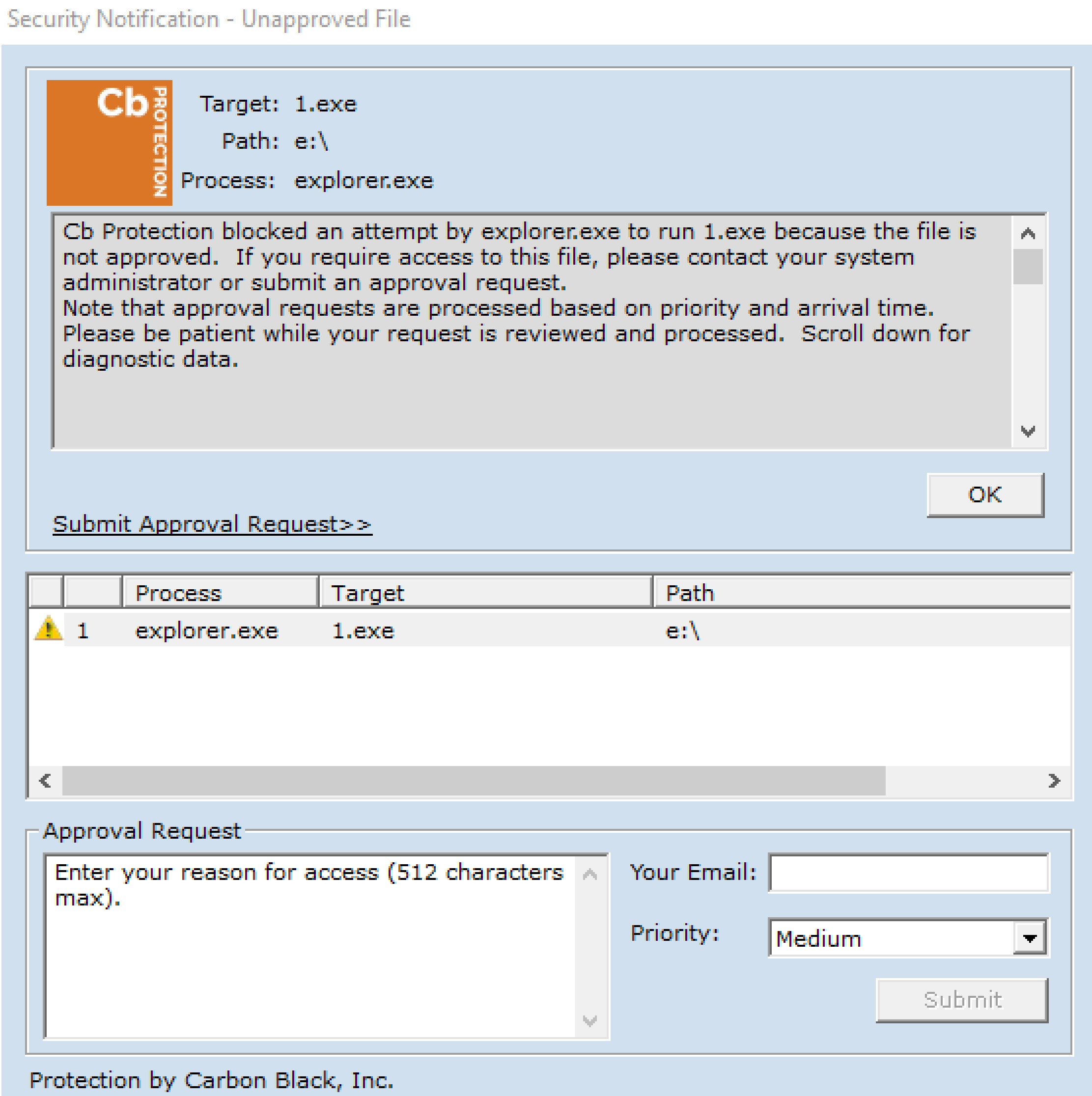
D.1.4.1 Configuration

- Application Allowlisting : Carbon Black
    - Agent installed on Engineering Workstation and configured to communicate to the Carbon Black Server.

D.1.4.2 Test Results

Carbon Black successfully detects and blocks the malicious file as shown by the Carbon Black notification in Figure D-6.

Figure D-6 Carbon Black Blocks the Execution of 1.exe for Build 4



D.2 Executing Scenario 2: Protect Host from Malware via Network Vector

An attacker who has already gained access to the corporate network attempts to pivot into the ICS environment through the DMZ. From a system in the DMZ, the attacker scans for vulnerable systems in the Testbed LAN environment to continue pivoting toward the ICS environments. In an attempt to establish a persistent connection into the ICS environment, the malicious file (1.exe) is copied to a system in the Testbed LAN environment and executed. The expected outcome is that the malicious file is

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

blocked by the application allowlisting tool, and the RDP and scanning network activity is observed by the behavioral anomaly detection tool.

D.2.1 Build 1

D.2.1.1 Configuration

- Application Allowlisting: Carbon Black
  - Agent installed on systems in the DMZ, Testbed LAN, and PCS VLAN 1 and 2 and configured to communicate to the Carbon Black Server.
- Behavior Anomaly Detection: Tenable.ot
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.2.1.2 Test Results

Abnormal network traffic is detected by Tenable.ot as shown in Figure D-7. Figure D-8 shows the initial RDP connection between an external system and the DMZ system, and Figure D-9 provides more detail of the session activity. Figure D-10 shows that Tenable.ot detected the VNC connection between the DMZ and the Testbed LAN. Figure D-11 shows a detected ports scan performed by the DMZ system target at a system in the Testbed LAN. Tenable.ot detected the RDP scan from the DMZ to the NESSUS VM in the Testbed LAN, as shown in Figure D-12, and Figure D-13 provides more details on that detected event. The execution of the malware (1.exe) is blocked by Carbon Black agent as shown in Figure D-14.

Figure D-7 Tenable.ot Dashboard Showing the Events that were Detected

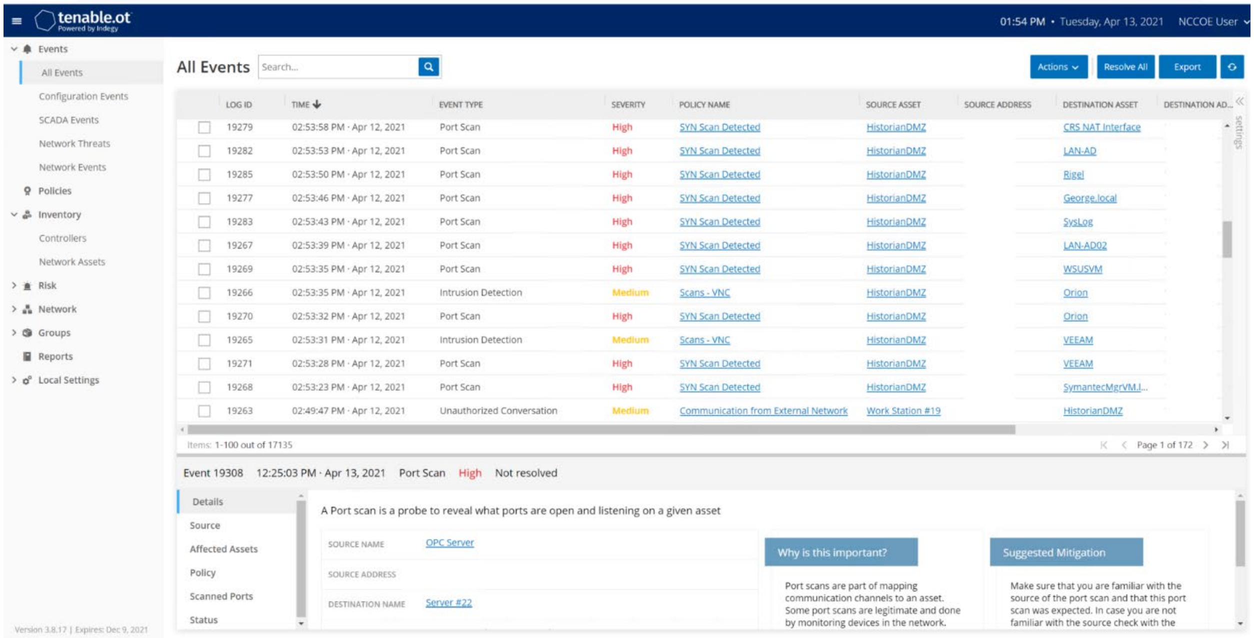


Figure D-8 Detected RDP Session Activity from External System to DMZ System

Figure D-9 Event Detection Detail for the RDP Connection from the External System to the Historian in the DMZ

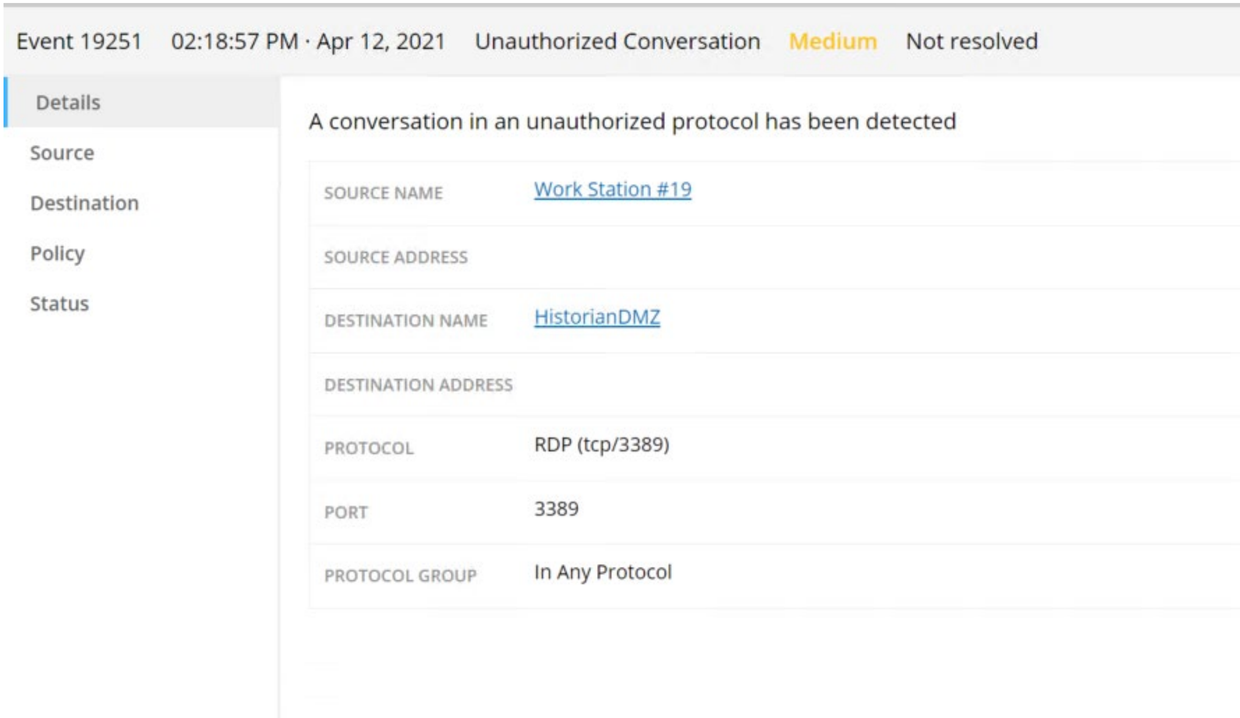


Figure D-10 Tenable.ot Detected VNC Connection Between the DMZ and the Testbed LAN

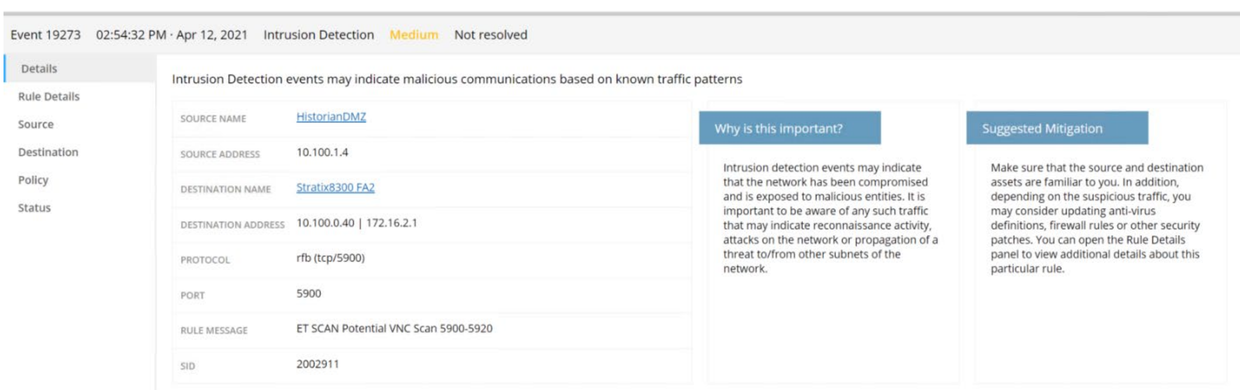


Figure D-11 Tenable.ot Event Detail for a Detected Port Scan from a DMZ System Targeting a System in the Testbed LAN

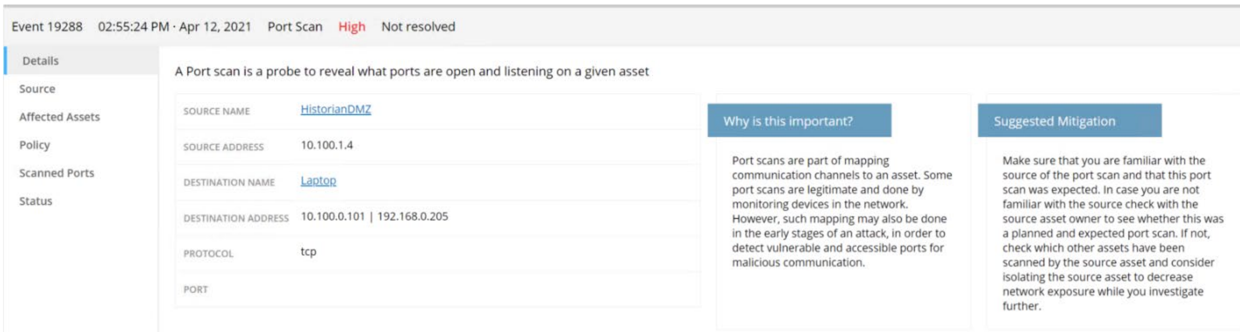


Figure D-12 Detected RDP from a DMZ system to a Testbed LAN system

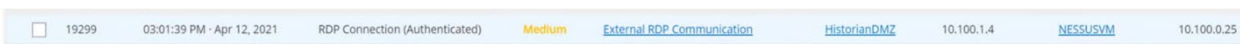


Figure D-13 Tenable.ot Event Detail Showing the RDP Connection Between the Historian in the DMZ to a Workstation in the Testbed LAN

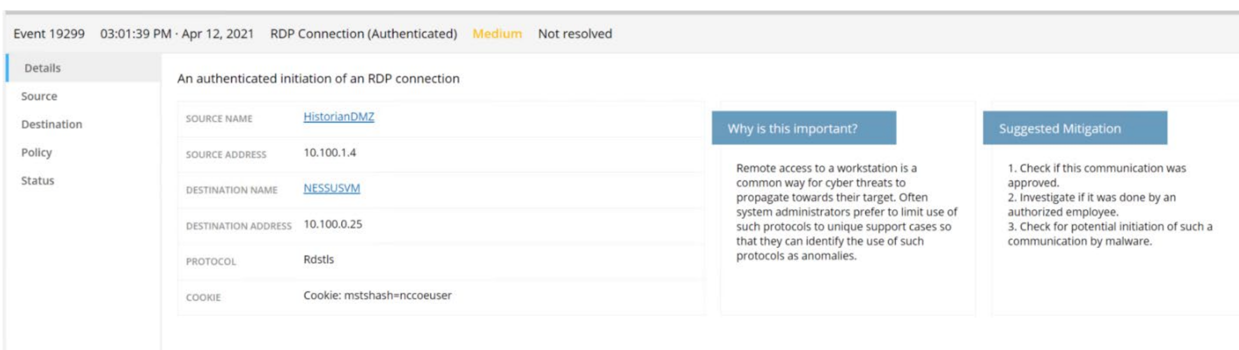
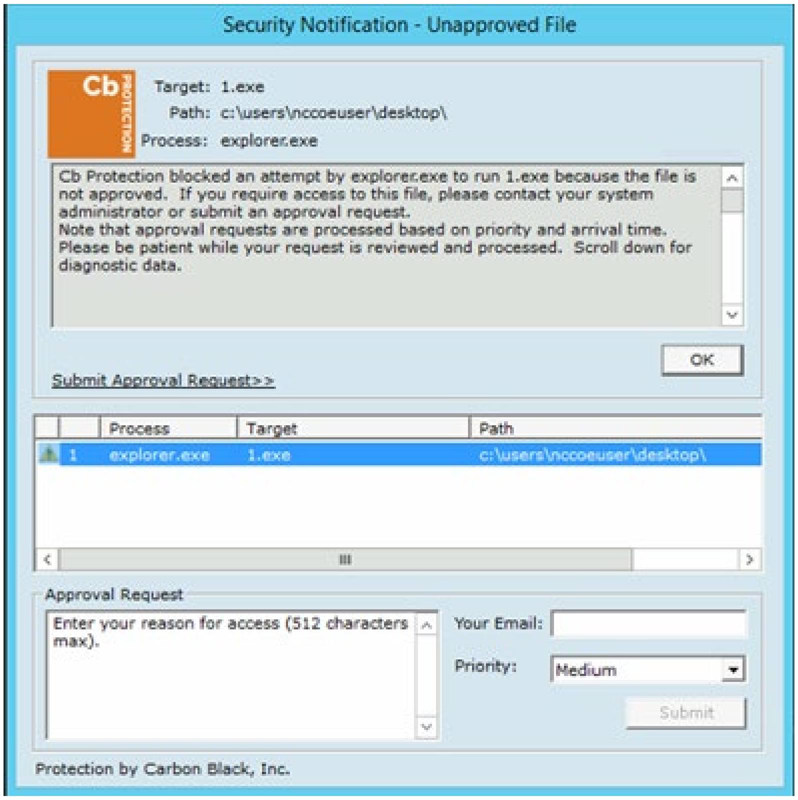


Figure D-14 Attempt to Execute 1.exe Failed



D.2.2 Build 2

D.2.2.1 Configuration

- Application Allowlisting: Windows SRP
    - Allowlisting policies are applied to systems in the DMZ, Testbed LAN, and PCS VLAN 1 and 2.
- Behavior Anomaly Detection: eyeInspect
    - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.2.2.2 Test Results

Figure D-15 shows the RDP alert for connection into the DMZ, while Figure D-16 shows the details of the alert. Figure D-17 shows a collection of suspicious activity detected by Forescout eyeInspect when scanning and an RDP connection is executed. Figure D-18 and Figure D-19 show details of a port scanning alert and the second RDP connection into the manufacturing environment, respectively. The attempt to execute malware (1.exe) is blocked by Windows SRP as shown in Figure D-20.

Figure D-15 Alert Dashboard Showing Detection of an RDP Session

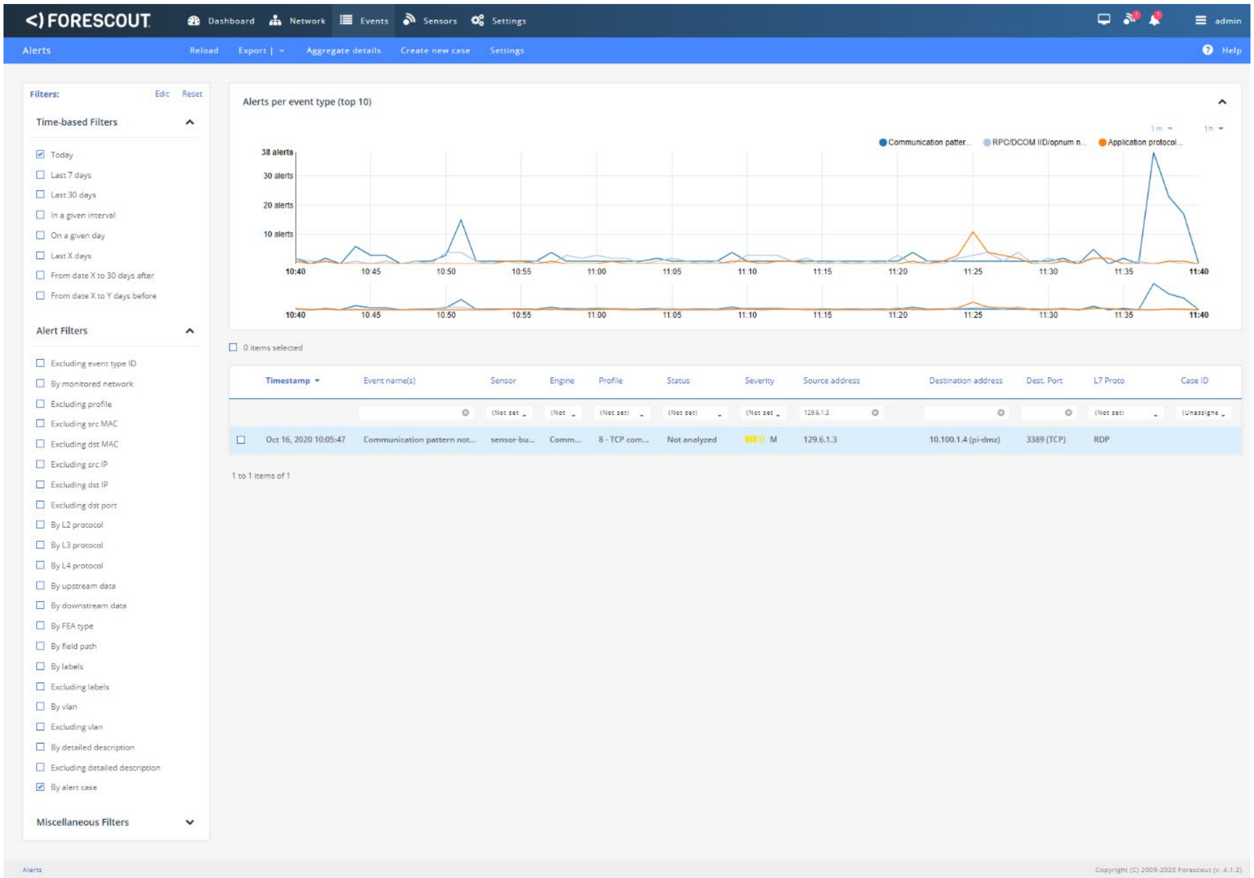


Figure D-16 Details of the Detected RDP Session Activity from an External System to DMZ System

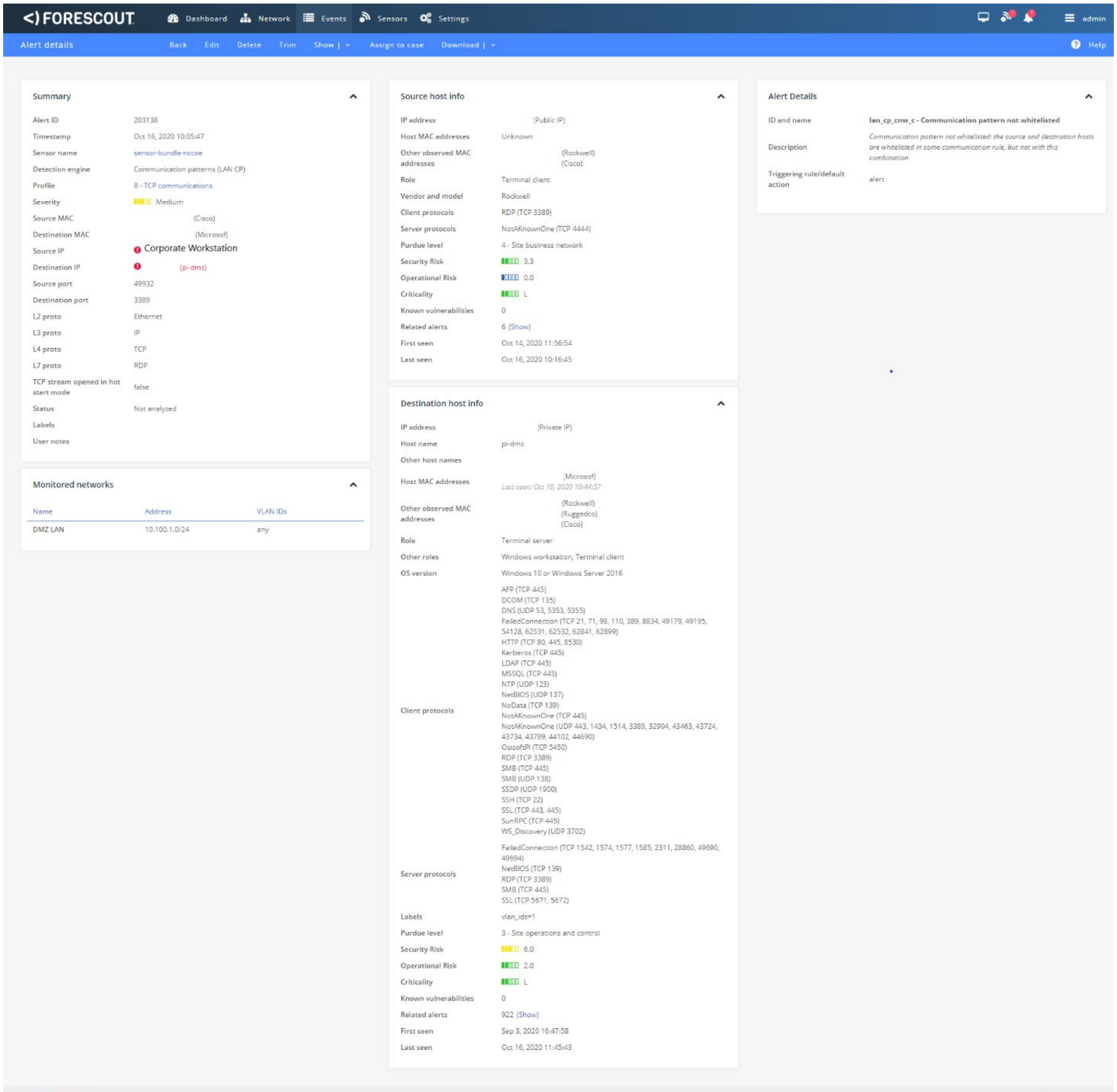


Figure D-17 Detection of Scanning Traffic and RDP Connection into Manufacturing Environment

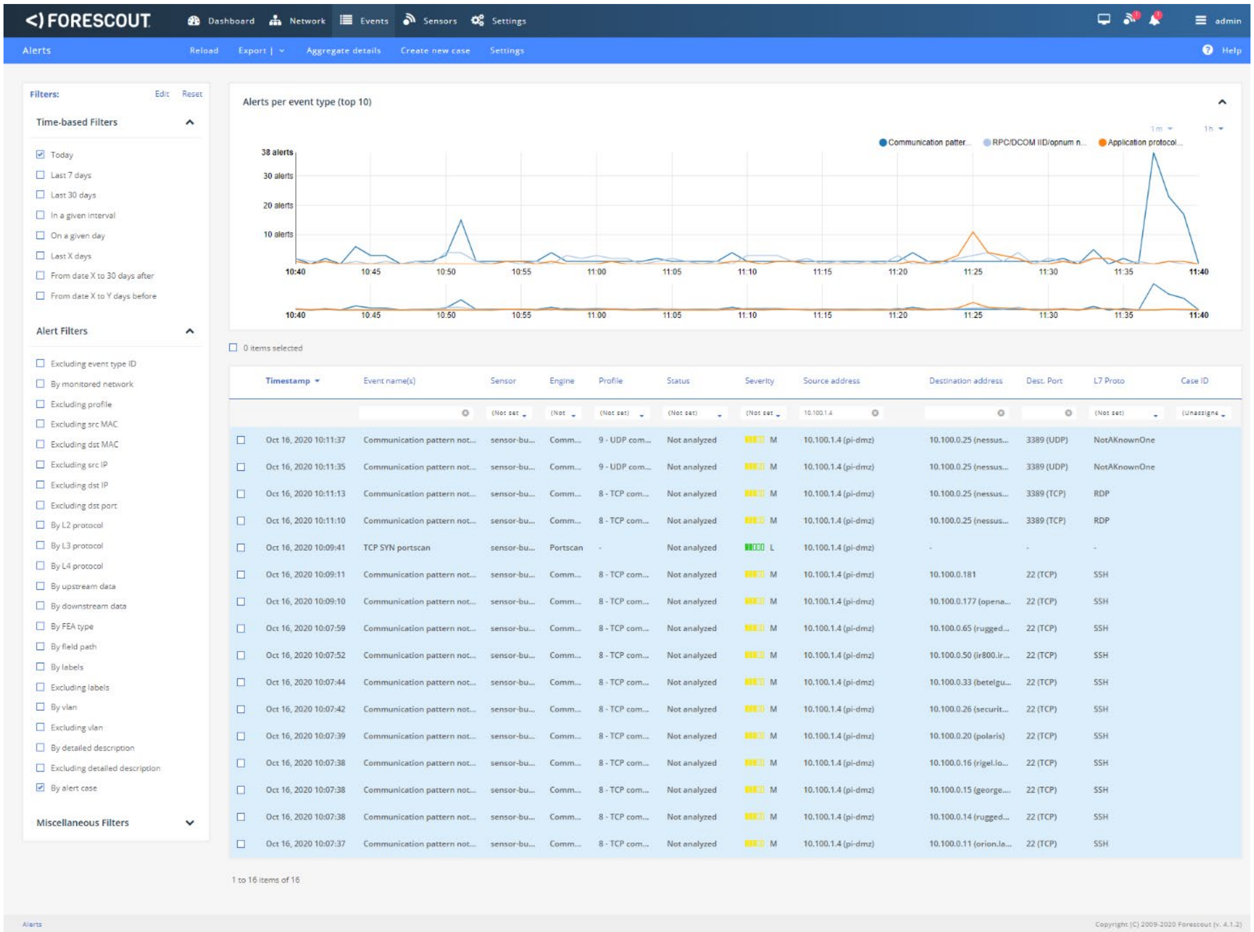


Figure D-18 Details of One of the Port Scan Alerts

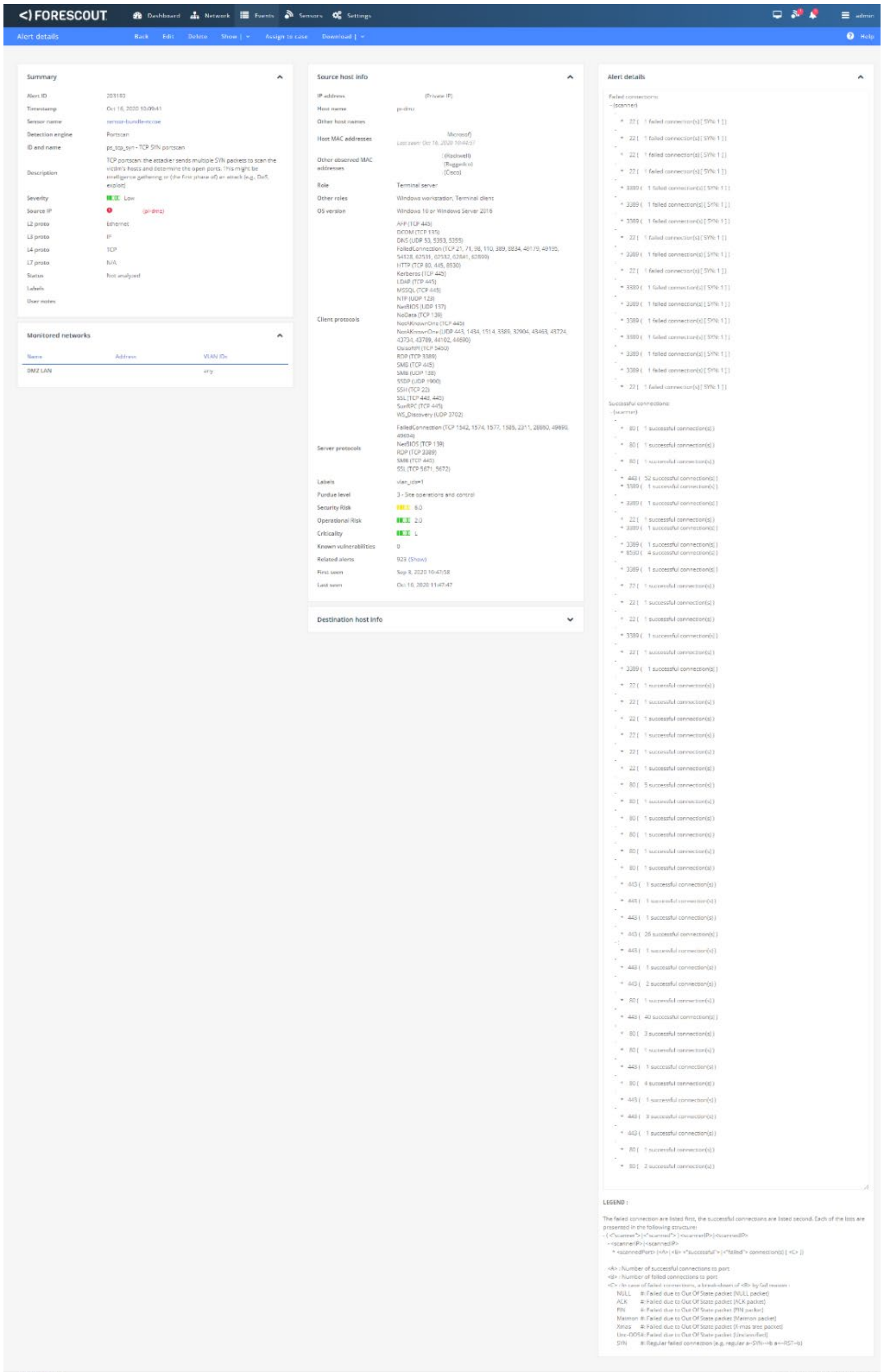


Figure D-19 Details of Alert for RDP Connection into Manufacturing Environment

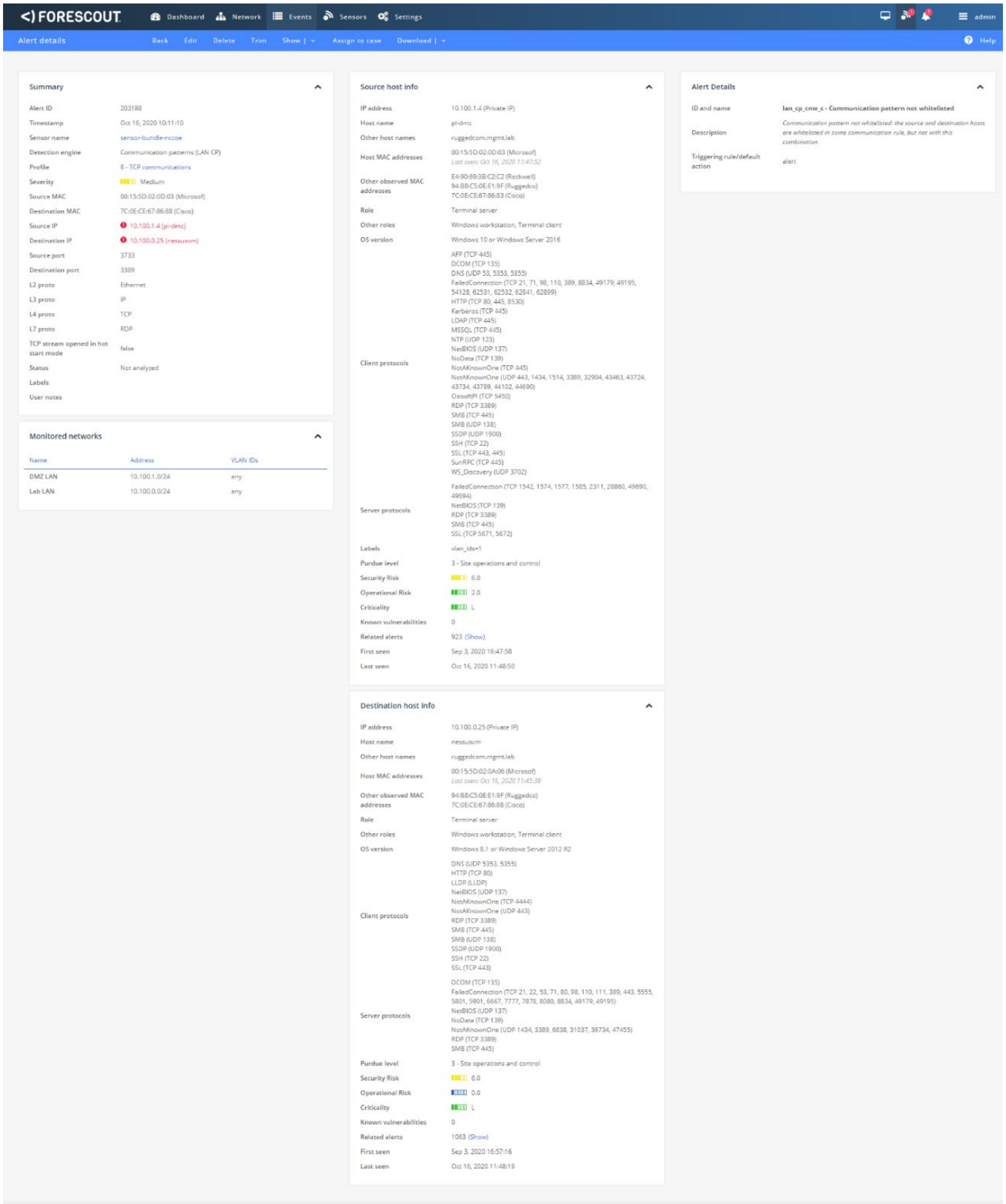
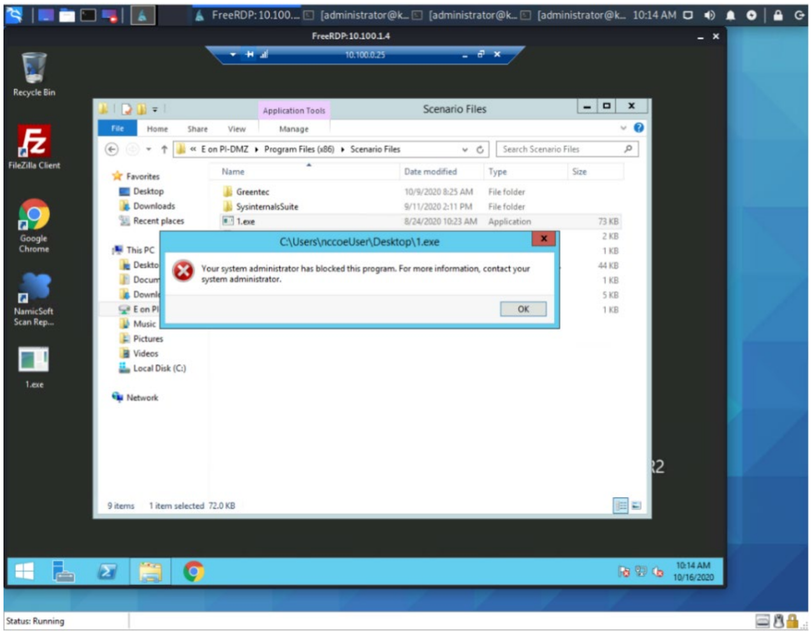


Figure D-20 Dialog Message Showing 1.exe was Blocked from Executing



D.2.3 Build 3

D.2.3.1 Configuration

- Application Allowlisting: Windows SRP
    - Allowlisting policies are applied to systems in the DMZ, Testbed LAN, and Supervisory LAN
- Behavior Anomaly Detection: Dragos
    - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.2.3.2 Test Results

Windows SRP blocks the attempted execution of 1.exe (Figure D-21). Figure D-22 shows the alerts generated by Dragos when it detected the remote connection to the target. Figure D-23 depicts the detected RDP session from an external system to the DMZ system. Figure D-24 depicts network scanning alert details. Figure D-25 depicts the RDP session from a DMZ system to the Testbed LAN system.

Figure D-21 Windows SRP blocked 1.exe From Executing

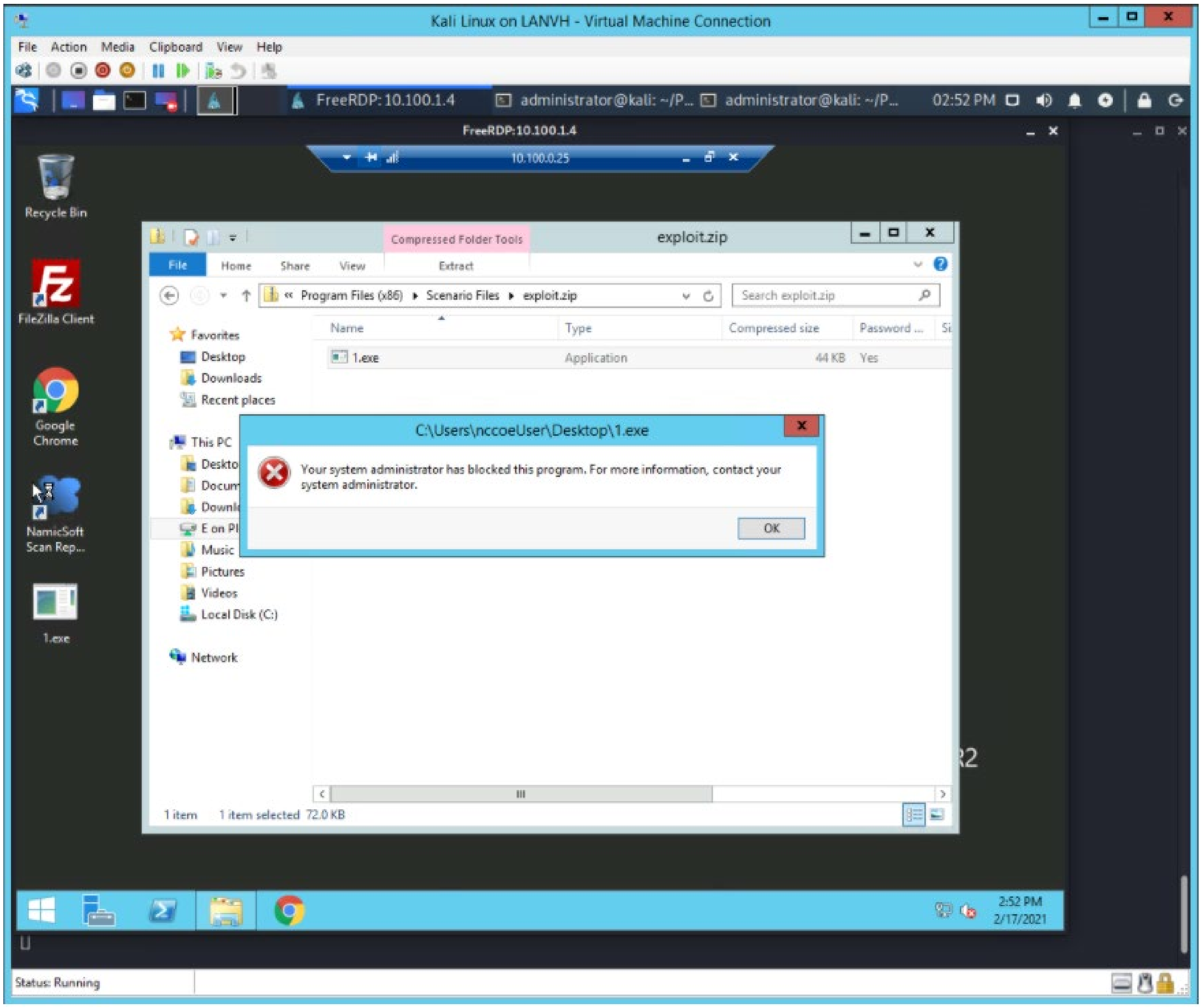


Figure D-22 Log of Alerts Detected by Dragos

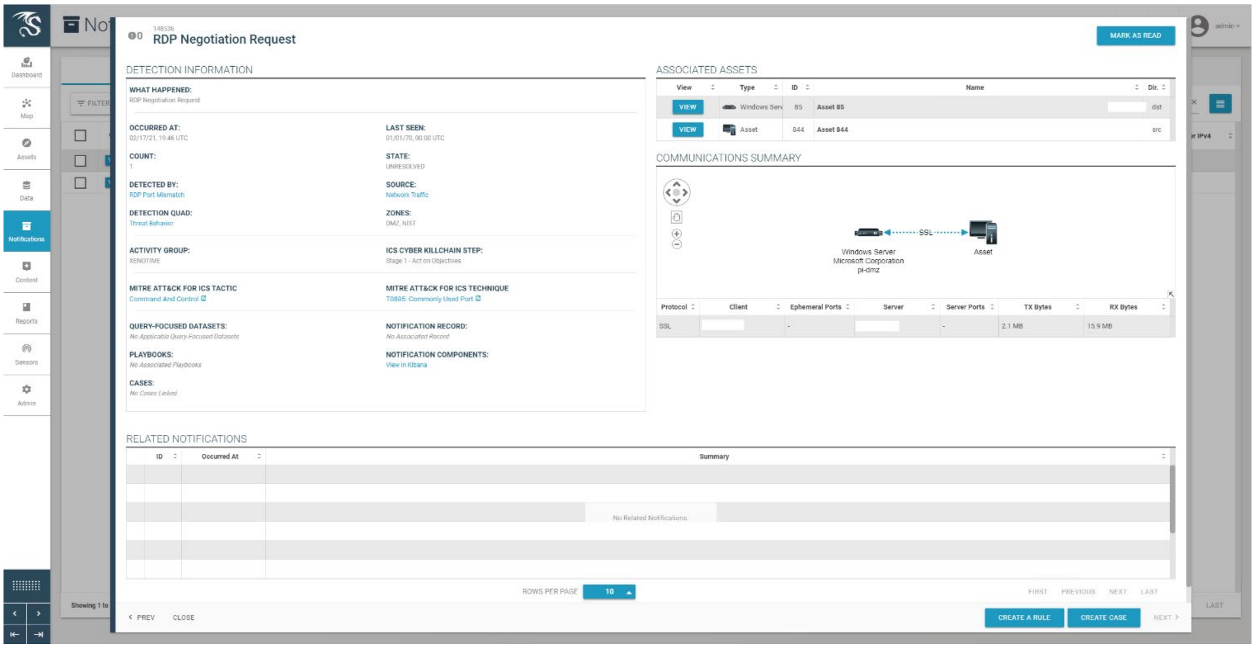


Figure D-23 Detail of RDP Session Activity Between an External System and a DMZ System

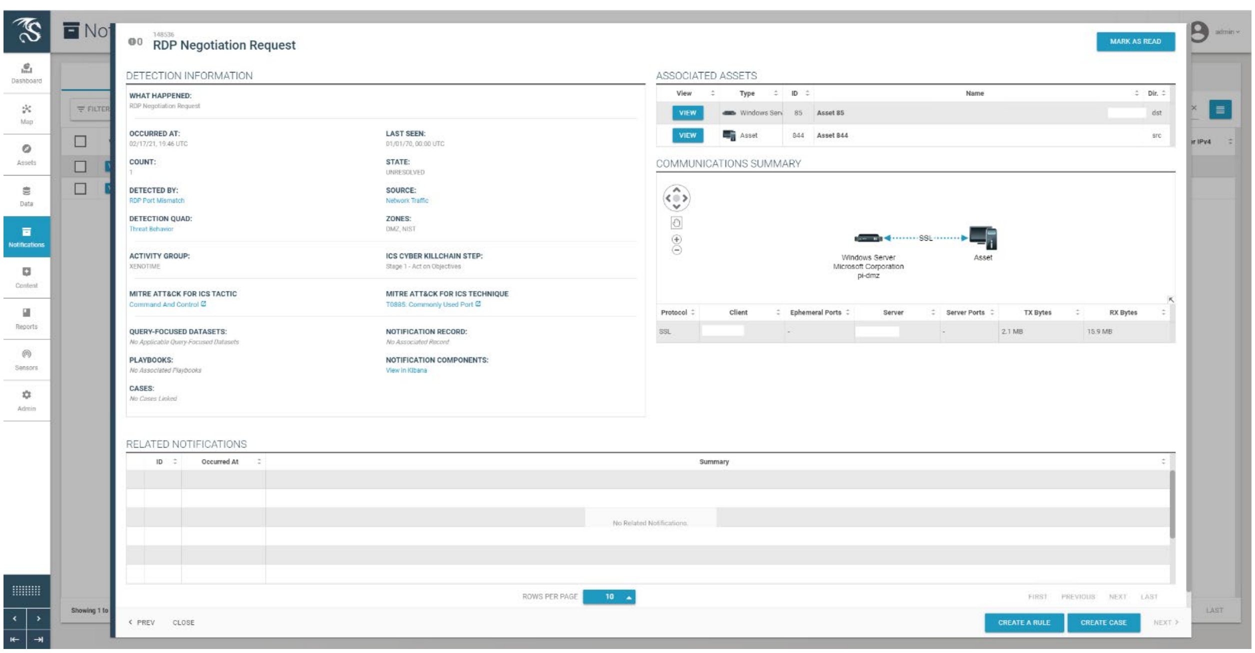


Figure D-24 Detail for Network Scanning Alert

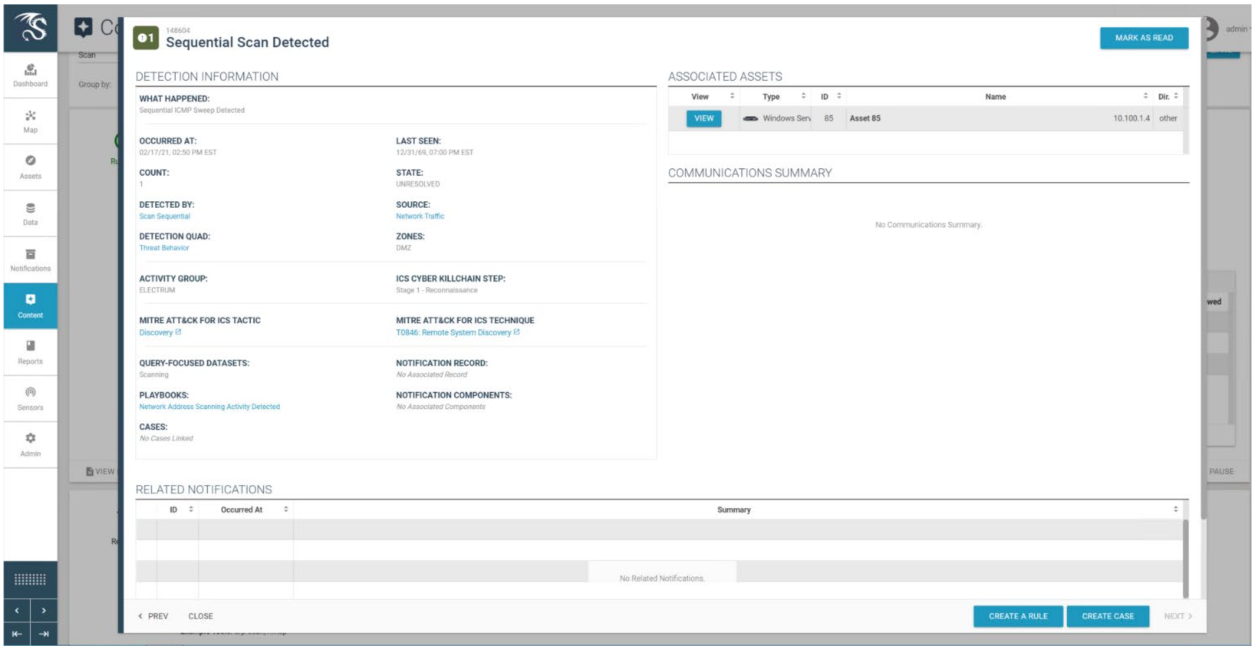
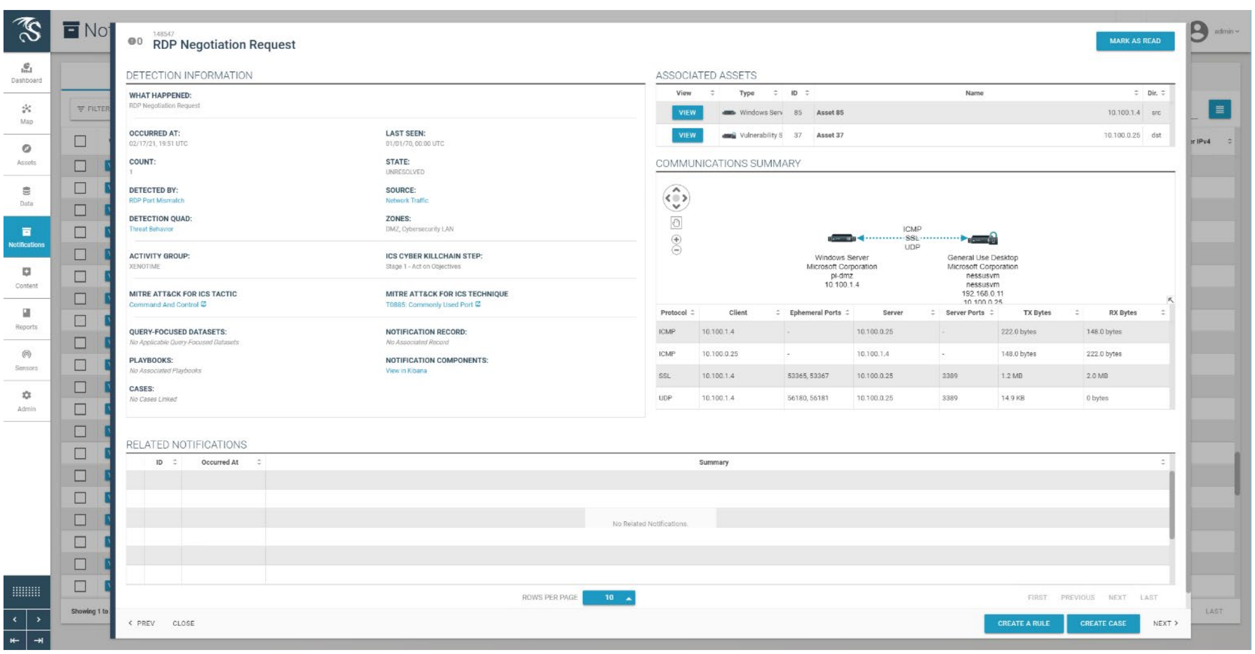


Figure D-25 Detail of RDP Session Activity Between a DMZ System and a Testbed LAN System



D.2.4 Build 4

D.2.4.1 Configuration

- Application Allowlisting: Carbon Black
  - Agent installed on systems in the DMZ, Testbed LAN, and Supervisory LAN and configured to communicate to the Carbon Black Server.
- Behavior Anomaly Detection: Azure Defender for IoT
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.2.4.2 Test Results

Azure Defender for IoT is able to detect the remote access connection to the DMZ as seen in Figure D-26. Figure D-27 shows detection of scanning activity, while Figure D-28 shows details of the scan. The RDP connection into the manufacturing environment is seen in Figure D-29. Carbon Black blocks 1.exe from executing as shown in Figure D-30.

Figure D-26 Azure Defender for IoT "info" Event Identified Remote Access Connection to the DMZ

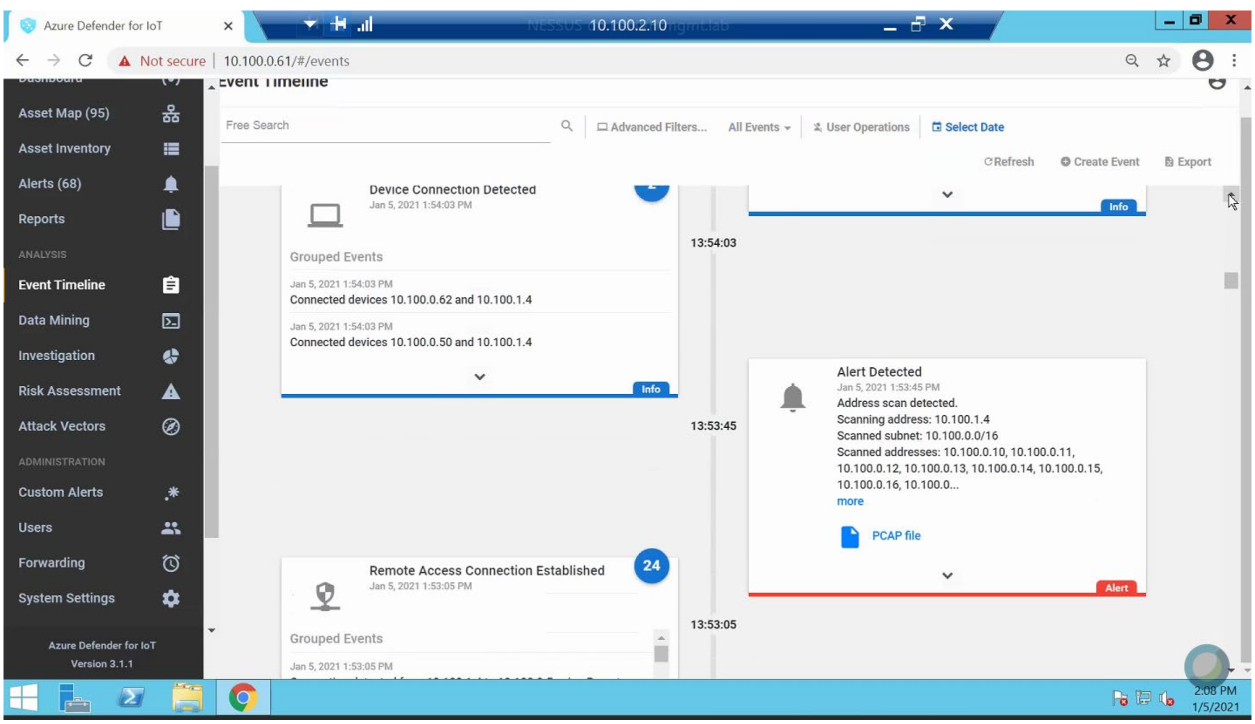


Figure D-27 Alert for Scanning Activity

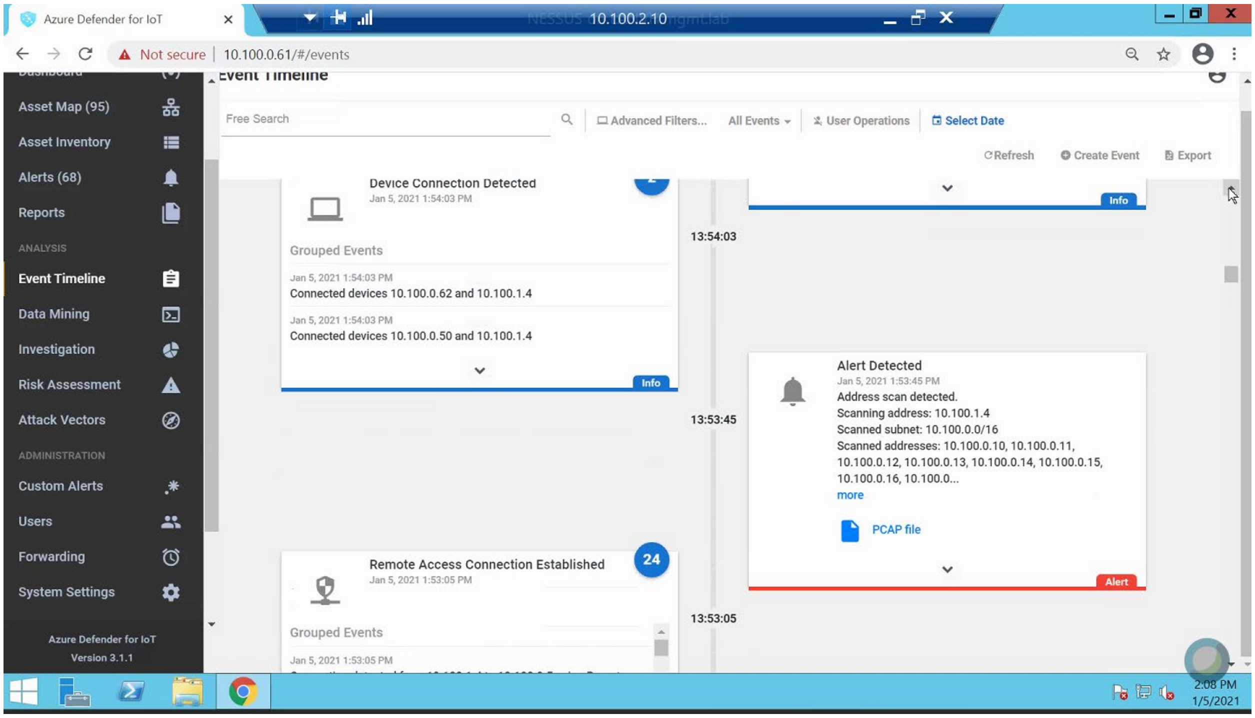


Figure D-28 Details for the Scanning Alert

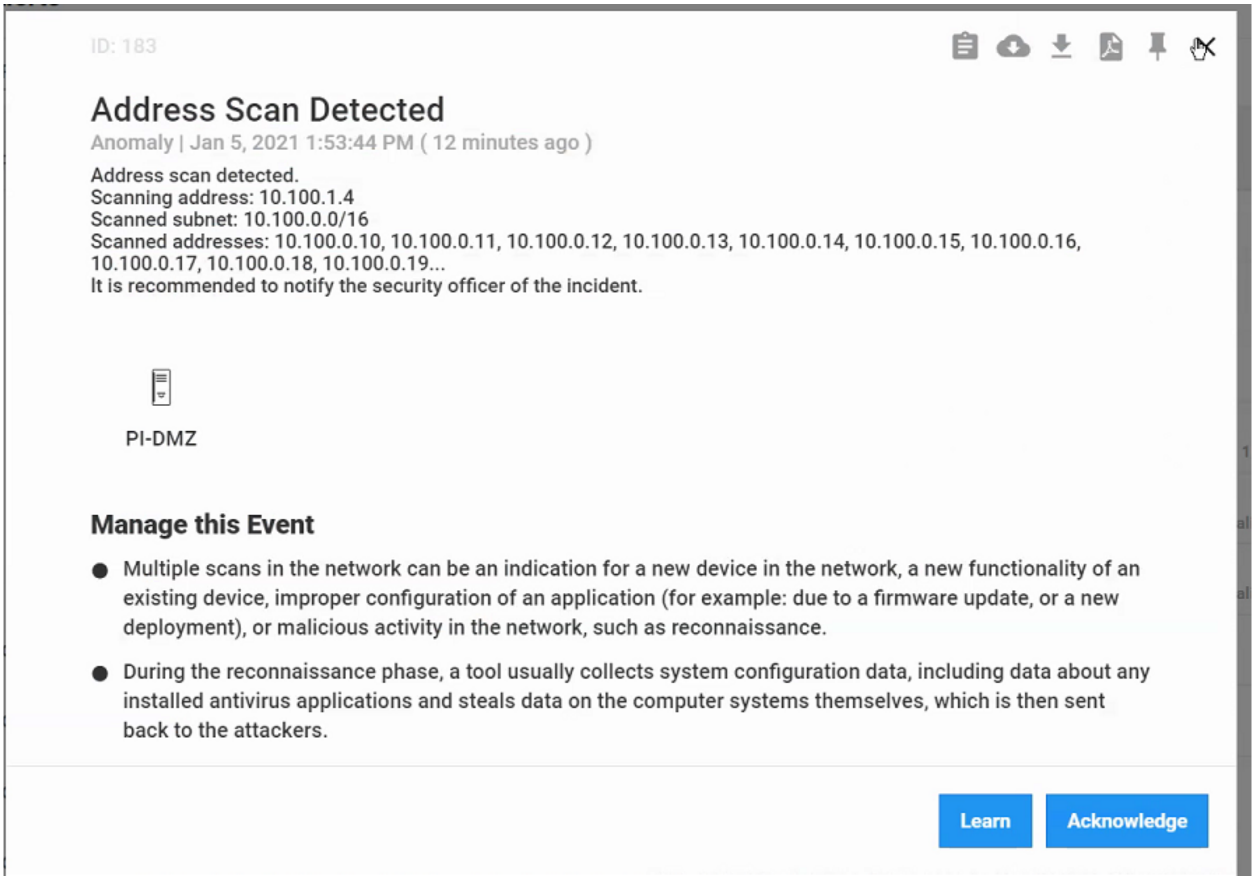


Figure D-29 Detection of RDP Connection into the Manufacturing Environment

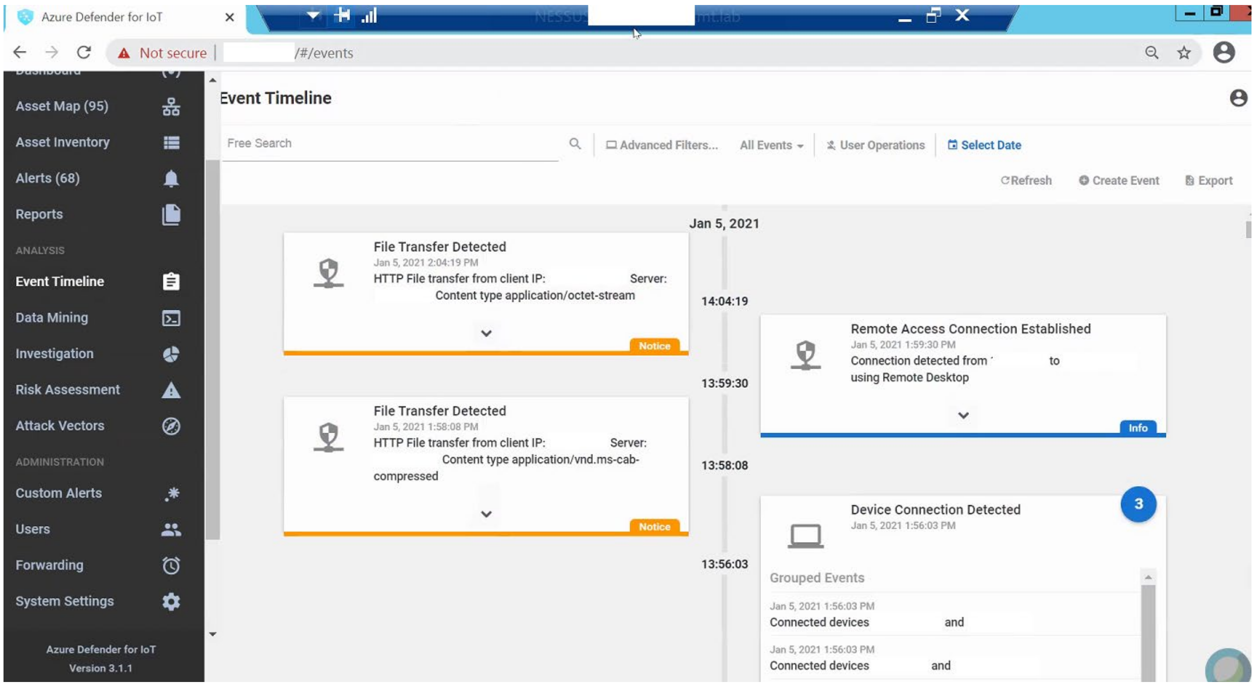
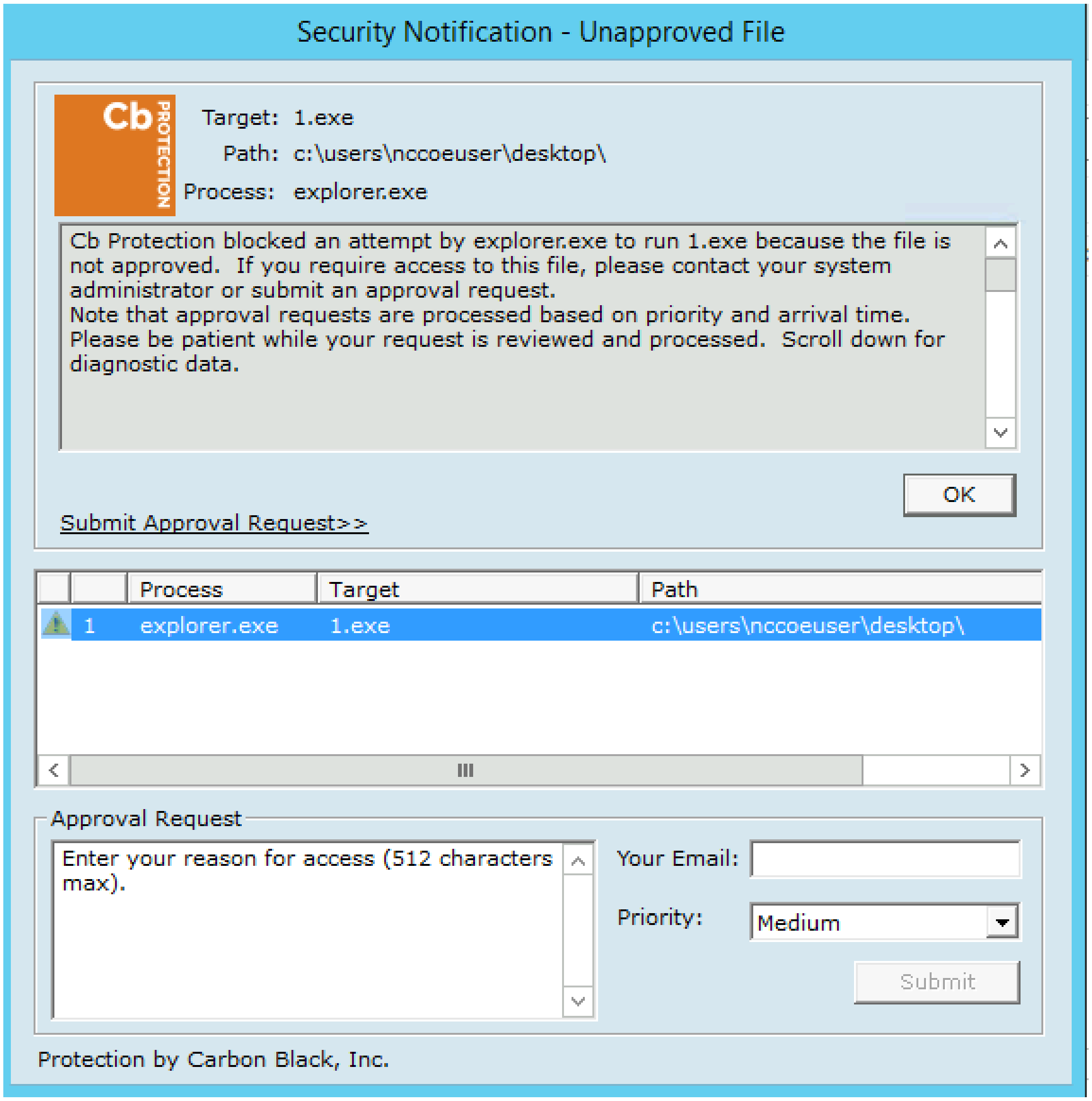


Figure D-30 Carbon Black Shows an Alert for Blocking File 1.exe



D.3 Executing Scenario 3: Protect Host from Malware via Remote Access Connections

An authorized user with an authorized remote workstation, infected with a worm-type malware, connects via remote access capabilities to the manufacturing environments. The malware on the remote host attempts to scan the manufacturing environment to identify vulnerable hosts. The expected result is that the remote access tools effectively stop the worm-type malicious code from propagating to the manufacturing environment from the infected remote workstation.

D.3.1 Build 1

D.3.1.1 Configuration

- Remote Access: Cisco VPN
  - Configured to allow authorized VPN users to access to ConsoleWorks web interface.
- User Authentication/User Authorization: ConsoleWorks
  - Configured for access PCS environment.

D.3.1.2 Test Results

Figure D-31 shows the remote connection being established through the Cisco AnyConnect VPN application through which a browser is used to access the ConsoleWorks web interface (Figure D-32). Once a connection to ConsoleWorks was established, the simulated worm attack was executed on the remote PC to scan the target network. The scan was successfully blocked by the VPN configuration.

Figure D-31 Secured VPN Connection to Environment with Cisco AnyConnect

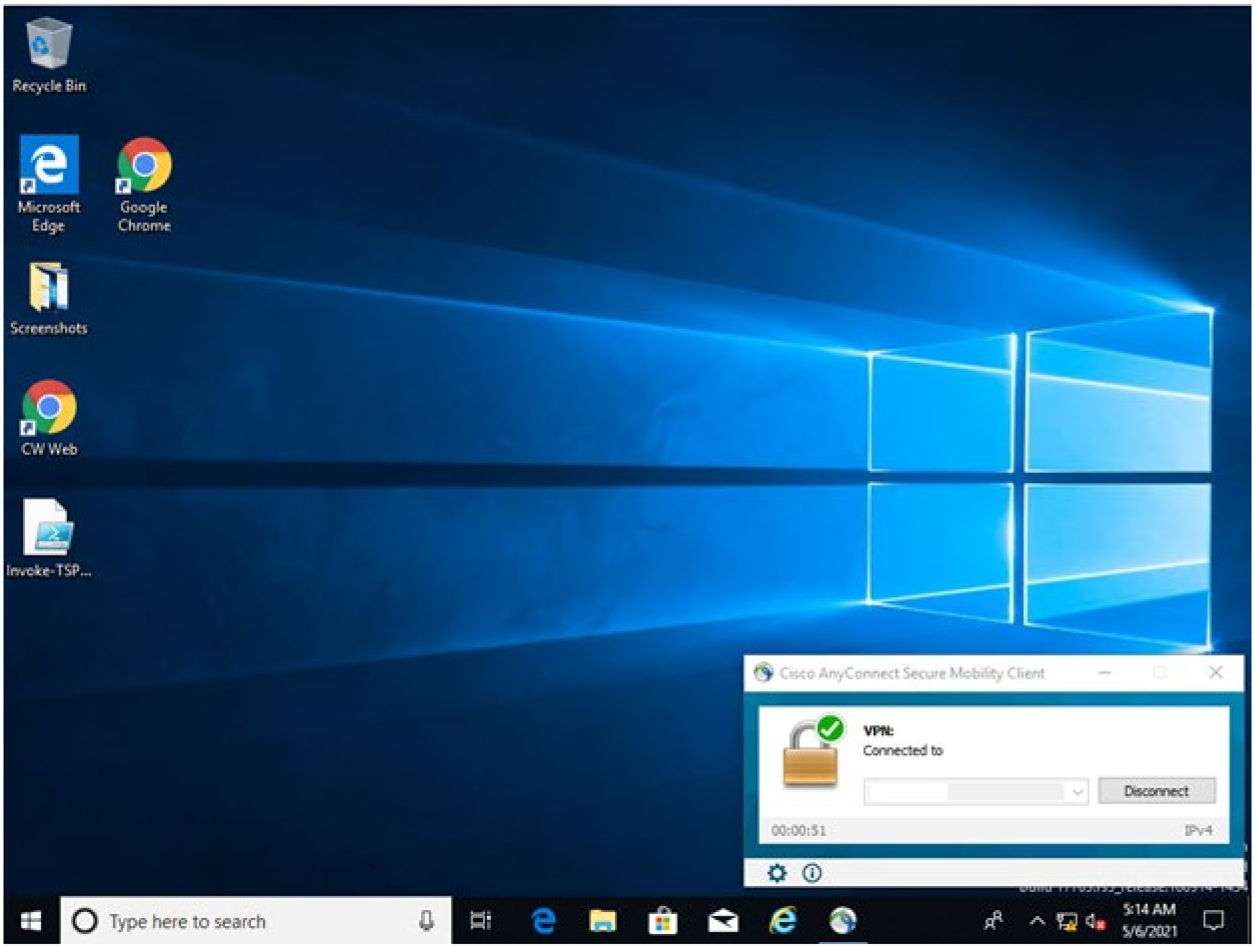
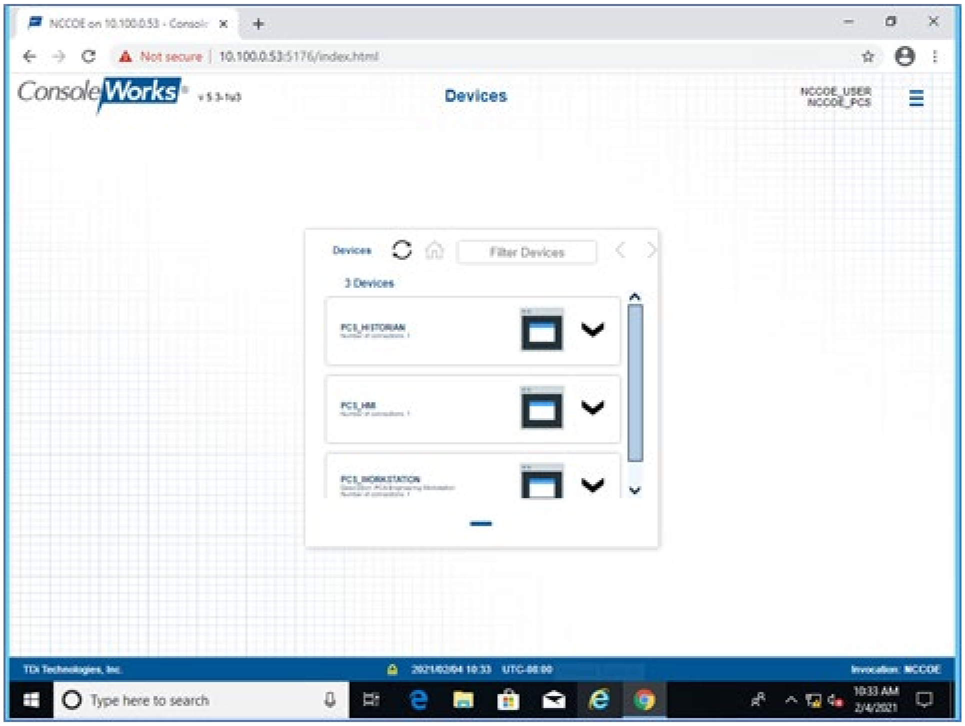


Figure D-32 Remote Access is Being Established Through ConsoleWorks



D.3.2 Build 2

D.3.2.1 Configuration

- Remote Access, User Authentication/User Authorization: Dispel
  - Dispel VDI is configured to allow authorized users to access PCS environment through the Dispel Enclave to the Dispel Wicket.

D.3.2.2 Test Results

The user connects to the Dispel VDI as shown in Figure D-33 and then connects to the PCS workstation as shown in Figure D-34. Once a connection to the NCCOE environment was established, the simulated worm attack was executed on the remote PC to scan the target network. The scan was successfully blocked by the Dispel VDI configuration.

Figure D-33 Dispel VDI with Interface for Connecting Through Dispel Enclave to Dispel Wicket ESI

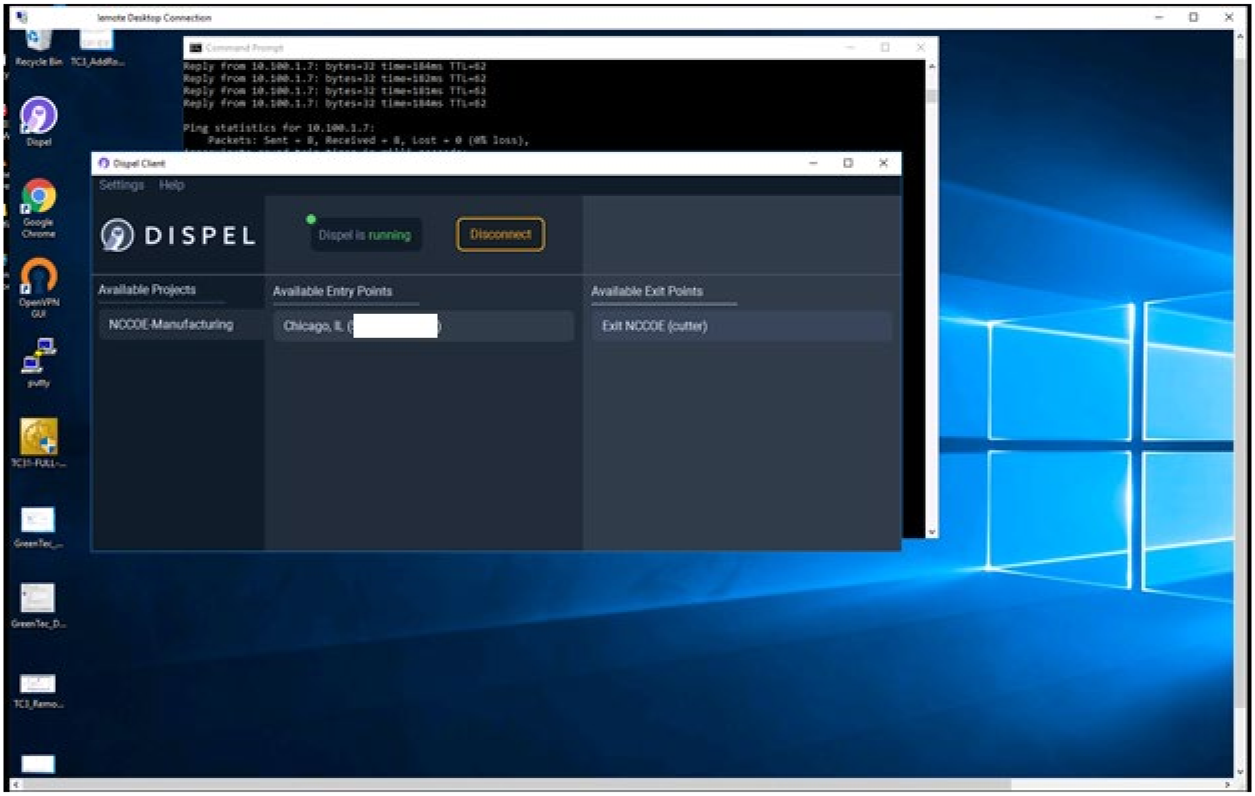


Figure D-34 Nested RDP Session Showing Dispel Connection into the PCS Workstation

D.3.3 Build 3

D.3.3.1 Configuration

- Remote Access: Cisco VPN
    - Configured to allow authorized VPN users to access to ConsoleWorks web interface.
- User Authentication/User Authorization: ConsoleWorks
    - Configured for access CRS environment.

D.3.3.2 Test Results

Figure D-35 shows the remote connection being established through the Cisco AnyConnect VPN application, where a browser is used to access the ConsoleWorks web interface (Figure D-36). Once a connection to ConsoleWorks was established, the simulated worm attack was executed on the remote PC to scan the target network. The scan was successfully blocked by the VPN configuration.

Figure D-35 VPN Connection to Manufacturing Environment

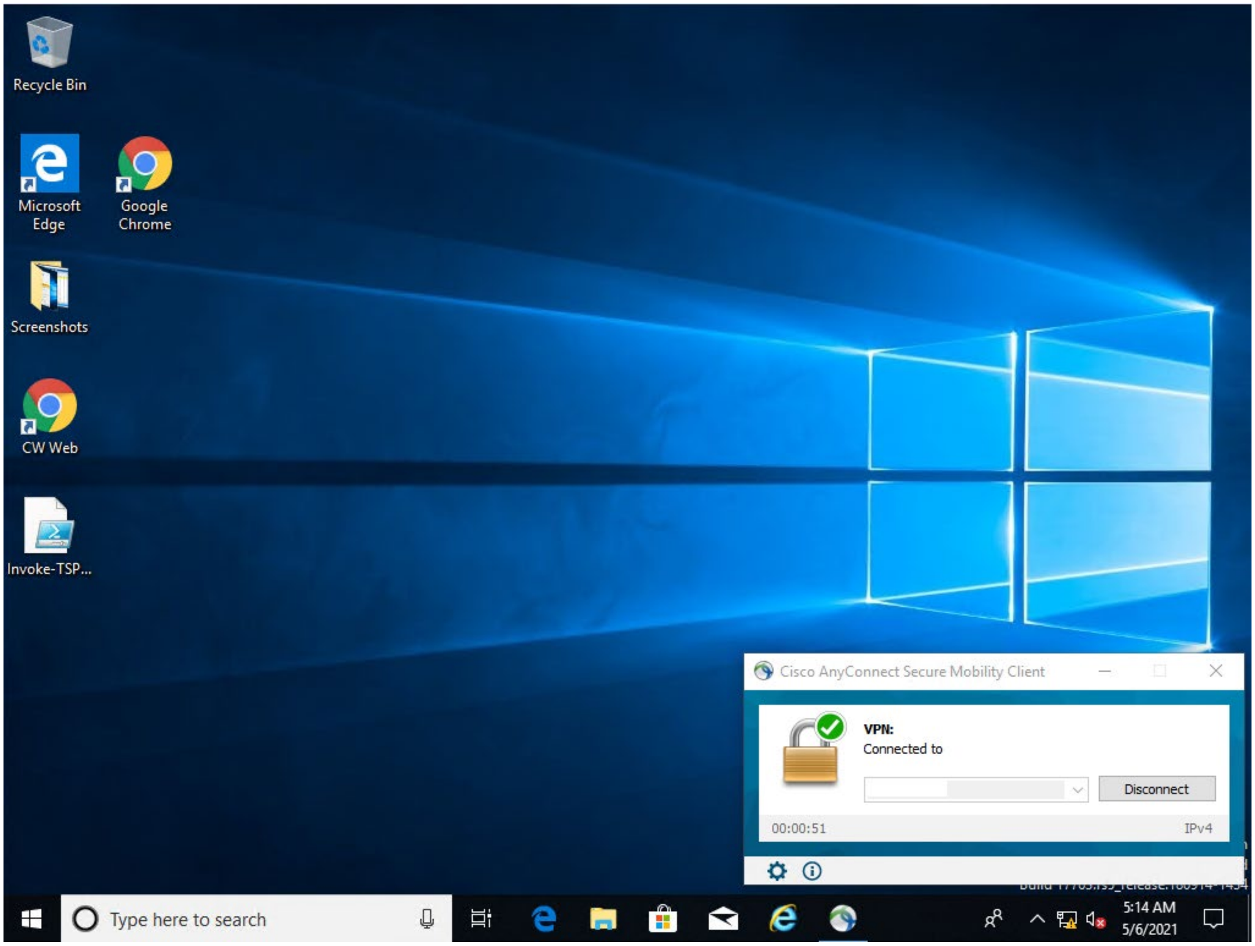
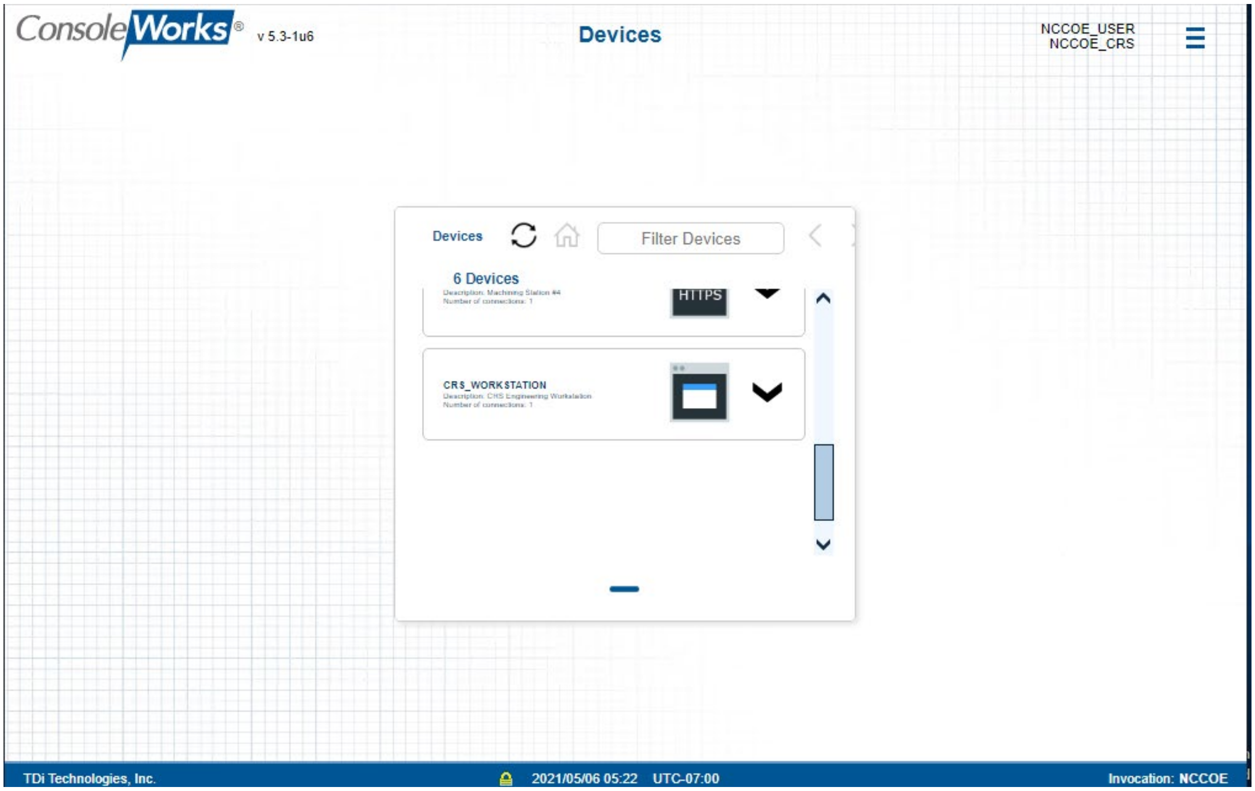


Figure D-36 Remote Access is Being Established Through ConsoleWorks



D.3.4 Build 4

D.3.4.1 Configuration

- Remote Access, User Authentication/User Authorization: Dispel
    - Dispel VDI is configured to allow authorized users to access the PCS environment through the Dispel Enclave to the Dispel Wicket.

D.3.4.2 Test Results

Figure D-37 shows the Dispel VDI desktop, which allows a connection to the CRS workstation in Figure D-38. Once a connection to the NCCOE environment was established, the simulated worm attack was executed on the remote PC to scan the target network. The scan was successfully blocked by the use of the Dispel VDI.

Figure D-37 Dispel VDI Showing Interface for Connecting Through Dispel Enclave to Dispel Wicket

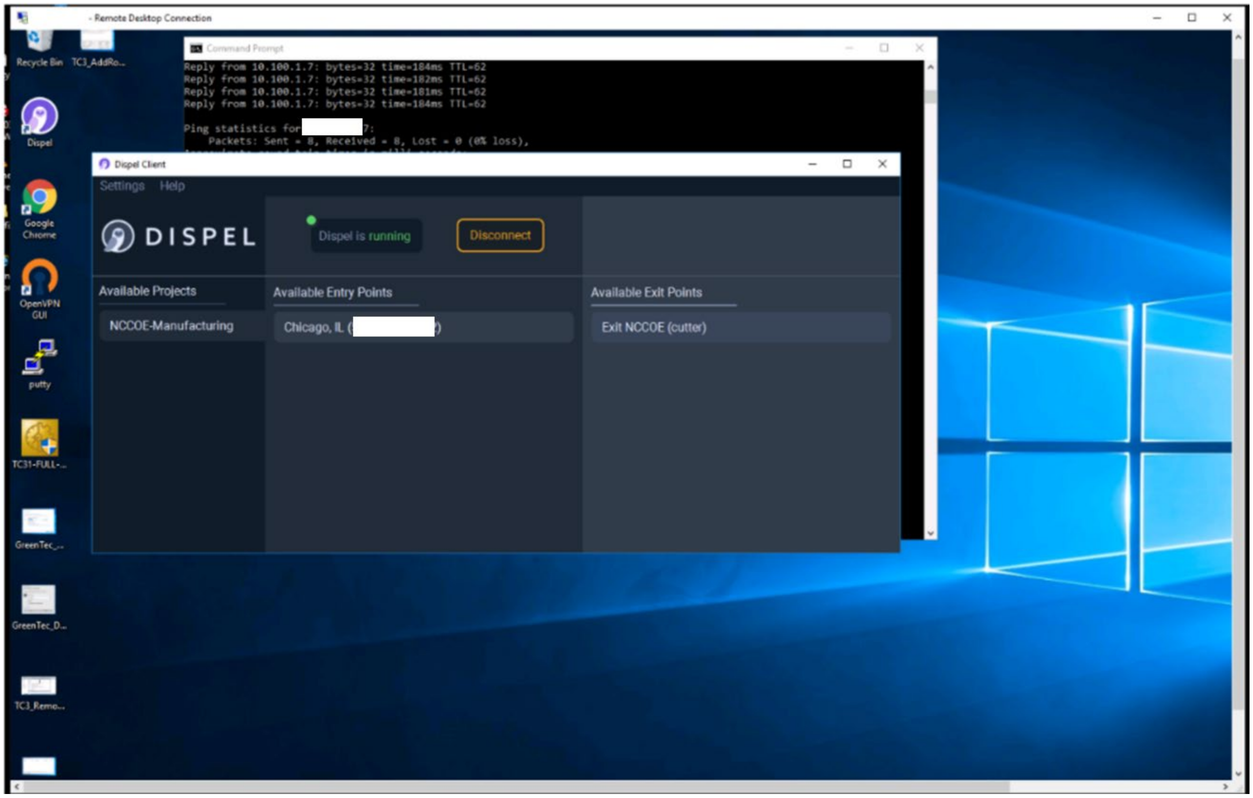
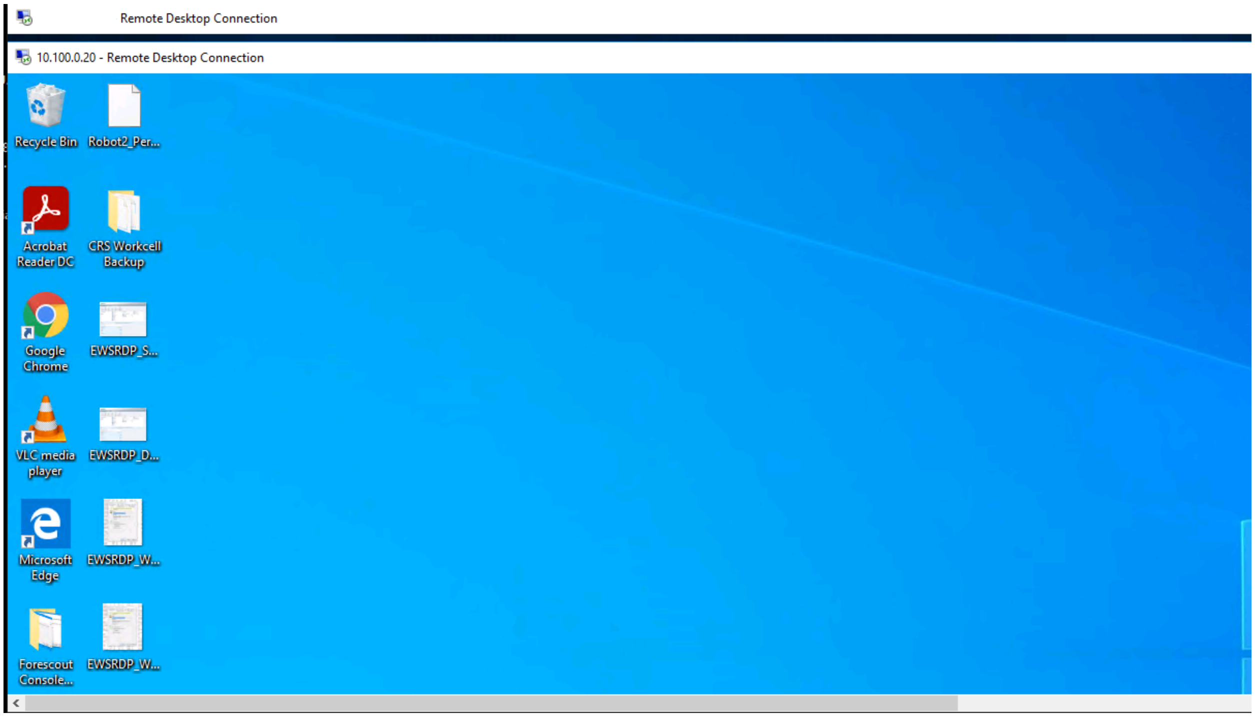


Figure D-38 Nested RDP Session Showing Dispel Connection into the CRS Workstation



D.4 Executing Scenario 4: Protect Host from Unauthorized Application Installation

An authorized user copies downloaded software installation files and executable files from a shared network drive to a workstation. The user attempts to execute or install the unauthorized software on the workstation. The expected result is that the application allowlisting tool prevents execution or installation of the software. Also, the behavioral anomaly detection identifies file transfer activity in the manufacturing environment.

D.4.1 Build 1

D.4.1.1 Configuration

- Application Allowlisting: Carbon Black
  - Agent installed on systems in the DMZ, Testbed LAN, and PCS VLAN 1 and 2 and configured to communicate to the Carbon Black Server.
- Behavior Anomaly Detection: Tenable.ot
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.4.1.2 Test Results

As shown in Figure D-39, Carbon Black is able to block and alert on the execution of putty.exe. Tenable.ot is able to detect the server message block (SMB) connection between an HMI in the Testbed LAN and the GreenTec server (Figure D-40). Details of that alert are shown in Figure D-41.

Figure D-39 Carbon Black Blocks the Execution of putty.exe and Other Files

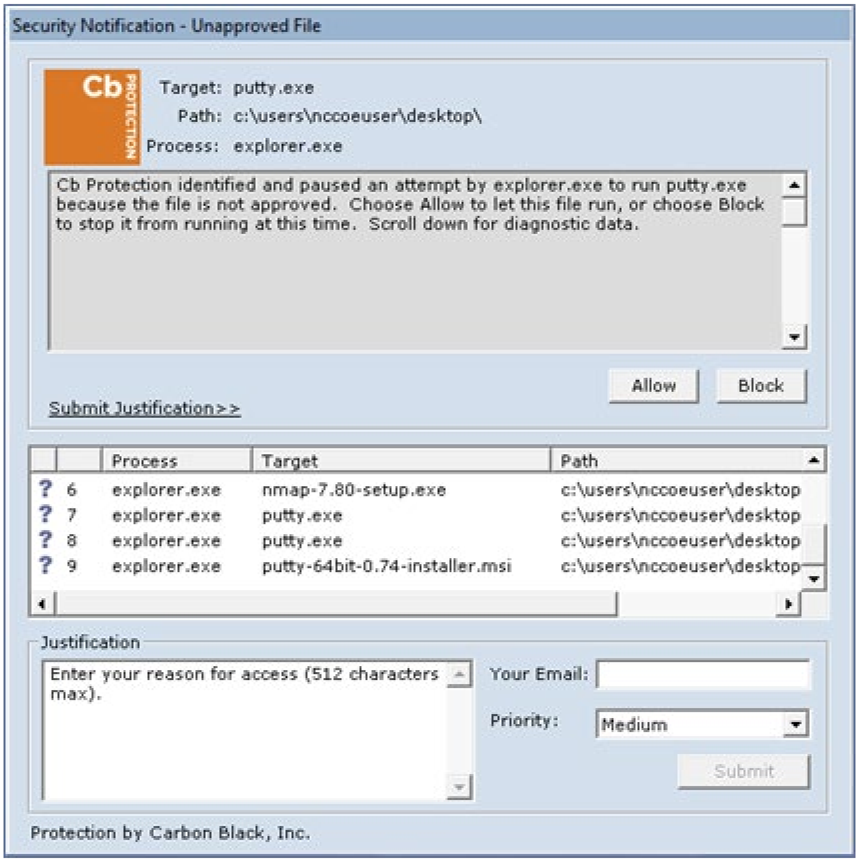


Figure D-40 Tenable.ot Alert With the SMB Connection Between the HMI and the GreenTec Server

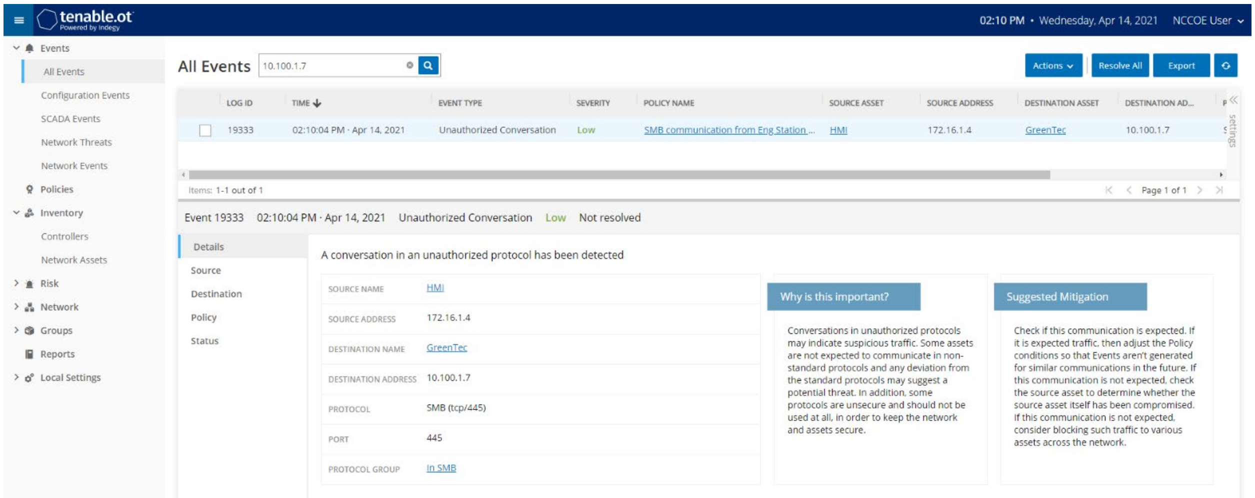
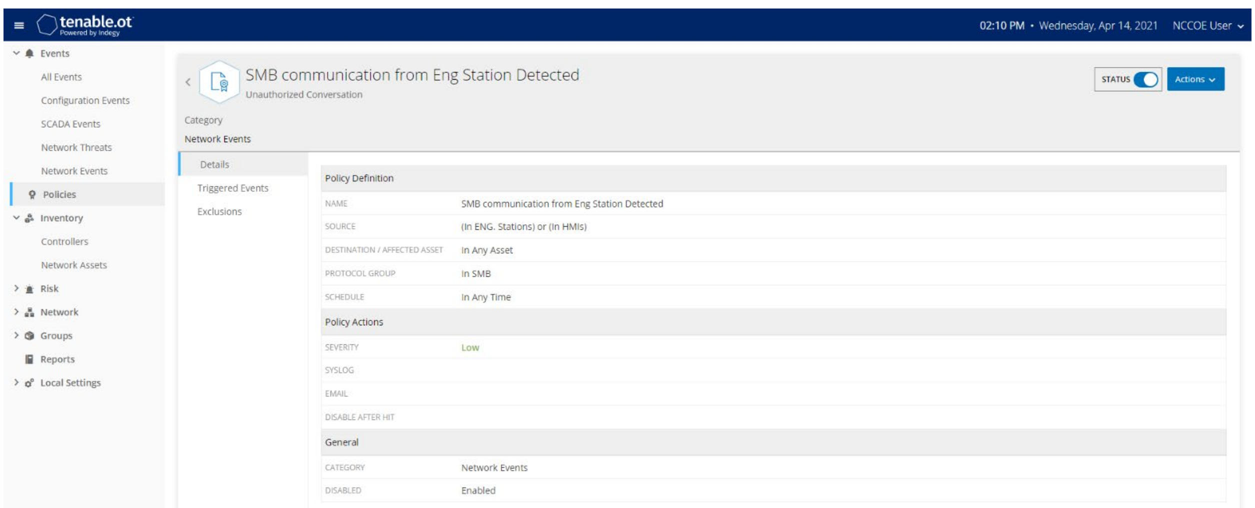


Figure D-41 Tenable.ot Alert Details of the SMB Connection Between the HMI and the network file system (NFS) Server in the DMZ



D.4.2 Build 2

D.4.2.1 Configuration

- Application Allowlisting: Windows SRP
    - Allowlisting policies are applied to systems in the DMZ, Testbed LAN, and PCS VLAN 1 and 2.

- Behavior Anomaly Detection: eyeInspect
    - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.4.2.2 Test Results

With Windows SRP enabled, putty.exe is not allowed to execute because it is not a permitted application under group policy, as shown in Figure D-42. Windows SRP also blocks the user's attempt to run putty-64bit-0.74-installer.msi. (Figure D-43). Forescout detected the file transfer activity (Figure D-44). Figure D-45 shows a detailed description of the alert that was generate for the file transfer activity.

Figure D-42 Putty.exe is Not Permitted to Run Based on the Windows SRP Configuration

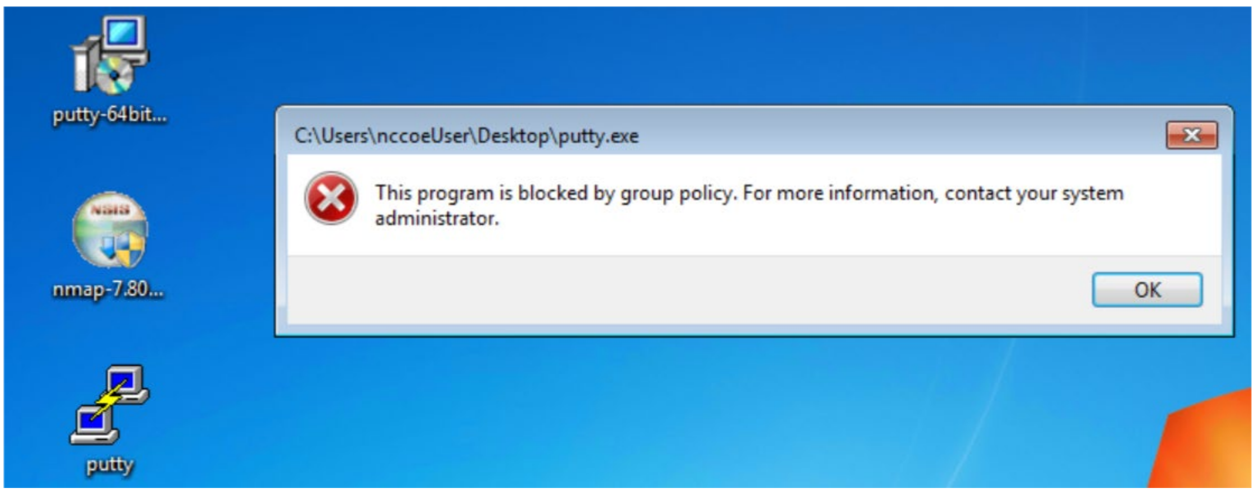


Figure D-43 putty-64bit-0.74-installer.msi is blocked by Windows SRP

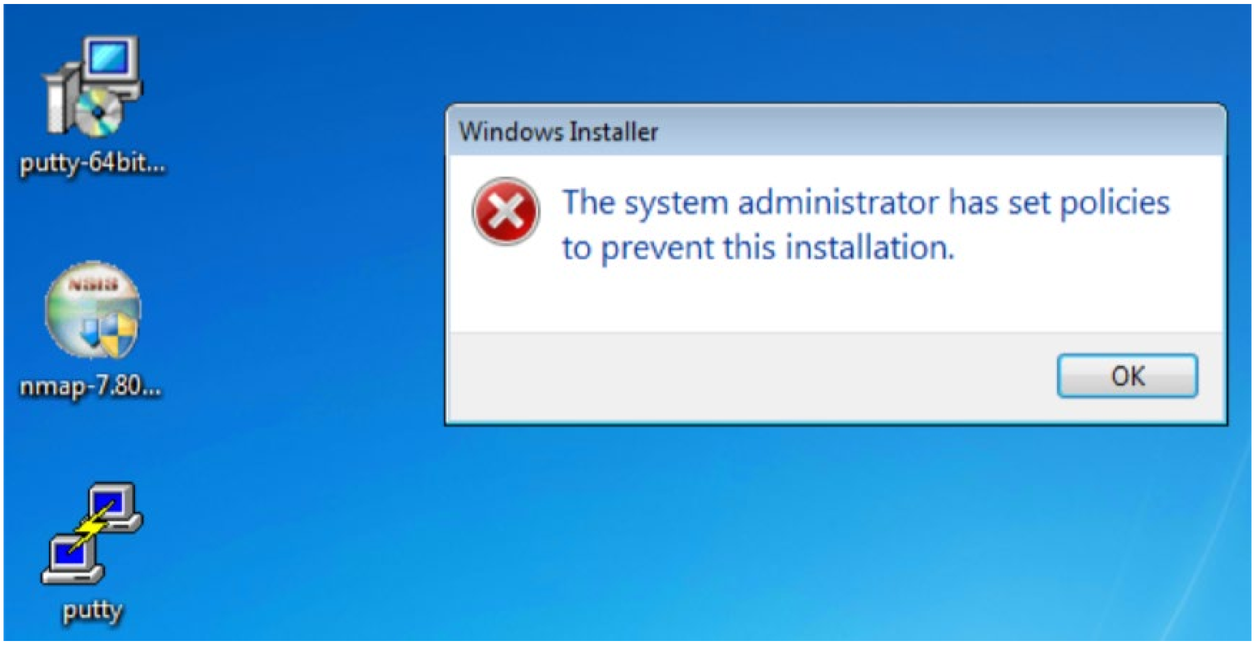
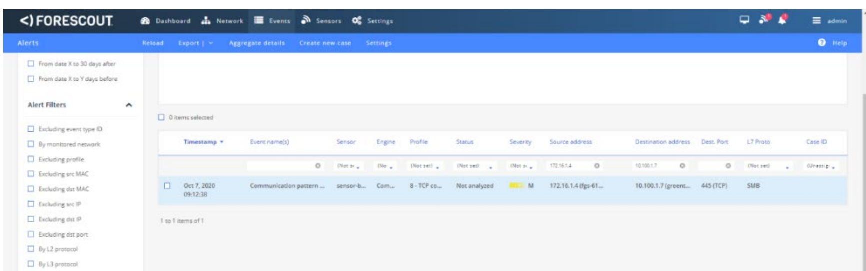
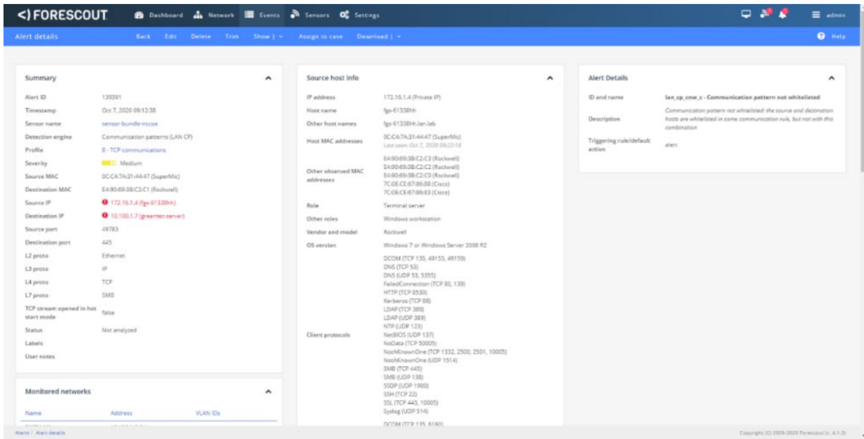


Figure D-44 Forescout Alert on the File Transfer Activity



This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Figure D-45 Forescout Alert Details for the File Transfer Activity



D.4.3 Build 3

D.4.3.1 Configuration

- Application Allowlisting : Windows SRP
  - Settings are applied to systems in the DMZ, Testbed LAN, and Supervisory LAN
- Behavior Anomaly Detection: Dragos
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.4.3.2 Test Results

With Windows SRP enabled, putty.exe is not allowed to execute because it is not a permitted application under group policy, as shown in Figure D-46. Windows SRP also blocks the user's attempt to run putty-64bit-0.74-installer.msi (Figure D-47). Dragos detected the file transfer activity (Figure D-48). Figure D-49 shows a detailed description of the alert that was generated for the file transfer activity.

Figure D-46 Putty.exe is Not Permitted to Run Based on the Windows SRP Configuration

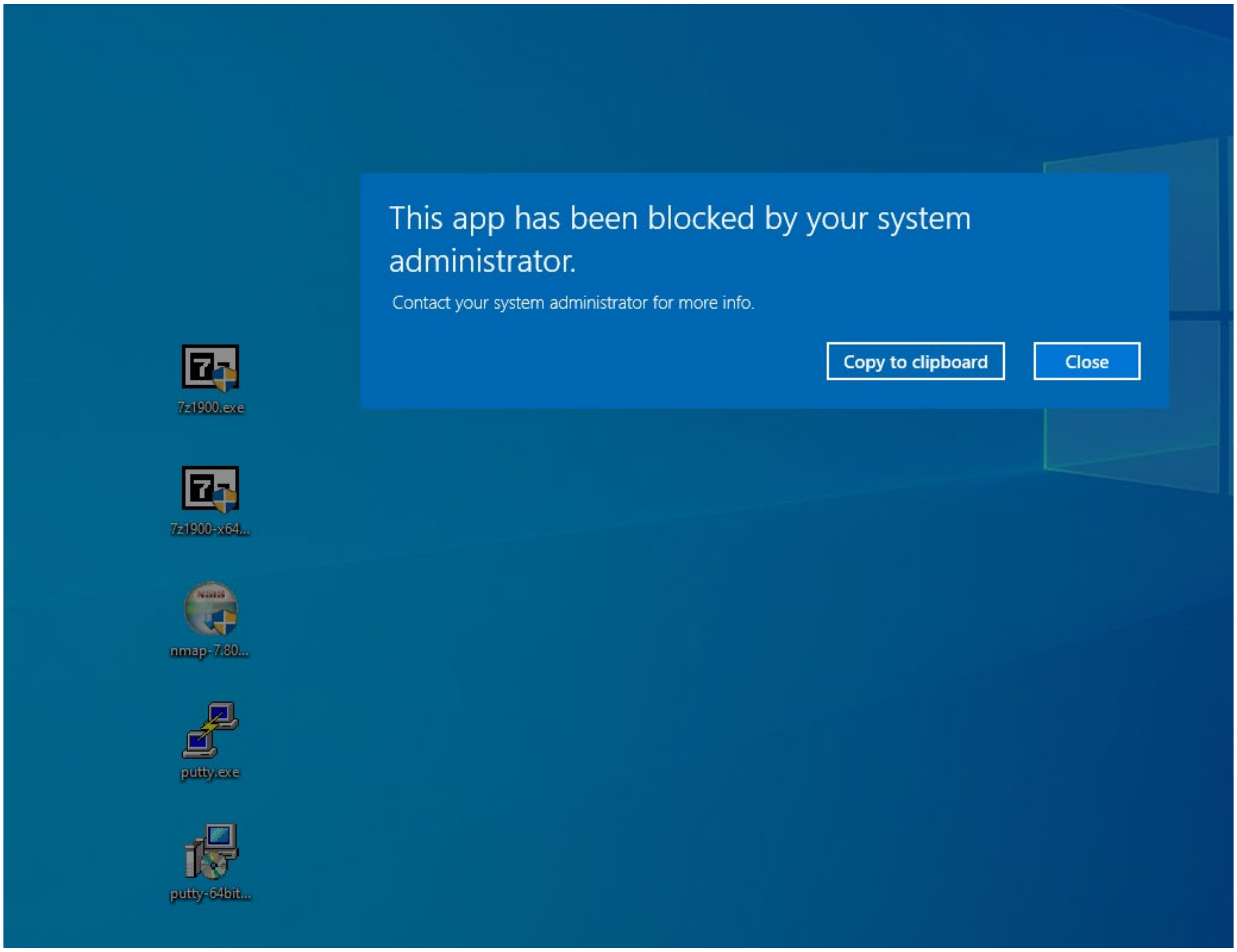


Figure D-47 putty-64bit-0.74-installer.msi is Blocked by Windows SRP

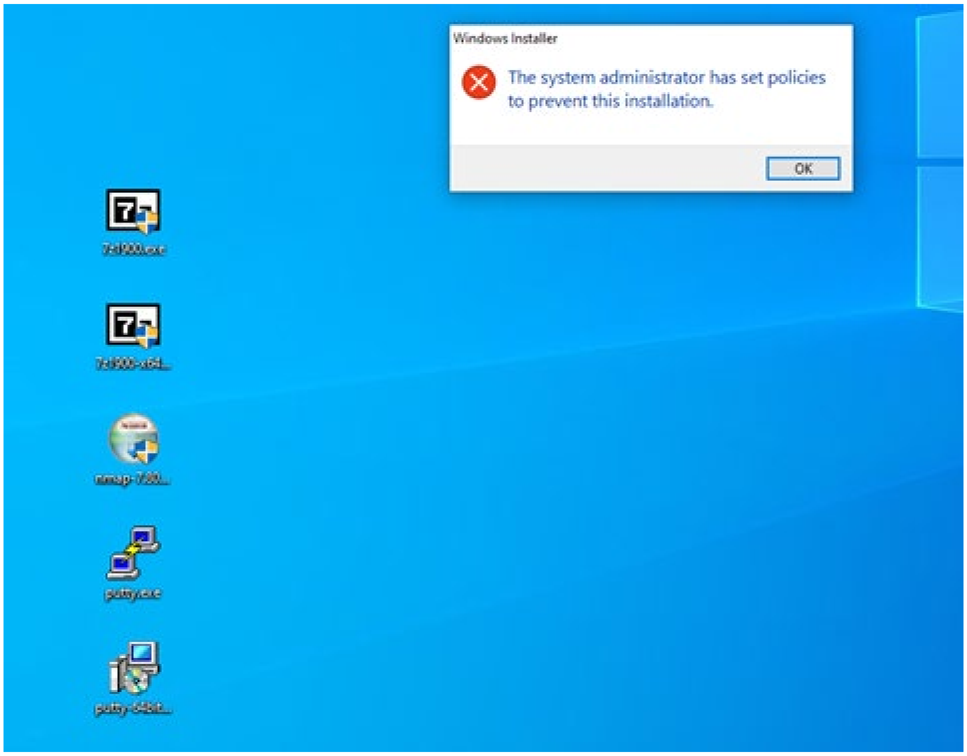


Figure D-48 Dragos Alert on the File Transfer Activity

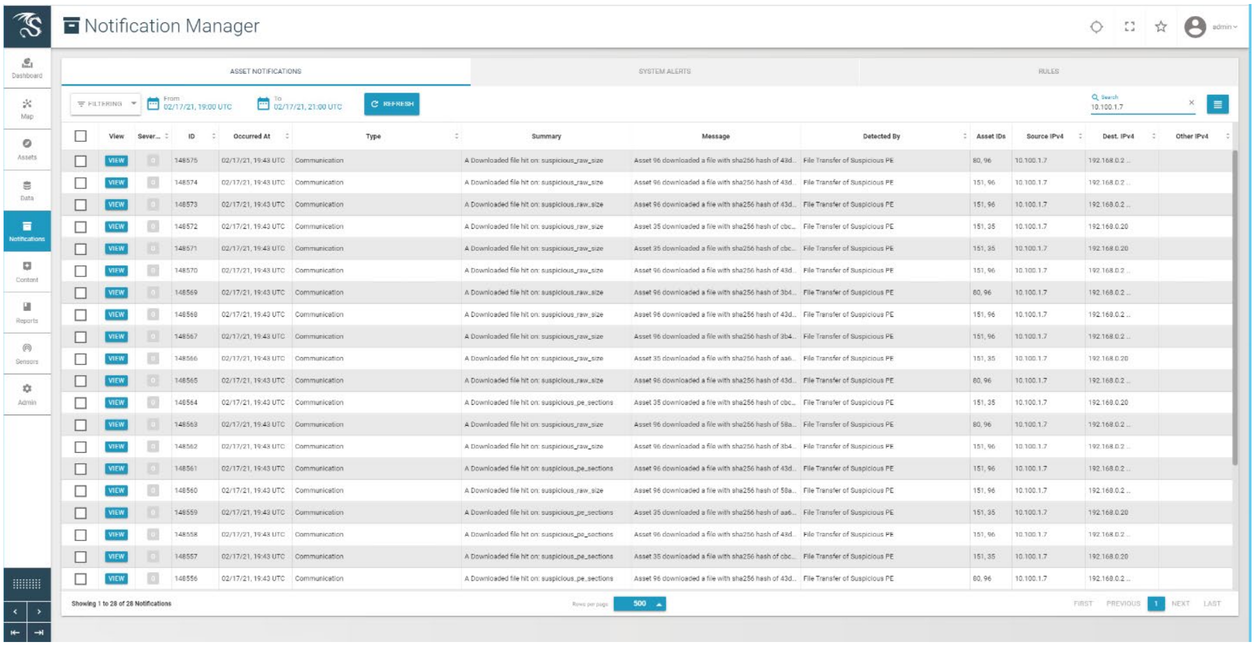
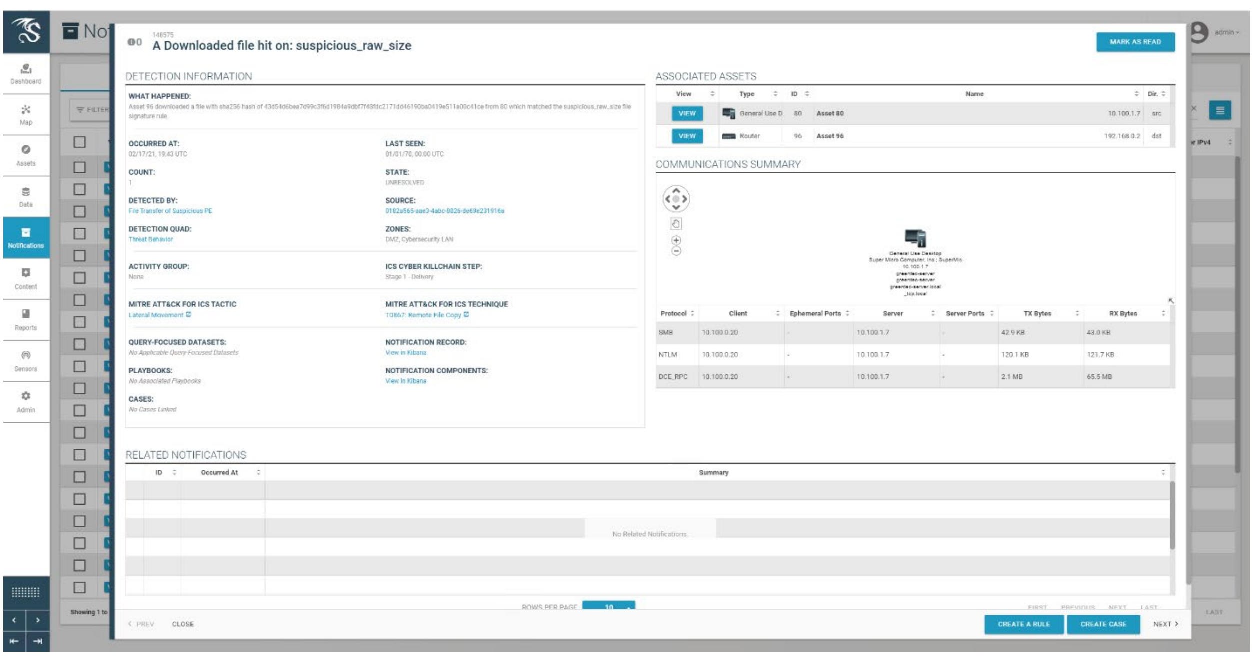


Figure D-49 Dragos Alert Details of the File Transfer Alert



D.4.4 Build 4

D.4.4.1 Configuration

- Application Allowlisting: Carbon Black

  - Agent installed on systems in the DMZ, Testbed LAN, and Supervisory LAN and configured to communicate to the Carbon Black Server.

- Behavior Anomaly Detection: Azure Defender for IoT

Configured to receive packet streams from DMZ, Testbed LAN and Supervisory LAN, and Control LAN.

D.4.4.2 Test Results

Carbon Black was able to block execution of putty.exe (Figure D-50) and the installation of putty-64bit-0.74-installer.msi (Figure D-51). Figure D-52 is the alert dashboard for Azure Defender for IoT that shows new activity has been detected. The detailed alert in Figure D-53 provides details of an RPC connection between the GreenTec server and the Testbed LAN. A timeline of events showing a file transfer has occurred is shown in Figure D-54.

Figure D-50 Carbon Black Alert Showing that putty.exe is Blocked from Executing

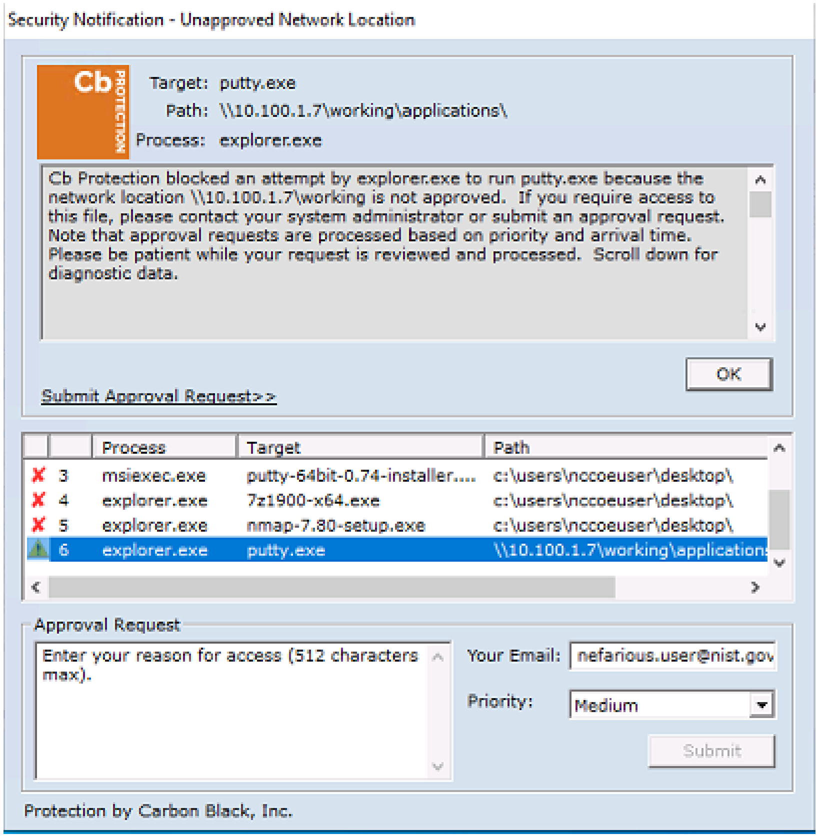


Figure D-51 Carbon Black Alert Showing Execution of putty-64bit-0.74-installer.msi Being Blocked

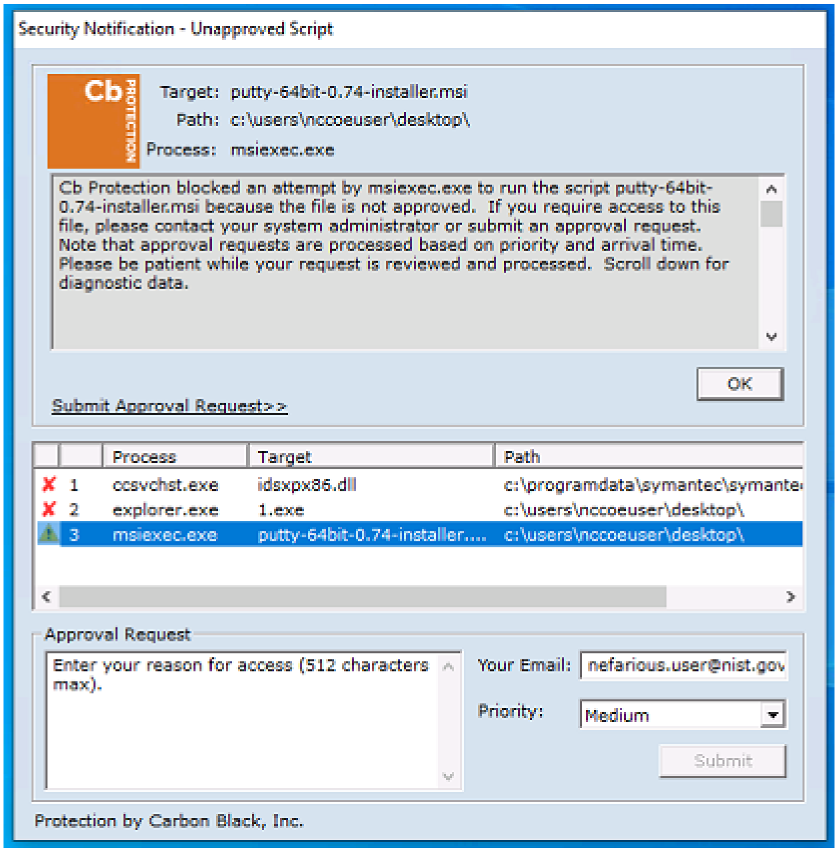


Figure D-52 Azure Defender for IoT Alert Dashboard Showing Detection of a New Activity

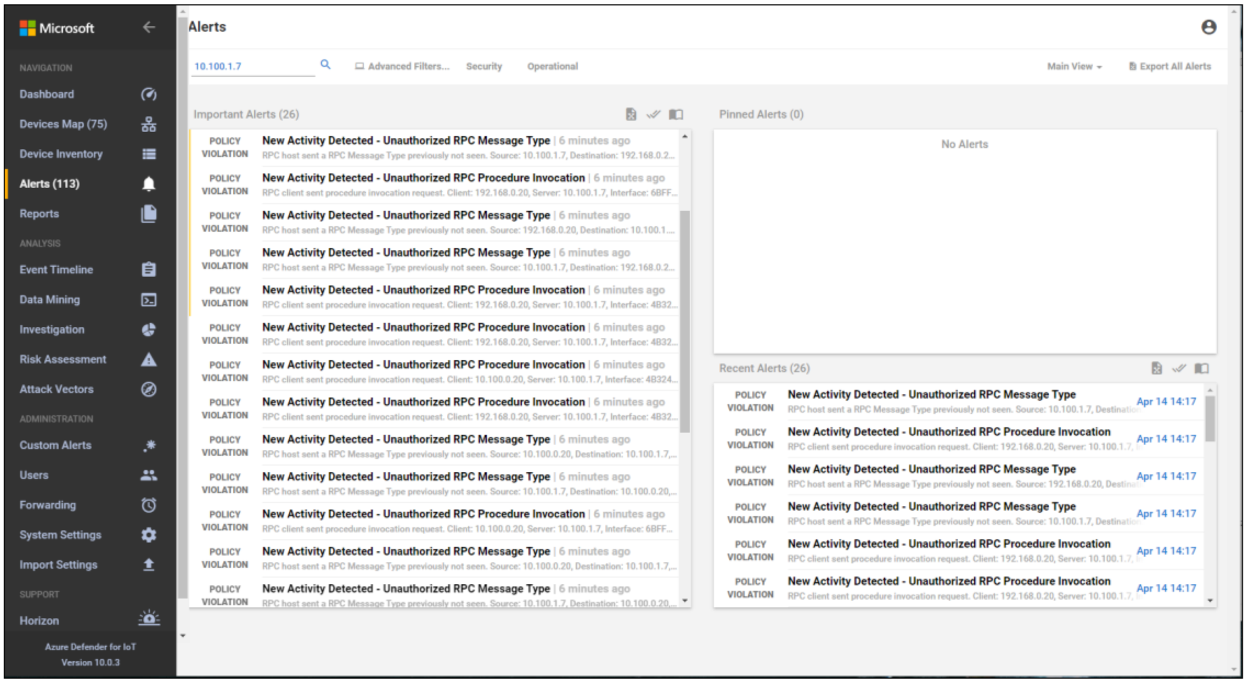


Figure D-53 Azure Defender for IoT Alert Details Showing RPC Connection Between the DMZ and the Testbed LAN

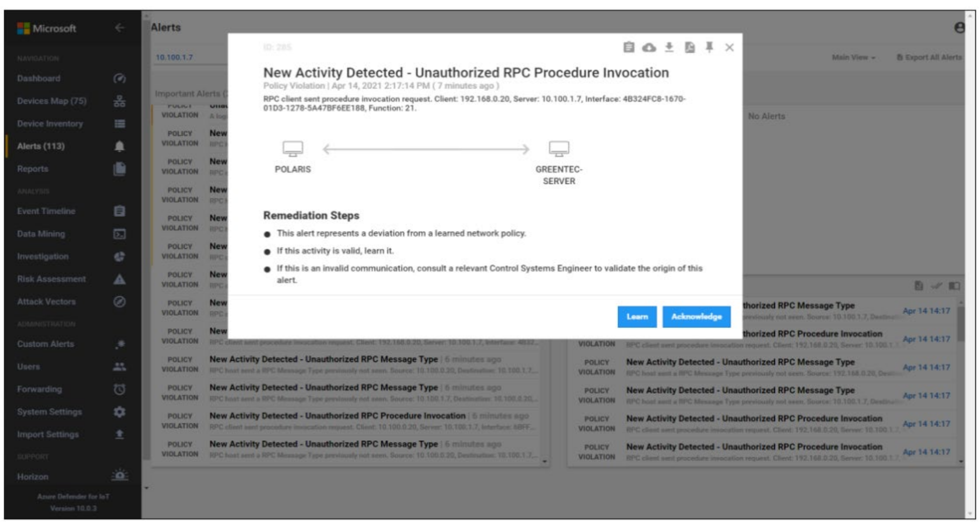
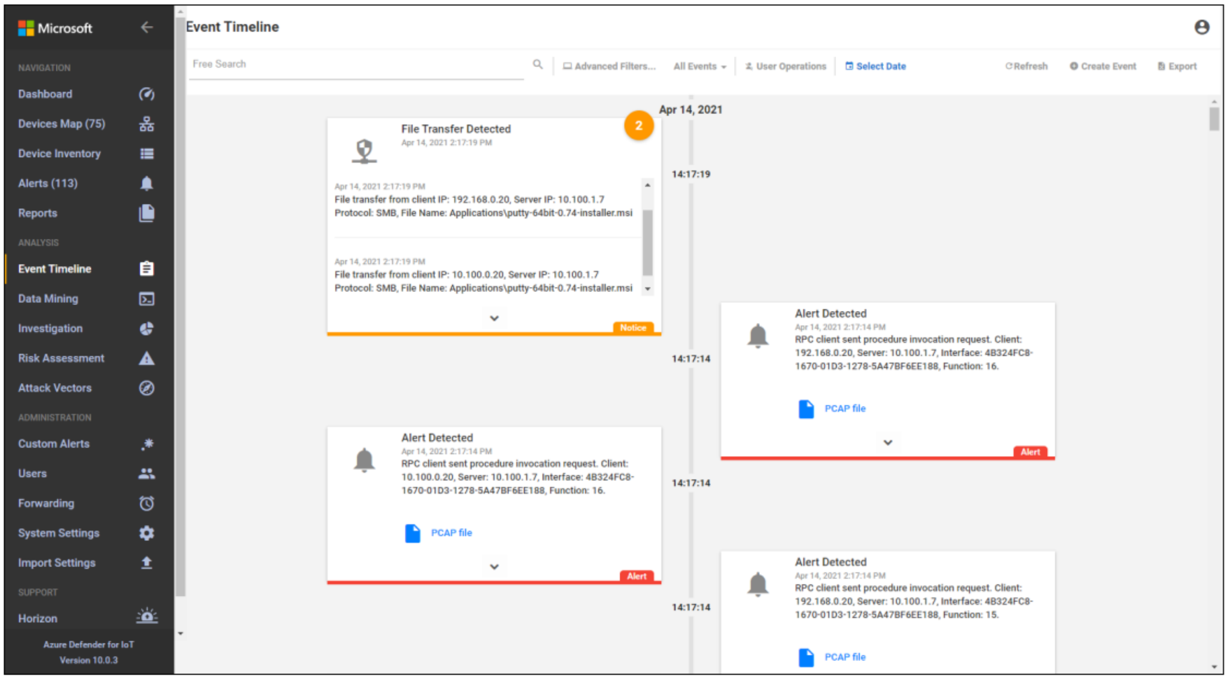


Figure D-54 Azure Defender for IoT Event Alert Timeline Showing the File Transfer



D.5 Executing Scenario 5: Protect from Unauthorized Addition of a Device

An authorized individual with physical access connects an unauthorized device on the manufacturing network and then uses it to connect to devices and scan the network. The expected result is behavioral anomaly detection identifies the unauthorized device.

D.5.1 Build 1

D.5.1.1 Configuration

- Behavior Anomaly Detection: Tenable.ot
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.5.1.2 Test Results

Tenable.ot detects and alerts on addition of a device to the environment. Figure D-55 shows an event reported by Tenable.ot when a device was connected to the wireless access point in the manufacturing environment. Tenable.ot also detects other activity from the device, as shown in Figure D-56, where the new device tries to establish a secure shell (SSH) connection to the network switch.

Figure D-55 Tenable.ot Event Showing a New Asset has Been Discovered

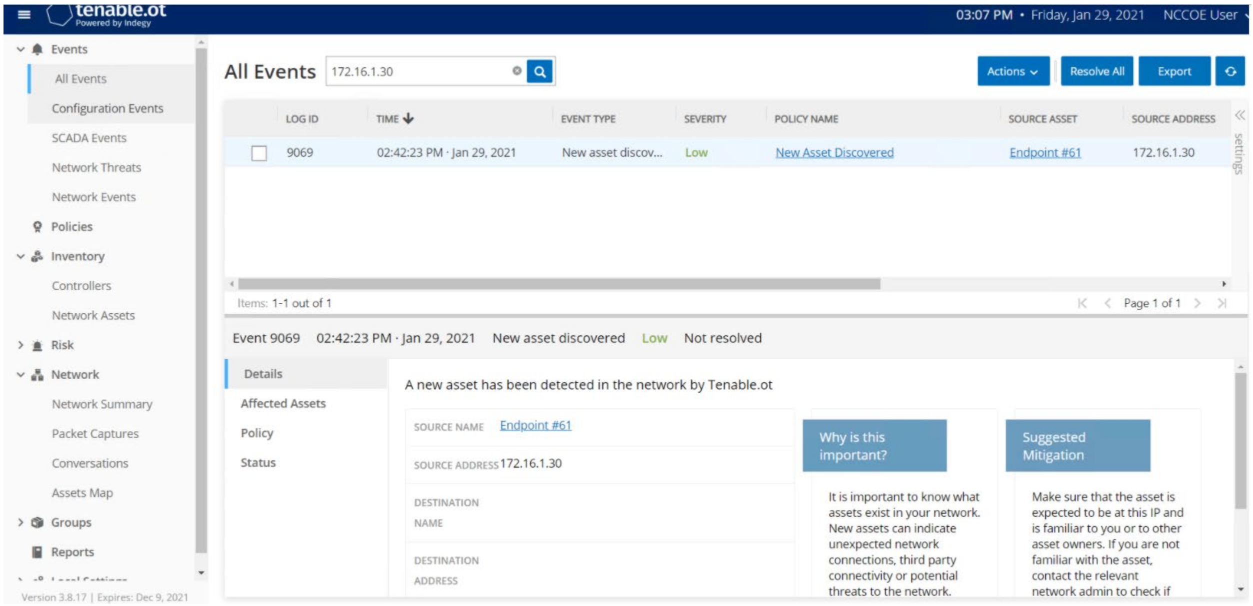
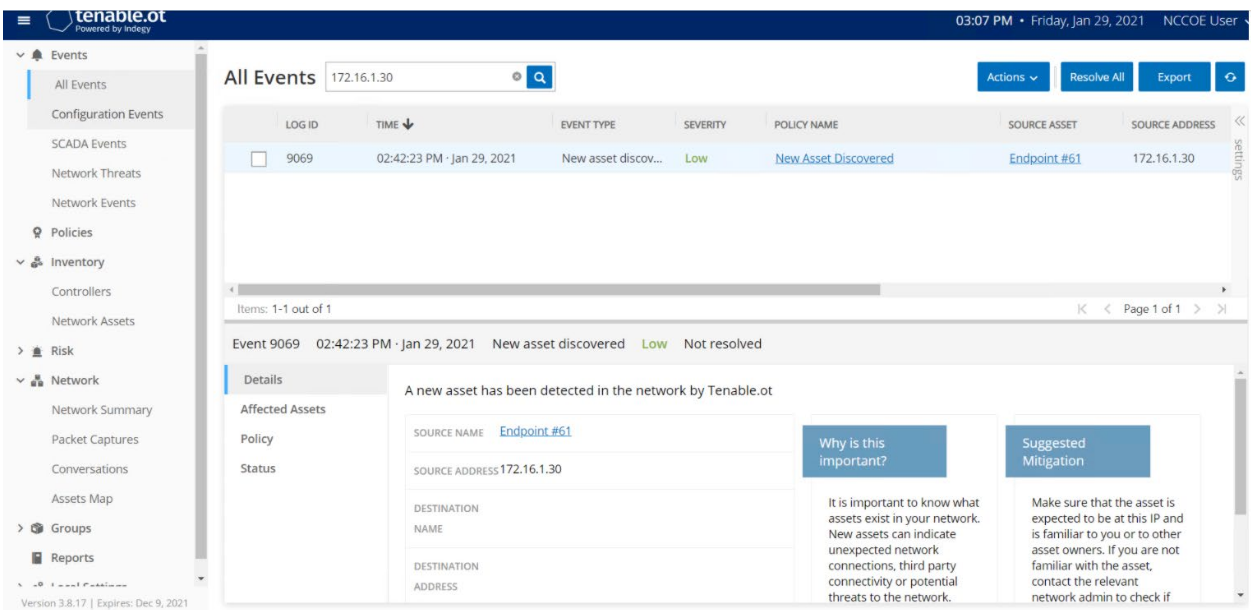


Figure D-56 Tenable.ot Event Showing Unauthorized SSH Activities



D.5.2 Build 2

D.5.2.1 Configuration

- Behavior Anomaly Detection: eyeInspect
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.5.2.2 Test Results

Forescout detects when an unauthorized device connects to a wireless access point in the manufacturing environment. Figure D-57 shows that Forescout raises an alert on the DNS request from the wireless access point to the gateway. The device establishes an SSH connection, which is detected by Forescout as shown in Figure D-58. A more detailed view of the alert is shown in Figure D-59.

Figure D-57 Forescout Alert on the DNS Request from the New Device

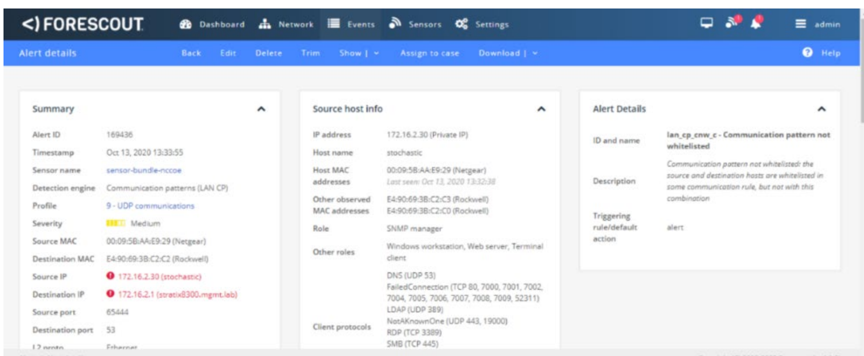
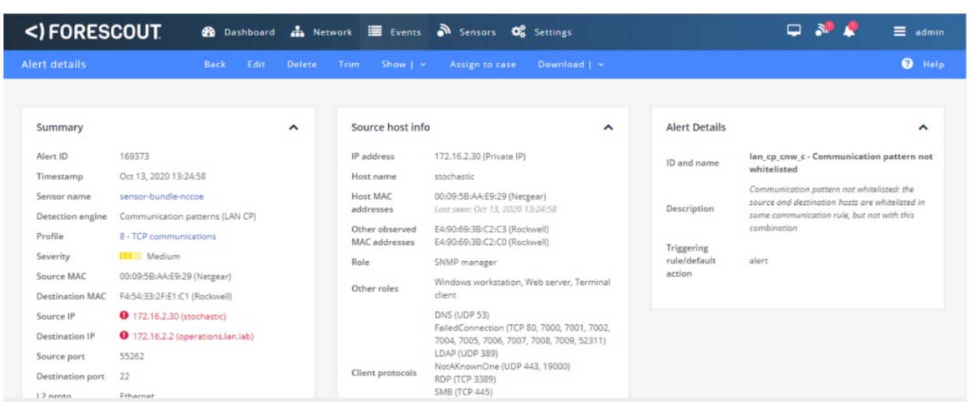


Figure D-58 Forescout alert showing the SSH connection

Figure D-59 Detailed Forescout alert of the Unauthorized SSH Connection



This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

D.5.3 Build 3

D.5.3.1 Configuration

- Behavior Anomaly Detection: Dragos
    - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.5.3.2 Test Results

Dragos detected the traffic generated by the new asset and generated several alerts as seen in the list of alerts in Figure D-60. Details of different aspects of the network scanning can be seen in Figure D-61 and Figure D-62. Details on the new device can also be seen in Figure D-63.

Figure D-60 Dragos Dashboard Showing Alerts Generated upon Detecting New Device and Network Scanning

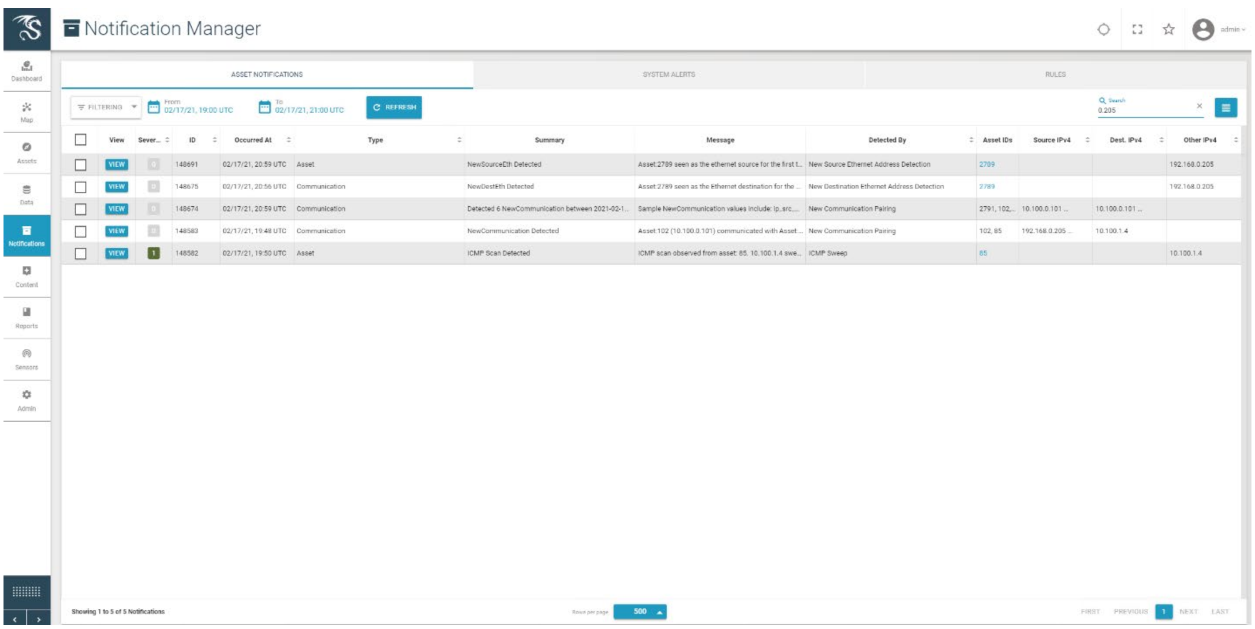


Figure D-61 Details of Network Scanning Activity

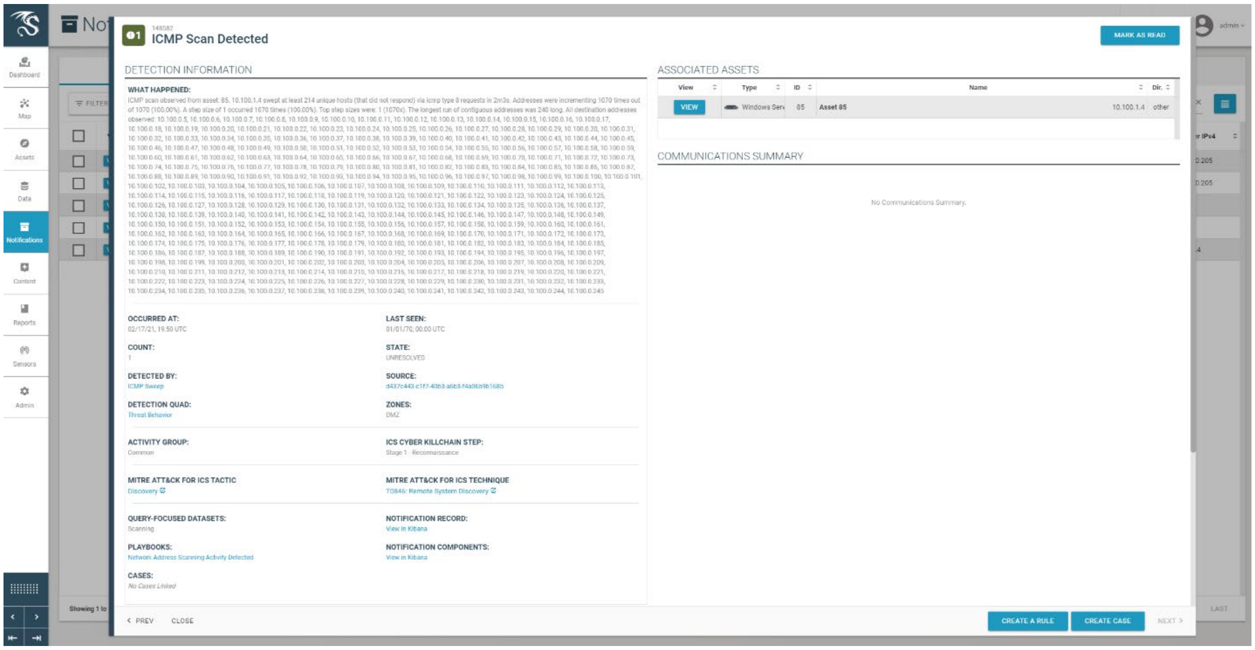


Figure D-62 Additional Details of Network Scanning Activity

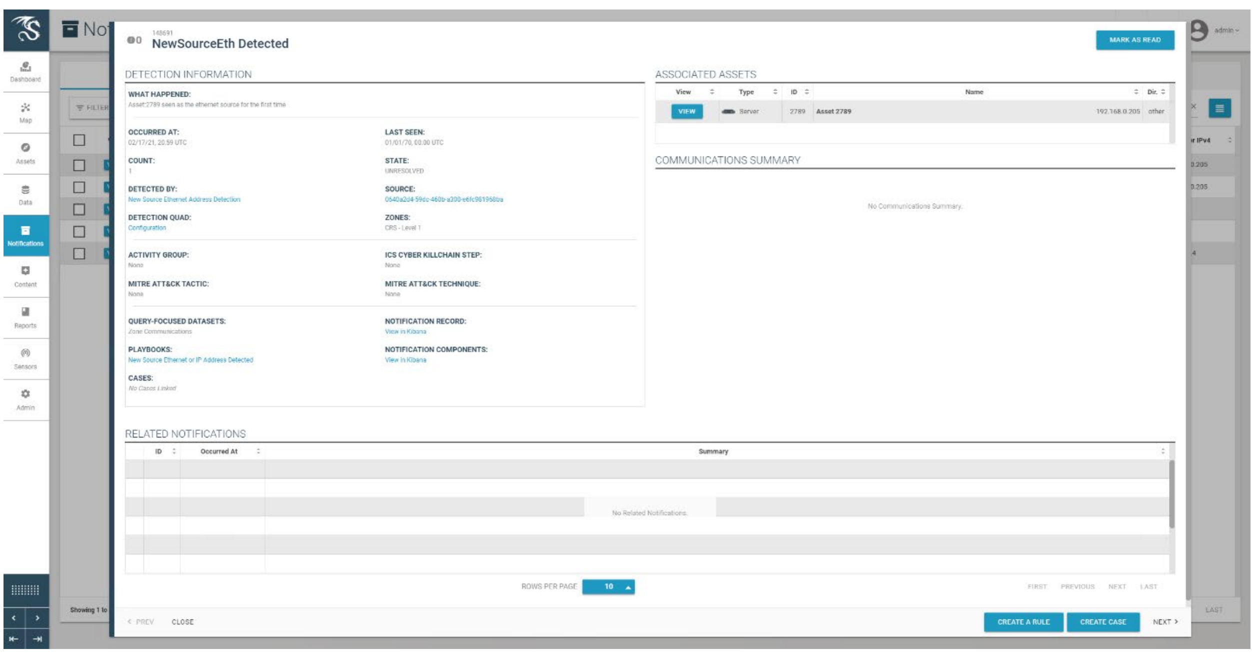
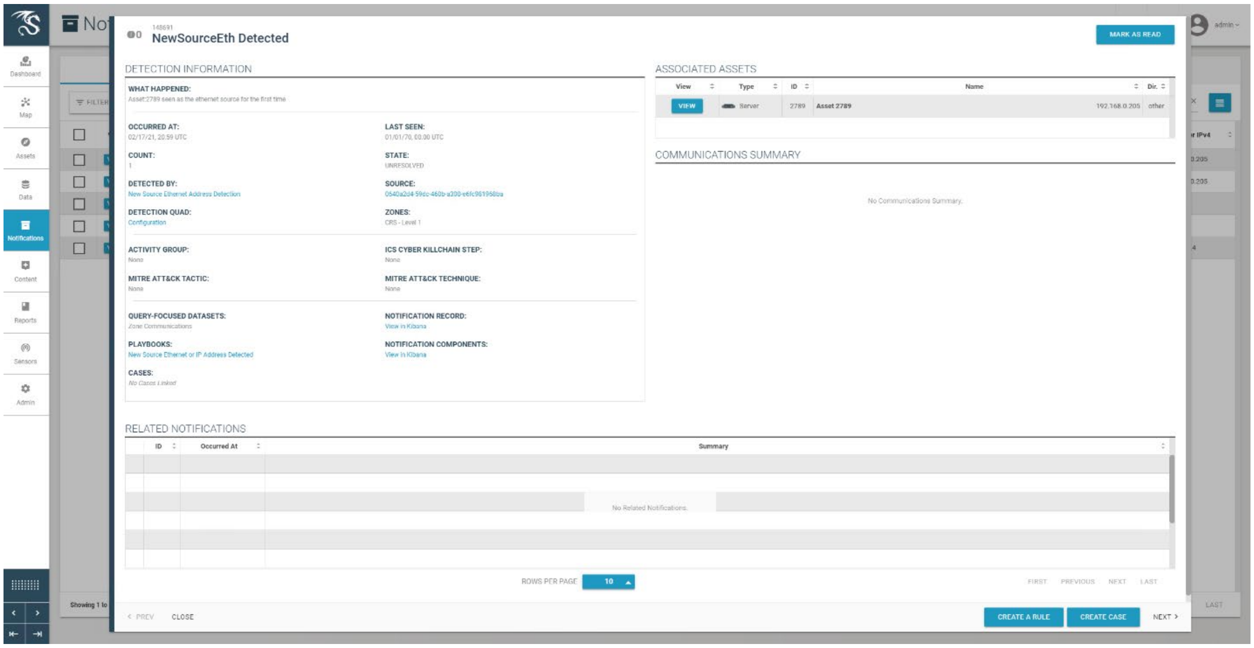


Figure D-63 Alert for New Asset on the Network



D.5.4 Build 4

D.5.4.1 Configuration

- Behavior Anomaly Detection: Azure Defender for IoT
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.5.4.2 Test Results

A “New Asset Detected” alert is shown on Azure Defender for IoT dashboard (Figure D-64) and on the Alert screen (Figure D-65). Figure D-66 shows the alert management options in Azure Defender for IoT. The details of the network scanning alert are shown in Figure D-67.

Figure D-64 Azure Defender for IoT Dashboard Showing the Alerts, Including for the New Asset

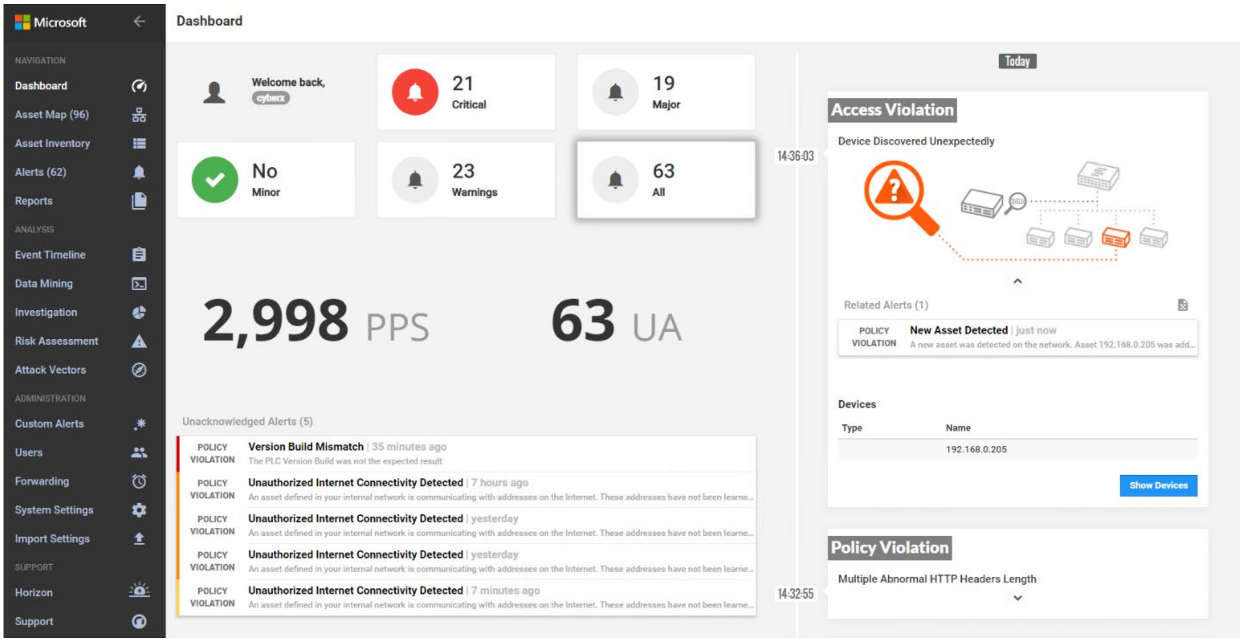


Figure D-65 Azure Defender for IoT Detects New Asset in the Environment

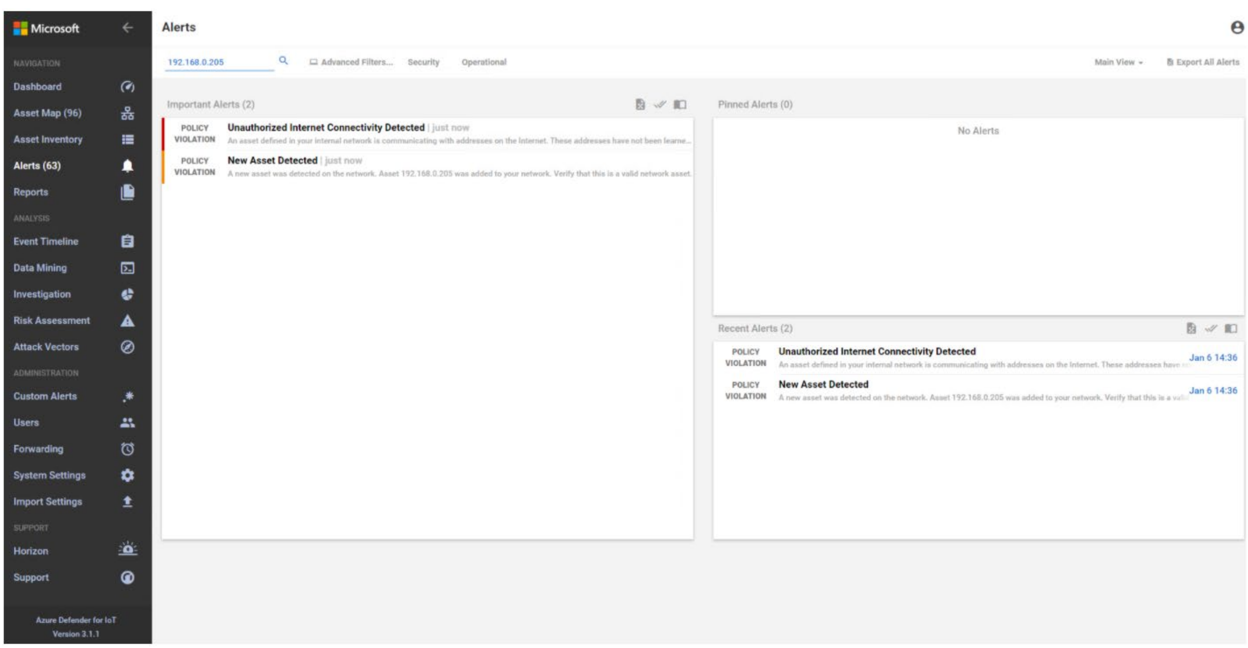


Figure D-66 Azure Defender for IoT Alert Management Options

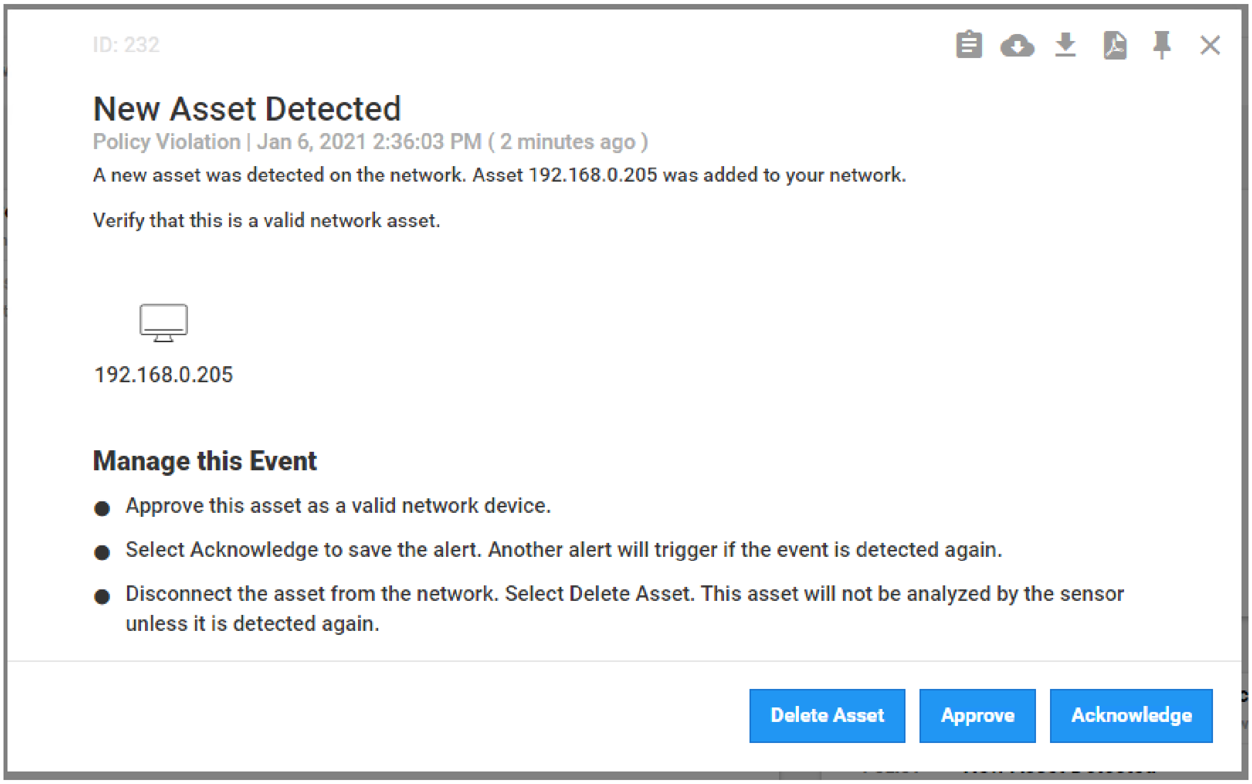
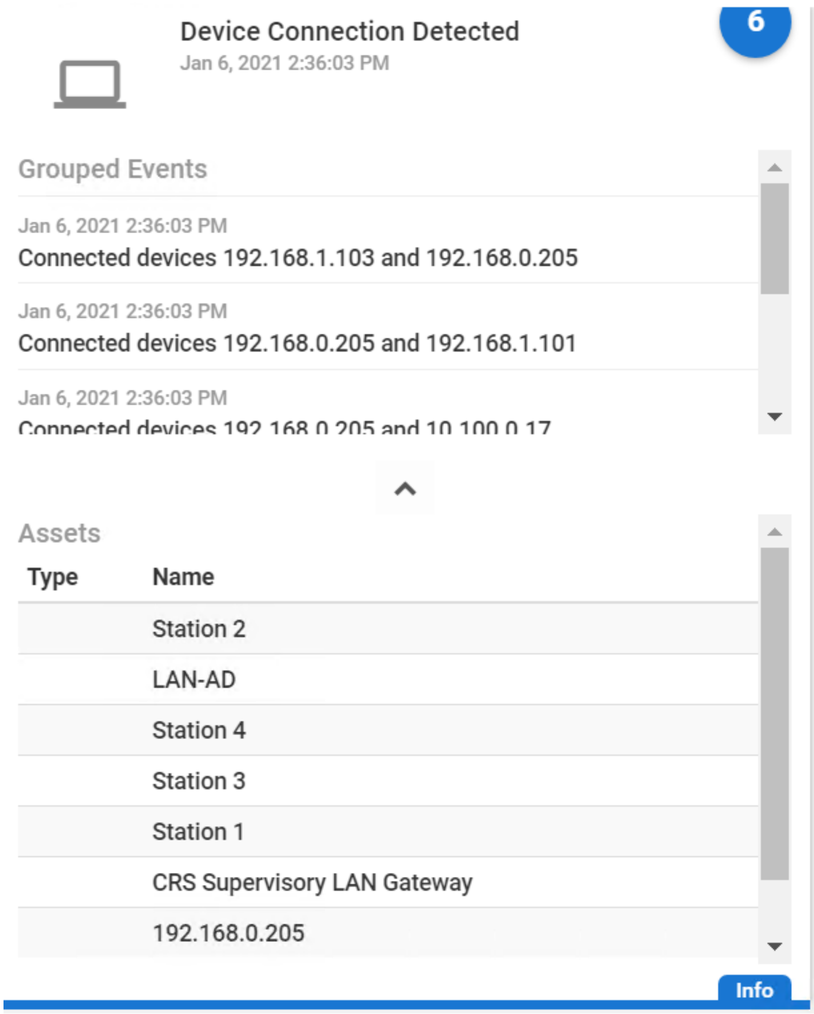


Figure D-67 Details for Network Scanning Alert



D.6 Executing Scenario 6: Detect Unauthorized Device-to-Device Communications

An authorized device that is installed on the network attempts to establish an unapproved connection that is not recorded in the baseline. The expected result is the behavioral anomaly detection products alert on the non-baseline network traffic.

D.6.1 Build 1

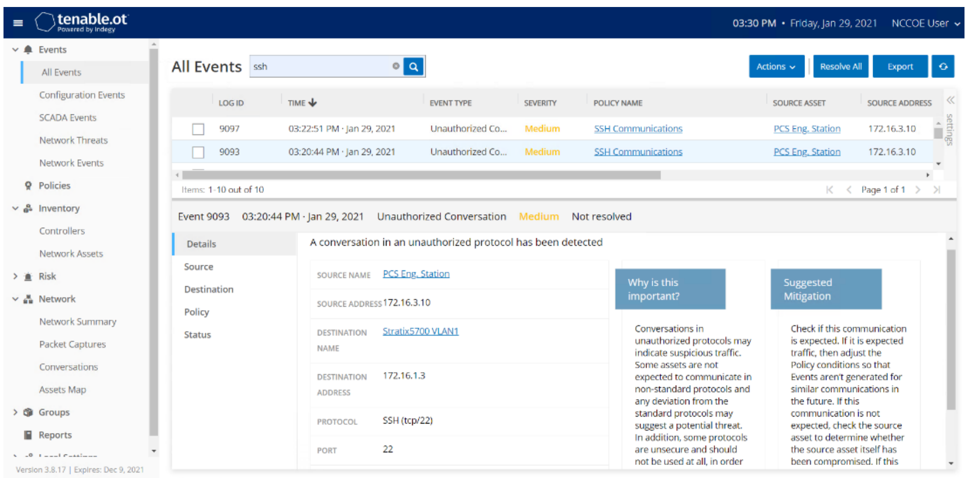
D.6.1.1 Configuration

- Behavior Anomaly Detection: Tenable.ot
    - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.6.1.2 Test Results

The unapproved SSH traffic is detected by Tenable.ot as shown in Figure D-68.

Figure D-68 Tenable.ot Event Log Showing the Unapproved SSH Traffic



D.6.2 Build 2

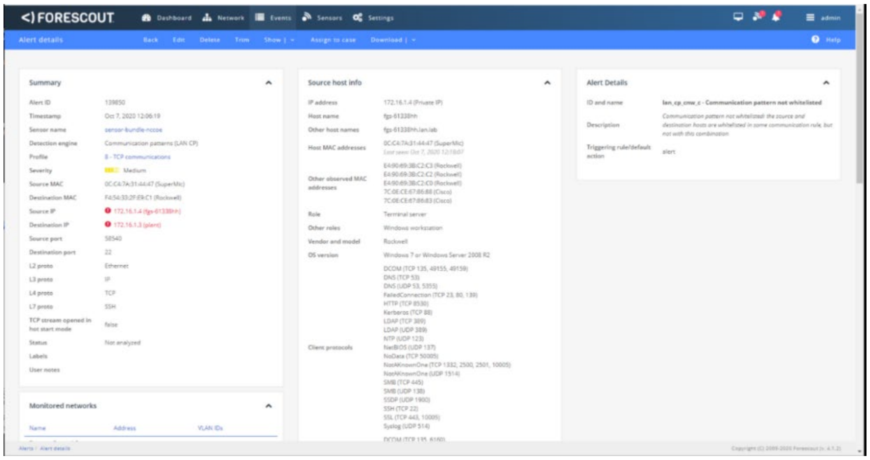
D.6.2.1 Configuration

- Behavior Anomaly Detection: eyeInspect
    - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

D.6.2.2 Test Results

SSH communication from HMI computer to the network switch is not defined in the baseline; Forescout flags this communication as shown in Figure D-69.

Figure D-69 Forescout Alert Showing the Unapproved SSH Traffic



D.6.3 Build 3

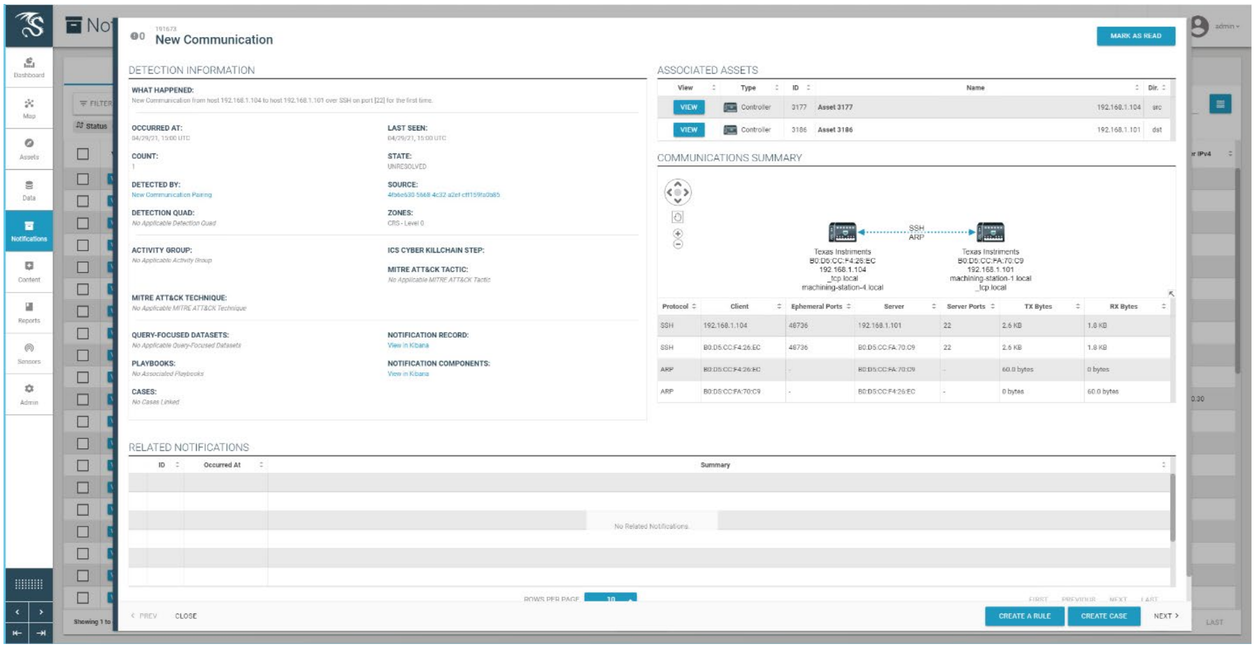
D.6.3.1 Configuration

- Behavior Anomaly Detection: Dragos
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.6.3.2 Test Results

Dragos detected the non-baseline SSH traffic as shown in Figure D-70.

Figure D-70 Dragos Alert Showing the Unapproved SSH Connection Between Devices



D.6.4 Build 4

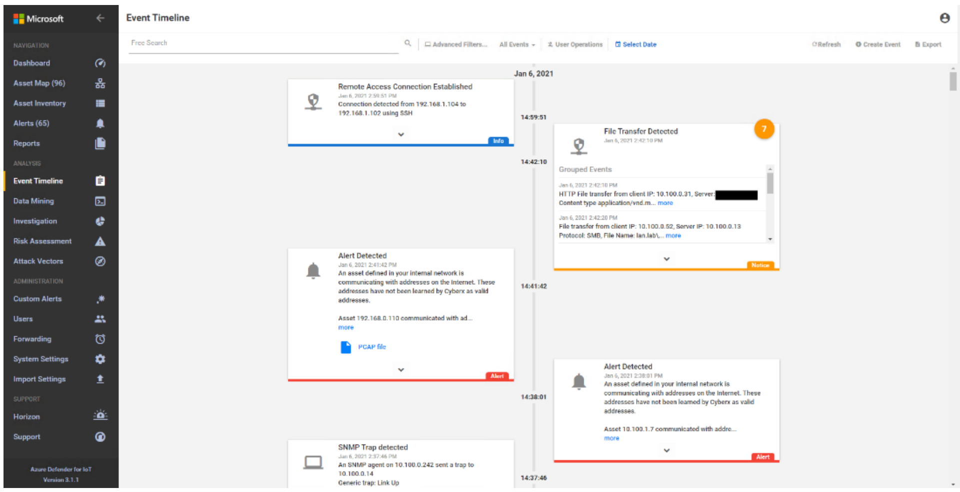
D.6.4.1 Configuration

- Behavior Anomaly Detection: Azure Defender for IoT
    - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.

D.6.4.2 Test Results

A device attempts to establish a remote access connection via SSH. Azure Defender for IoT was able to detect this activity as shown in Figure D-71.

FigureD-71 Azure Defender for IoT Event Identified the Unauthorized SSH Connection



D.7 Executing Scenario 7: Protect from Unauthorized Deletion of Files

An authorized user attempts to delete files on an engineering workstation and a shared network drive within the manufacturing system. The expected result is the file integrity checking tools in the environment alert on the deletion or prevent deletion entirely.

D.7.1 Build 1

D.7.1.1 Configuration

- File Integrity Checking: Carbon Black
  - Agent installed on workstations and configured to communicate to the Carbon Black Server.
- File Integrity Checking: WORMdisk
  - Network file share on server is configured to use WORMdisk.

D.7.1.2 Test Results

Carbon Black reports file deleting activities as shown in Figure D-72. GreenTec protects the files on its drive from being deleted.

Figure D-72 Event Messages from Carbon Black Showing File Deletion Attempts

Timestamp	Se...	Type	Subtype	Source	Description	IP Address	User	Process Nar
Feb 3 2021 01:35:55 PM	Info	Policy Enforcement	Report write (Custom Rule)	LAN\FGS-47631EHH	'c:\users\administrator\downloads\ra\nccoe_test_file.txt' was deleted by 'FGS-47631EHH\Administrator'.	172.16.3.10	FGS-47631EHH\Admini...	explorer.exe
Feb 3 2021 01:35:50 PM	Info	Policy Enforcement	Report write (Custom Rule)	LAN\FGS-47631EHH	'c:\users\administrator\downloads\ra\testscenarios\nccoe_test_file.txt' was deleted by 'FGS-47631EHH\Administrator'.	172.16.3.10	FGS-47631EHH\Admini...	explorer.exe
Feb 3 2021 01:35:35 PM	Info	Policy Enforcement	Report write (Custom Rule)	LAN\FGS-47631EHH	'c:\users\administrator\documents\tesim\nccoe_test_file.txt' was deleted by 'FGS-47631EHH\Administrator'.	172.16.3.10	FGS-47631EHH\Admini...	explorer.exe

D.7.2 Build 2

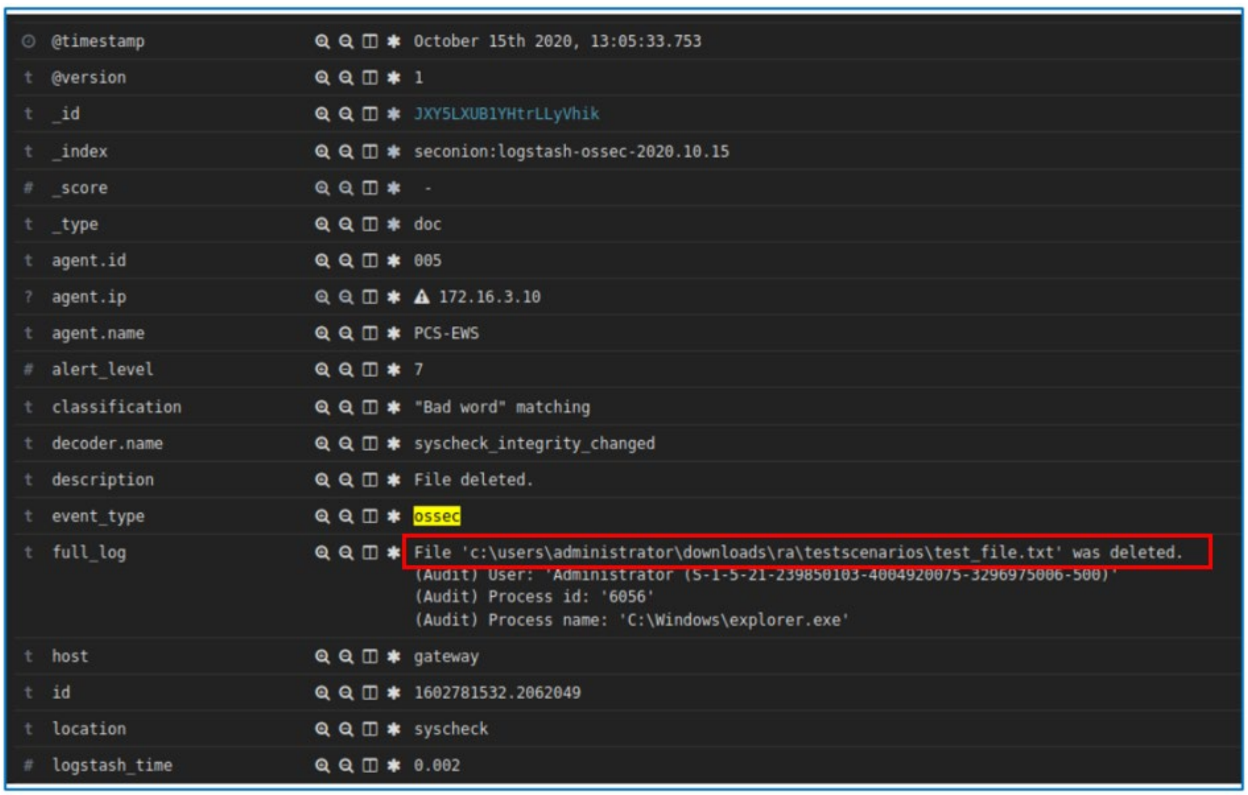
D.7.2.1 Configuration

- File Integrity Checking: Security Onion
  - The agent is installed on workstations and configured to communicate to the Security Onion Server.
- File Integrity Checking: WORMdisk
  - Network file share on server is configured to use WORMdisk.

D.7.2.2 Test Results

Security Onion Wazuh alerts on file deletion as shown in Figure D-73. Files stored on a storage drive protected by GreenTec are protected from deletion.

Figure D-73 Security Onion Wazuh Alert Showing a File Has Been Deleted



D.7.3 Build 3

D.7.3.1 Configuration

- File Integrity Checking: Security Onion
  - Agent installed on workstations and configured to communicate to the Security Onion Server.
- File Integrity Checking: WORMdisk
  - Network file share on server is configured to use WORMdisk.

D.7.3.2 Test Results

Security Onion Wazuh detected the file deletions as shown in the Security Onion Server log in Figure D-74. Files stored on a storage drive protected by GreenTec are protected from deletion.

Figure D-74 Alert from Security Onion for a File Deletion

D.7.4 Build 4

D.7.4.1 Configuration

- File Integrity Checking: Carbon Black

- Agent installed on workstations and configured to communicate to the Carbon Black Server.

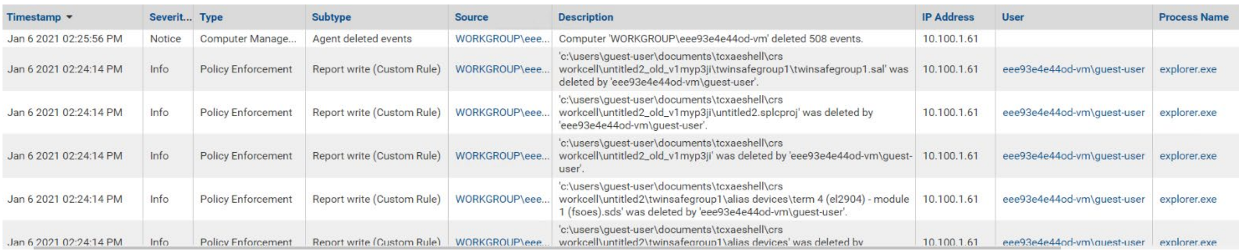
■ File Integrity Checking: WORMdisk

- Network file share on server is configured to use WORMdisk.

D.7.4.2 Test Results

The attempts to delete a file are detected by Carbon Black as shown in Figure D-75. Files stored on a storage drive protected by GreenTec are protected from deletion.

Figure D-75 Carbon Black Alerts Showing That a File Has Been Deleted



Timestamp	Severit...	Type	Subtype	Source	Description	IP Address	User	Process Name
Jan 6 2021 02:25:56 PM	Notice	Computer Manage...	Agent deleted events	WORKGROUP\eee...	Computer 'WORKGROUP\eee93e4e44od-vm' deleted 508 events.	10.100.1.61		
Jan 6 2021 02:24:14 PM	Info	Policy Enforcement	Report write (Custom Rule)	WORKGROUP\eee...	'c:\users\guest-user\documents\tcxaeshell\crs workcell\untitled2_old_v1myp3ji\twinsafegroup1\twinsafegroup1.sal' was deleted by 'eee93e4e44od-vm\guest-user'.	10.100.1.61	eee93e4e44od-vm\guest-user	explorer.exe
Jan 6 2021 02:24:14 PM	Info	Policy Enforcement	Report write (Custom Rule)	WORKGROUP\eee...	'c:\users\guest-user\documents\tcxaeshell\crs workcell\untitled2_old_v1myp3ji\untitled2.splcproj' was deleted by 'eee93e4e44od-vm\guest-user'.	10.100.1.61	eee93e4e44od-vm\guest-user	explorer.exe
Jan 6 2021 02:24:14 PM	Info	Policy Enforcement	Report write (Custom Rule)	WORKGROUP\eee...	'c:\users\guest-user\documents\tcxaeshell\crs workcell\untitled2_old_v1myp3ji' was deleted by 'eee93e4e44od-vm\guest-user'.	10.100.1.61	eee93e4e44od-vm\guest-user	explorer.exe
Jan 6 2021 02:24:14 PM	Info	Policy Enforcement	Report write (Custom Rule)	WORKGROUP\eee...	'c:\users\guest-user\documents\tcxaeshell\crs workcell\untitled2\twinsafegroup1\alias devices\term 4 (el2904) - module 1 (fsoes).sds' was deleted by 'eee93e4e44od-vm\guest-user'.	10.100.1.61	eee93e4e44od-vm\guest-user	explorer.exe
Jan 6 2021 02:24:14 PM	Info	Policy Enforcement	Report write (Custom Rule)	WORKGROUP\eee...	'c:\users\guest-user\documents\tcxaeshell\crs workcell\untitled2\twinsafegroup1\alias devices' was deleted by	10.100.1.61	eee93e4e44od-vm\guest-user	explorer.exe

D.8 Executing Scenario 8: Detect Unauthorized Modification of PLC Logic

An authorized user performs an unapproved or unauthorized modification of the PLC logic through the secure remote access tools. The expected result is the behavioral anomaly detection tools will detect and capture the activity, flagging it for review.

The behavior anomaly detection tools can detect program downloads to the PLC. Program download detection needs to be correlated with the maintenance management system to determine if the download was authorized and approved. This was not demonstrated as part of this scenario.

D.8.1 Build 1

D.8.1.1 Configuration

■ Behavior Anomaly Detection: Tenable.ot

- Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

■ Remote Access: Cisco VPN

- Configured to allow authorized VPN users to access to ConsoleWorks web interface.

■ User Authentication/User Authorization: ConsoleWorks

- Configured for accessing the PCS environment

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

D.8.1.2 Test Results

In this build, a remote session Studio 5000 Logix Designer is established to perform PLC file operations as shown in Figure D-76 and Figure D-77. Tenable.ot is able to detect the PLC file modifications as shown in Figure D-78 with details shown in Figure D-79 and Figure D-80.

Figure D-76 Remote Access to Systems in PCS Network is Established Through ConsoleWorks

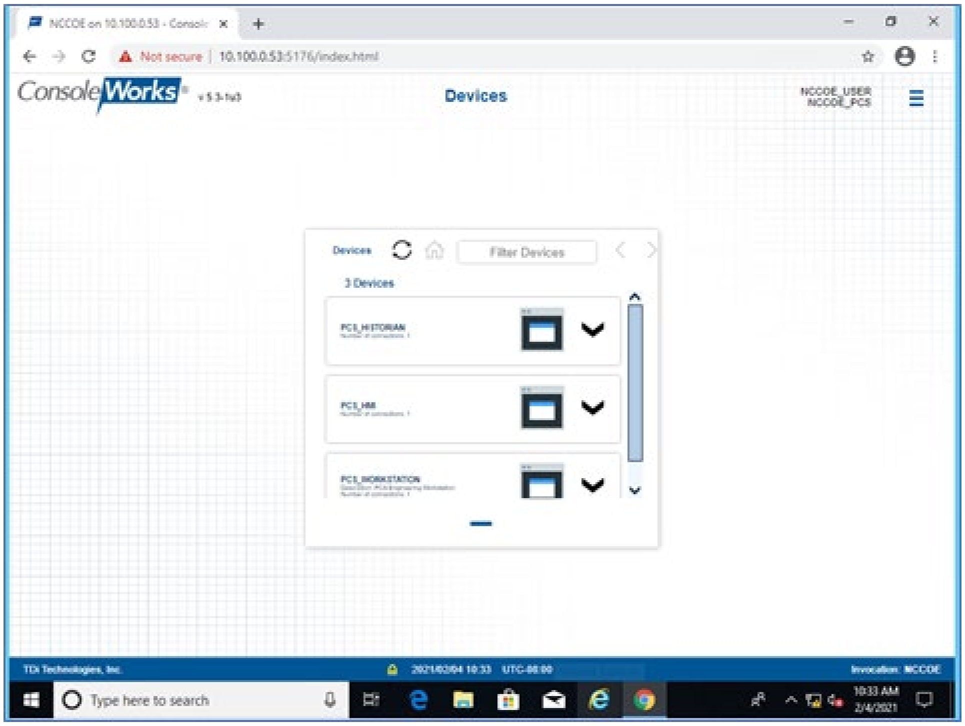


Figure D-77 Remote Session into Studio 5000 to Perform PLC File Operations

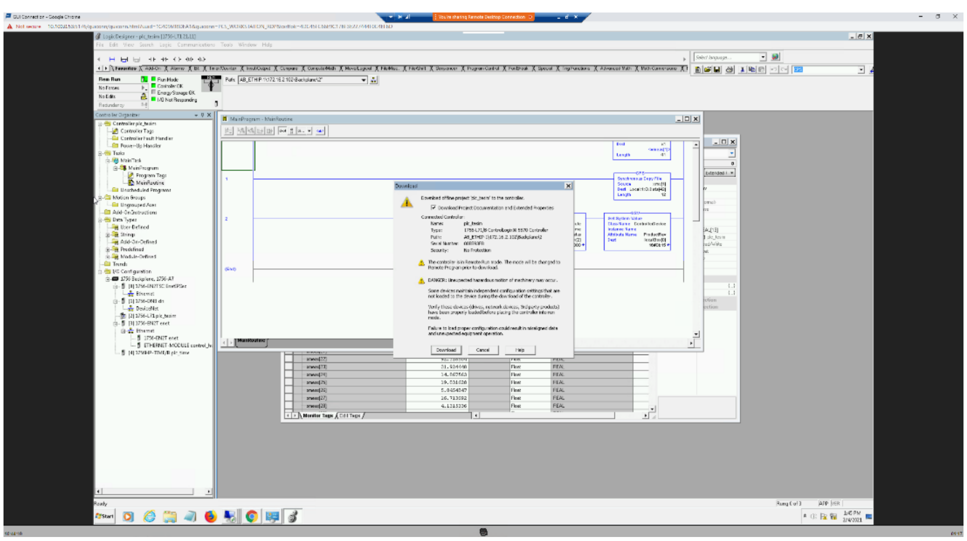


Figure D-78 Tenable.ot Detected the Transfer of PLC Logic File to the Rockwell PLC

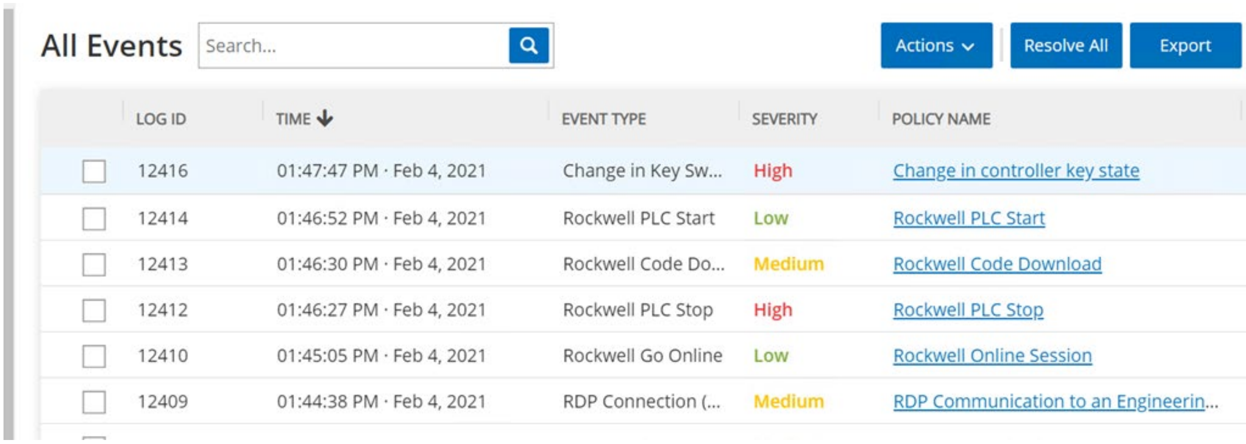


Figure D-79 Tenable.ot PLC Stop alert details

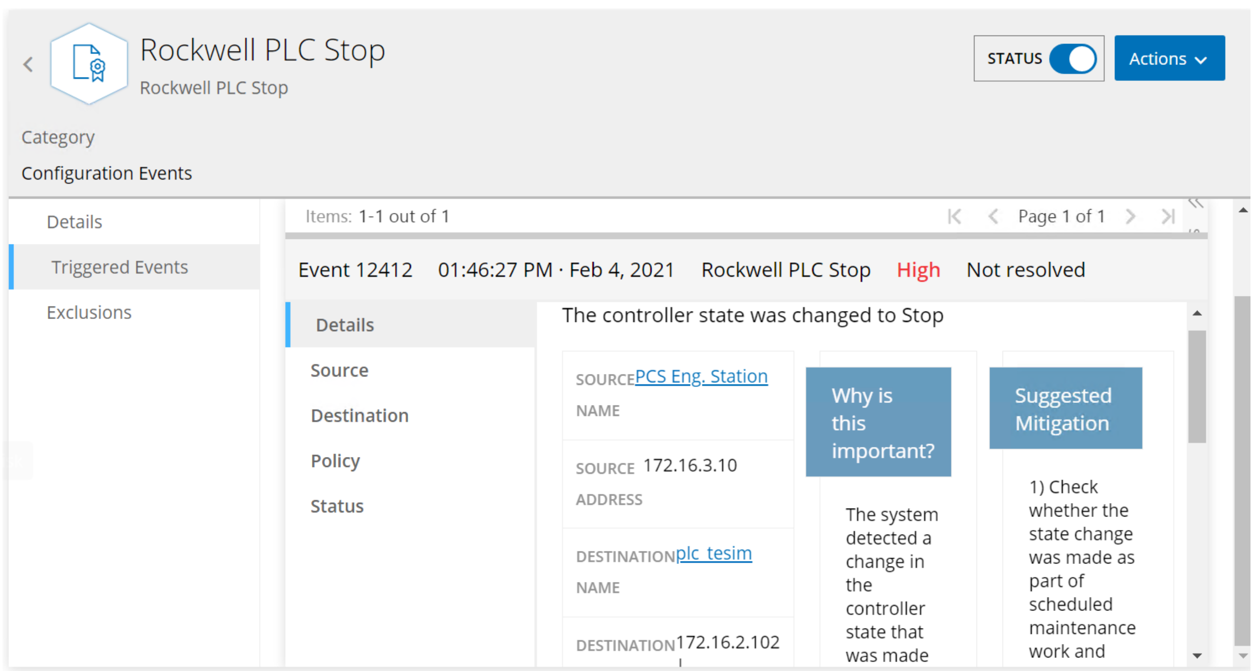
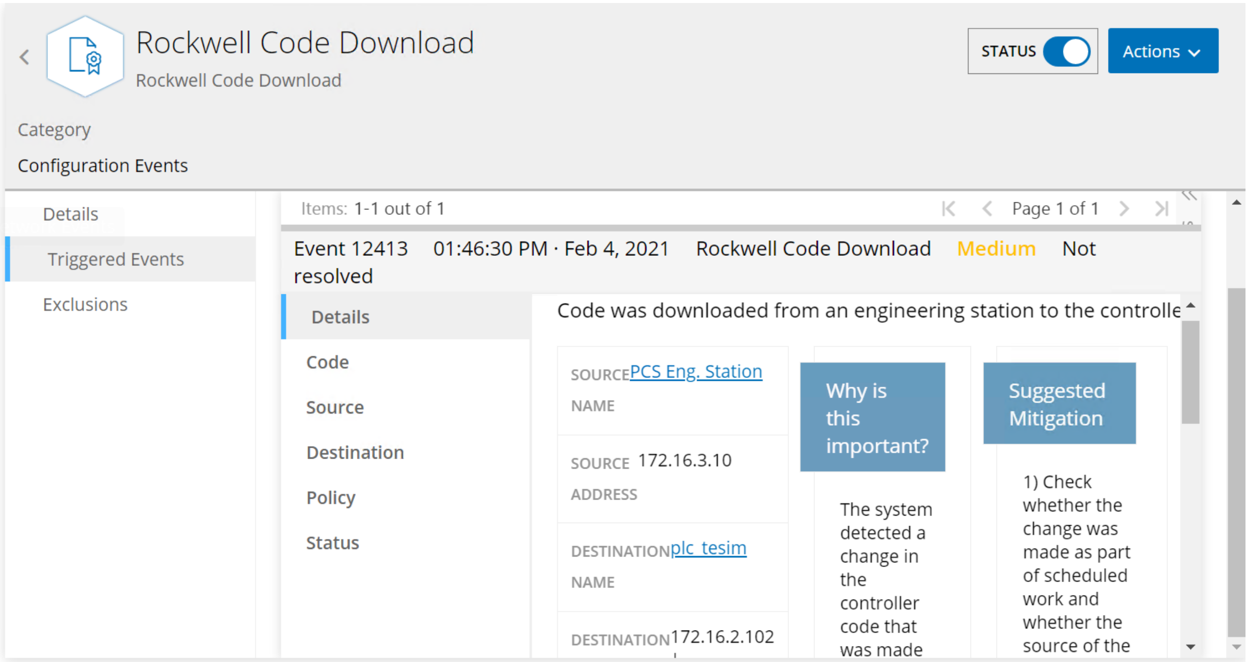


Figure D-80 Tenable.ot PLC Program Download Alert Details



D.8.2 Build 2

D.8.2.1 Configuration

- Behavior Anomaly Detection: eyeInspect
    - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.
- Remote Access, User Authentication/User Authorization: Dispel
    - Dispel VDI is configured to allow authorized users to access PCS environment through the Dispel Enclave to the Dispel Wicket.

D.8.2.2 Test Results

As shown in Figure D-81 the authorized user establishes a session into the manufacturing environment using the Dispel VDI. The user connects to the engineering workstation and launches the Studio 5000 Logix Designer as shown in Figure D-82 to modify the PLC logic. Figure D-83, Figure D-84 and Figure D-85 show that Forescout is able to detect the traffic between the engineering workstation and the PLC, including details of the Stop command and Download command.

Figure D-81 Remote Access to Systems in PCS Network is Being Established Through Dispel

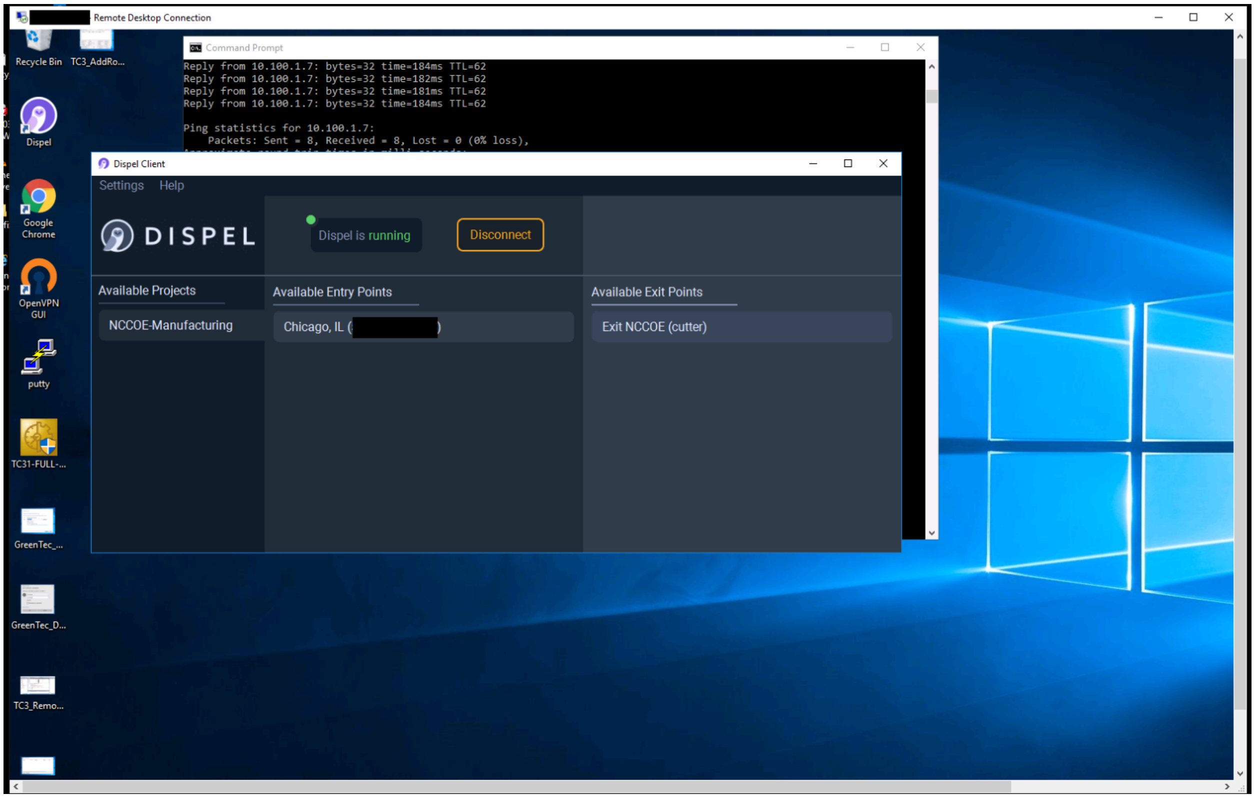


Figure D-82 Modifying the Parameters for the Allen-Bradley PLC Controller Using Studio 5000

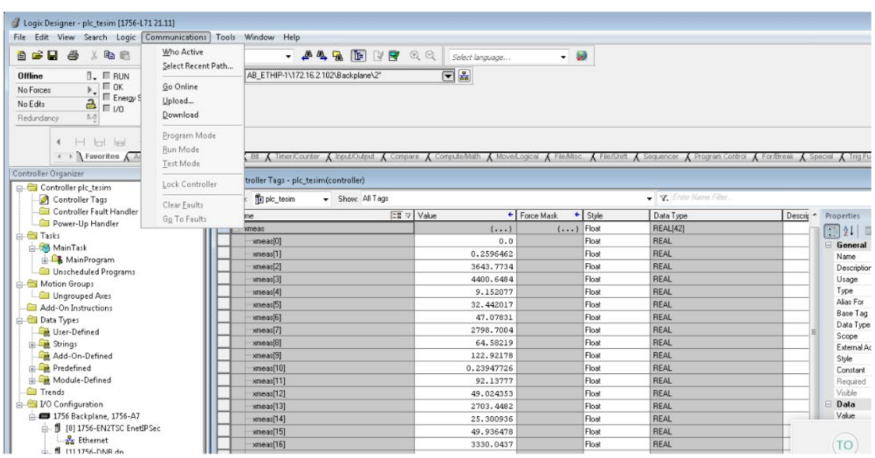


Figure D-83 Forescout Alerts Showing It Detected the Traffic Between the Engineering Workstation and the PLC

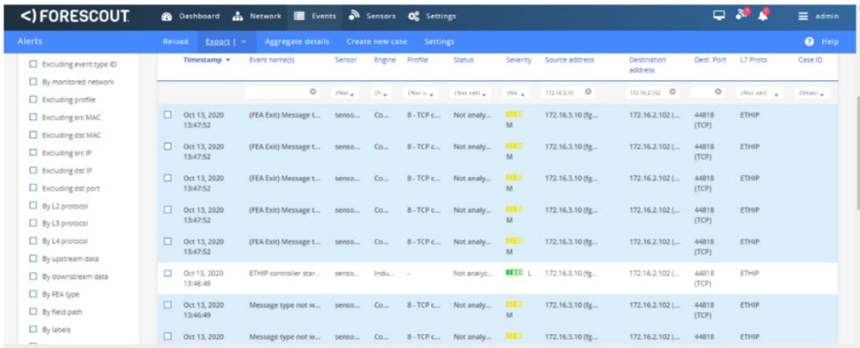


Figure D-84 Forescout Alert Details for the Stop Command Issued to the PLC

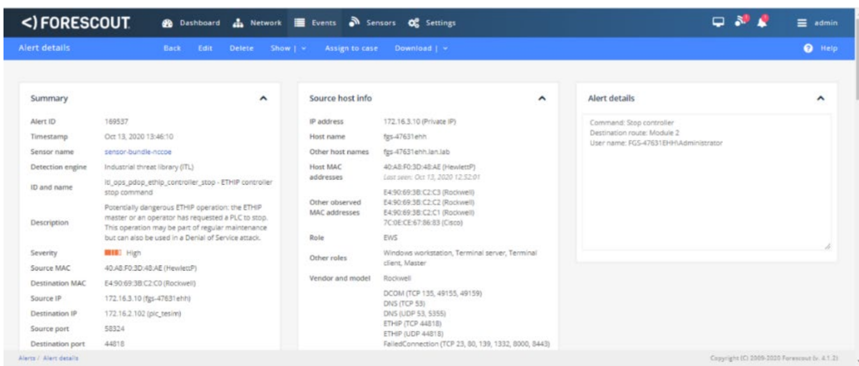
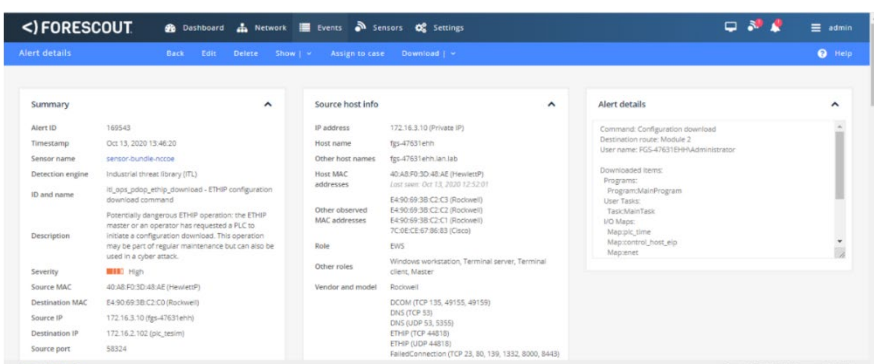


Figure D-85 Forescout Alert Details for the Configuration Download Command



D.8.3 Build 3

D.8.3.1 Configuration

- Behavior Anomaly Detection: Dragos
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.
- Remote Access: Cisco VPN
  - Configured to allow authorized VPN users to access to ConsoleWorks web interface.
- User Authentication/User Authorization: ConsoleWorks
  - Configured for accessing the CRS environment.

D.8.3.2 Test Results

In this build, a remote session to the CRS workstation is established to perform PLC file operations as shown in Figure D-86 and Figure D-87. Dragos is able to detect the PLC file modifications as shown in Figure D-88 with details shown in Figure D-89.

Figure D-86 VPN Connection to the Manufacturing Environment

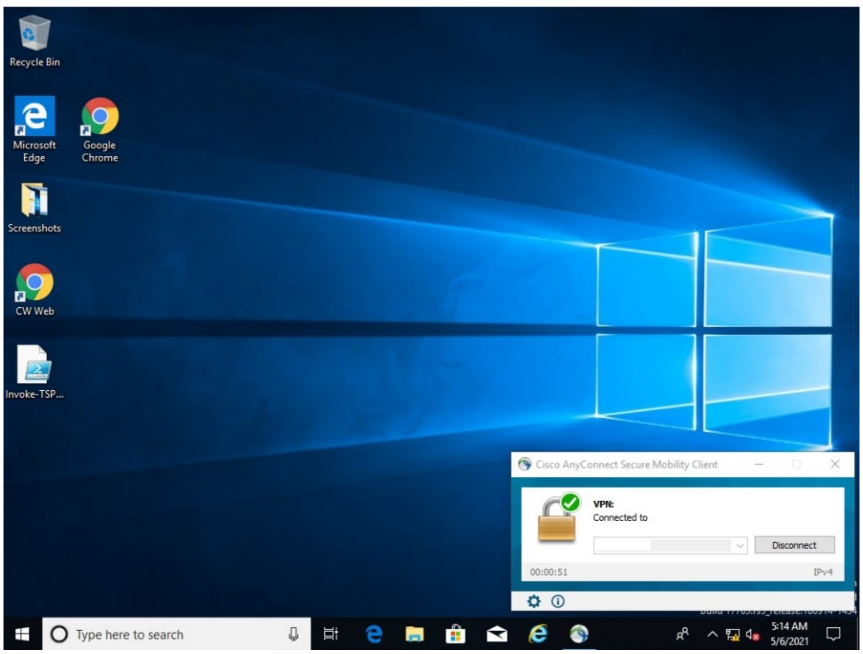


Figure D-87 Remote Access is Being Established through ConsoleWorks

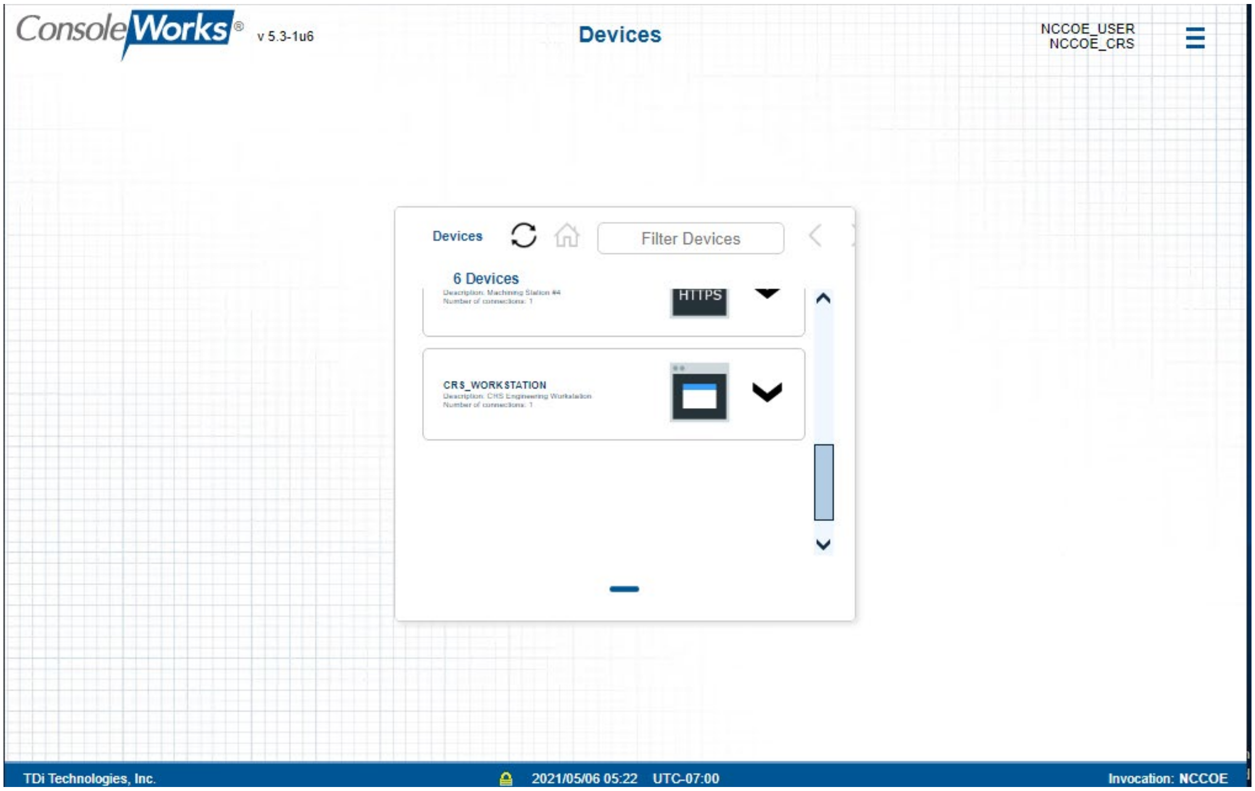


Figure D-88 Dragos Notification Manager Showing Detection of the Transfer of PLC Logic File to the Beckhoff PLC

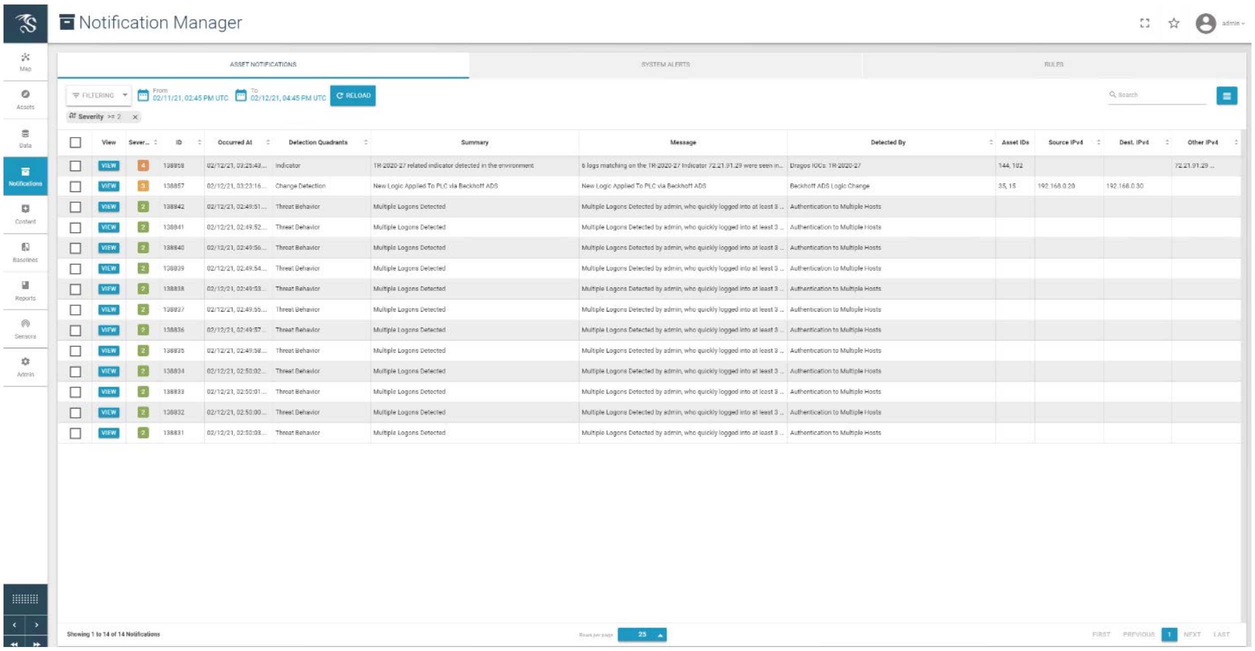
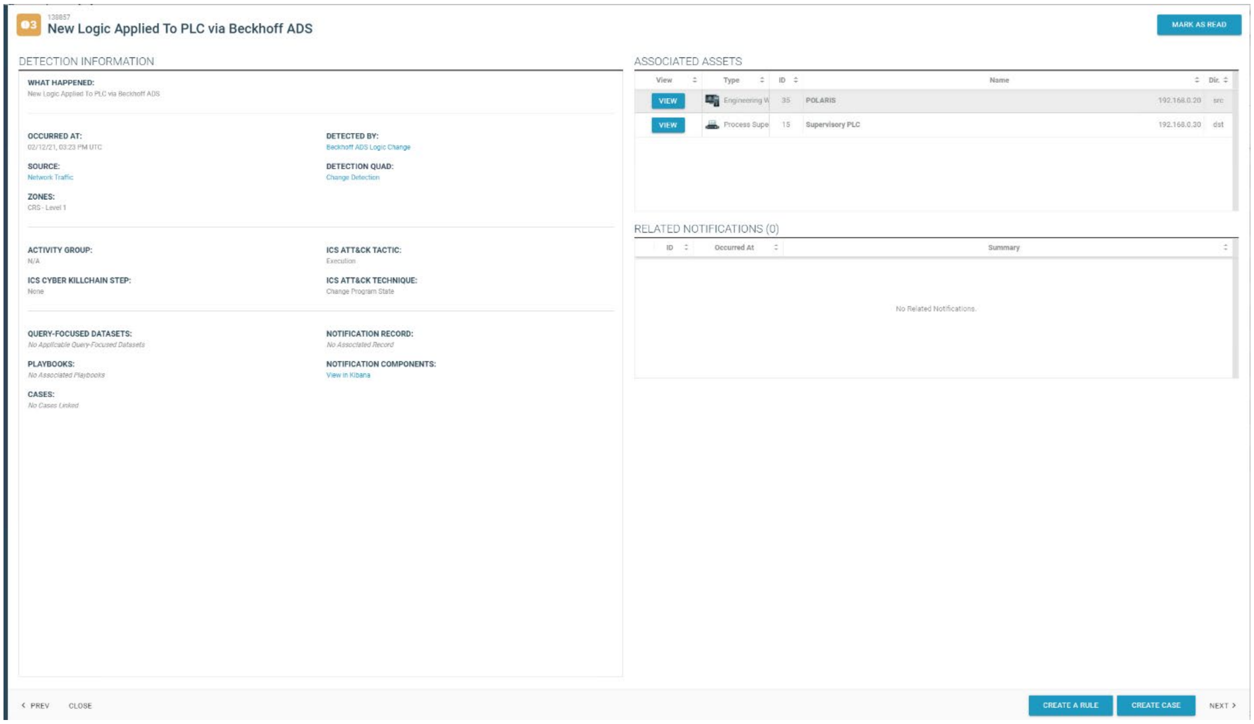


Figure D-89 Dragos Alert Details for the PLC Logic File Download



D.8.4 Build 4

D.8.4.1 Configuration

- Behavior Anomaly Detection: Azure Defender for IoT
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.
- Remote Access, User Authentication/User Authorization: Dispel
  - Dispel VDI is configured to allow authorized users to access the PCS environment through the Dispel Enclave to the Dispel Wicket.

D.8.4.2 Test Results

Figure D-90 and Figure D-91 show the connection to the CRS environment through the Dispel VDI. The changes to the PLC programs are detected by Azure Defender for IoT, as shown in Figure D-92, because the Dispel VDI is not an authorized programming device.

Figure D-90 Dispel VDI with Interface for Connecting Through Dispel Enclave to Dispel Wicket

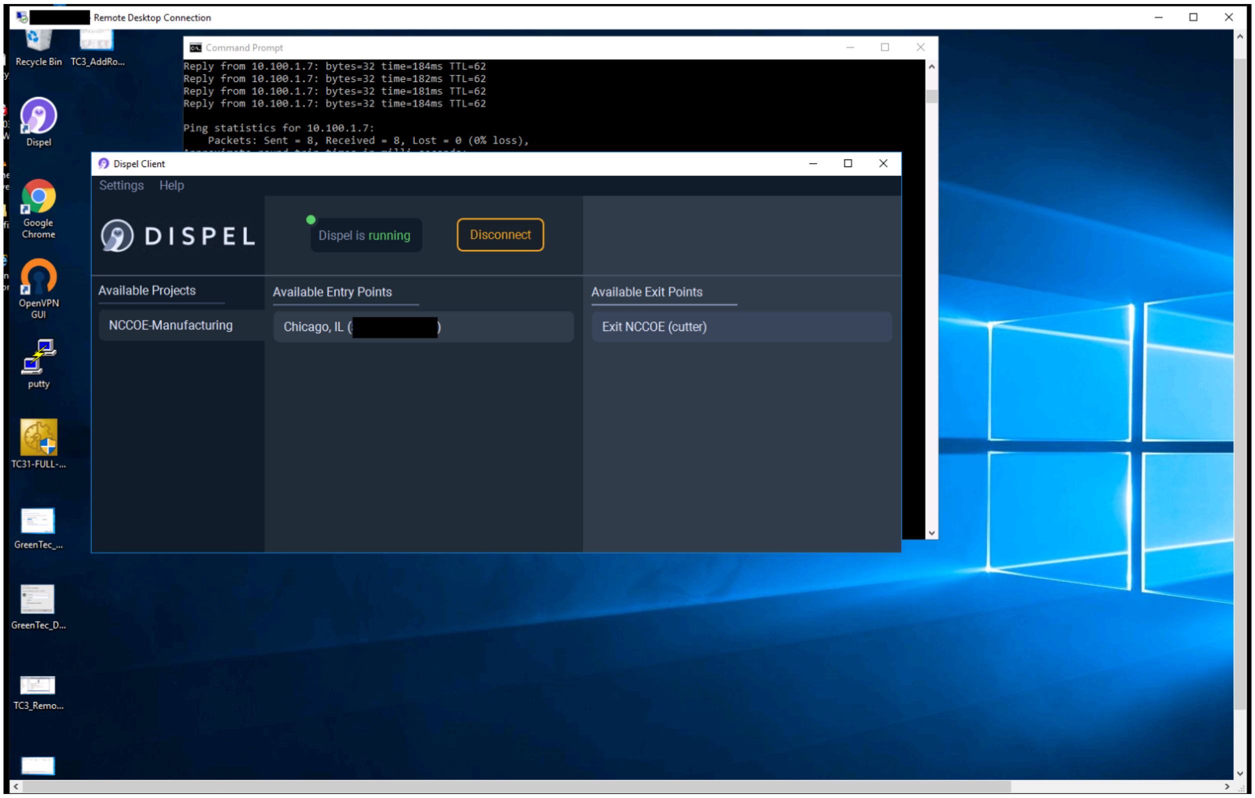


Figure D-91 Nested RDP Connections Showing Dispel Connection into the CRS Workstation

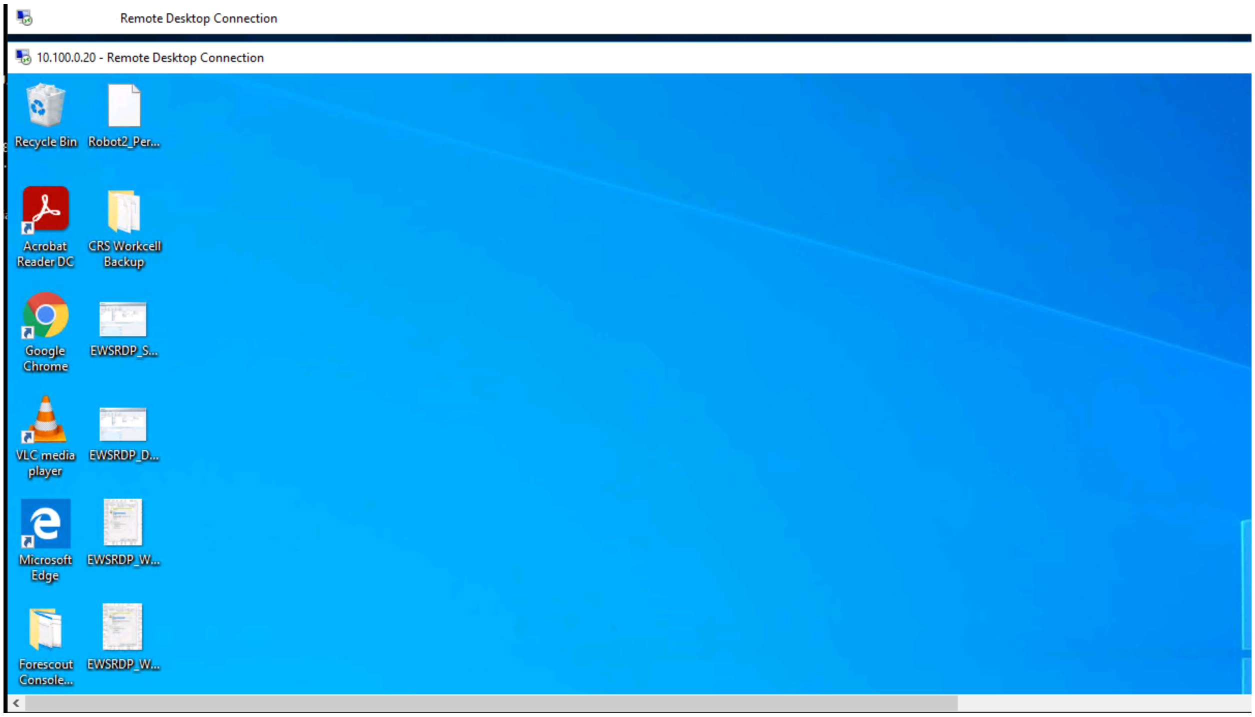
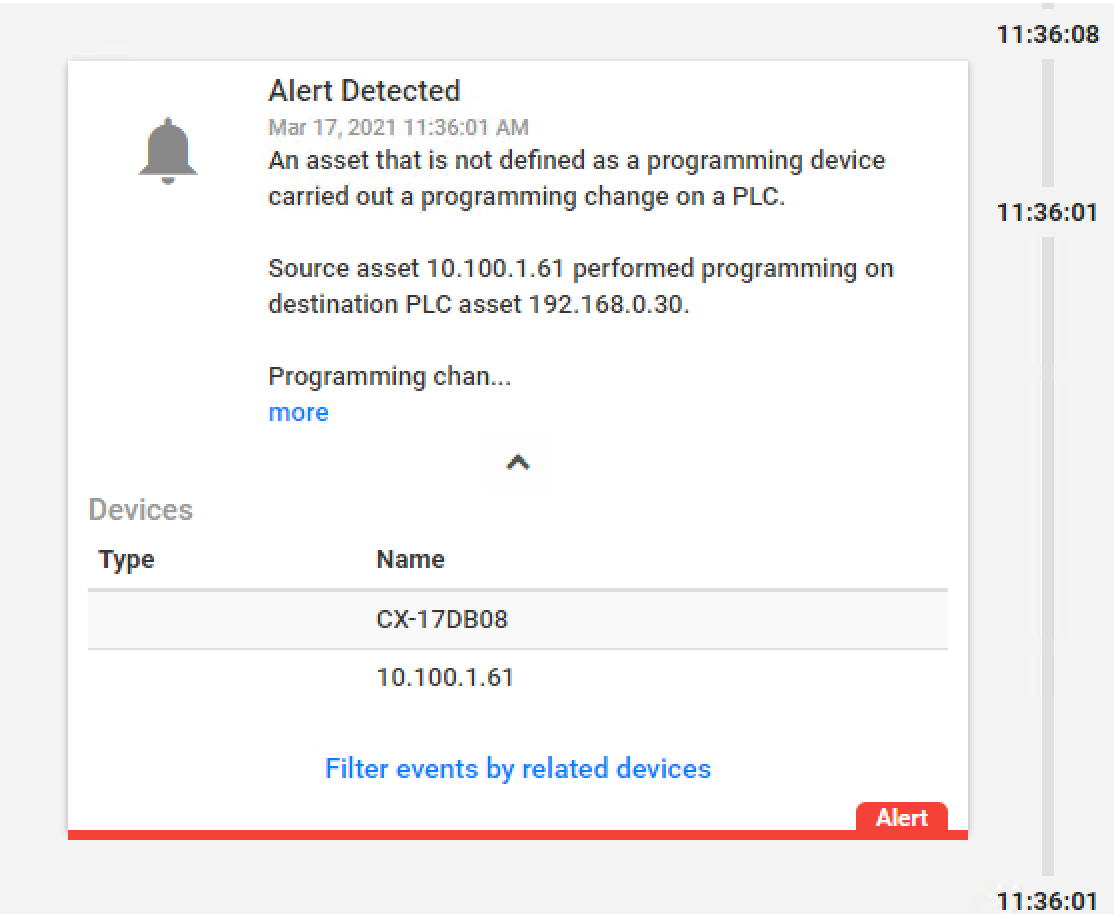


Figure D-92 Azure Defender for IoT Alert for Unauthorized PLC Programming



D.9 Executing Scenario 9: Protect from Modification of Historian Data

An attacker who has already gained access to the corporate network attempts to modify historian archive data located in the DMZ. The expected result is the behavioral anomaly detection products detect the connection to the historian archive. File modification is prevented by the file integrity checking capability.

D.9.1 Build 1

D.9.1.1 Configuration

- Behavior Anomaly Detection: Tenable.ot

- Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.
- File Integrity Checking: ForceField
  - PI Server is configured to use ForceField drive.

D.9.1.2 Test Results

Figure D-93 shows Tenable.ot detecting the remote access connections. Figure D-94 shows that GreenTec successfully blocks the attacker from deleting archive data.

Figure D-93 Tenable.ot alert Shows SMB Connection from External Workstation to the Historian

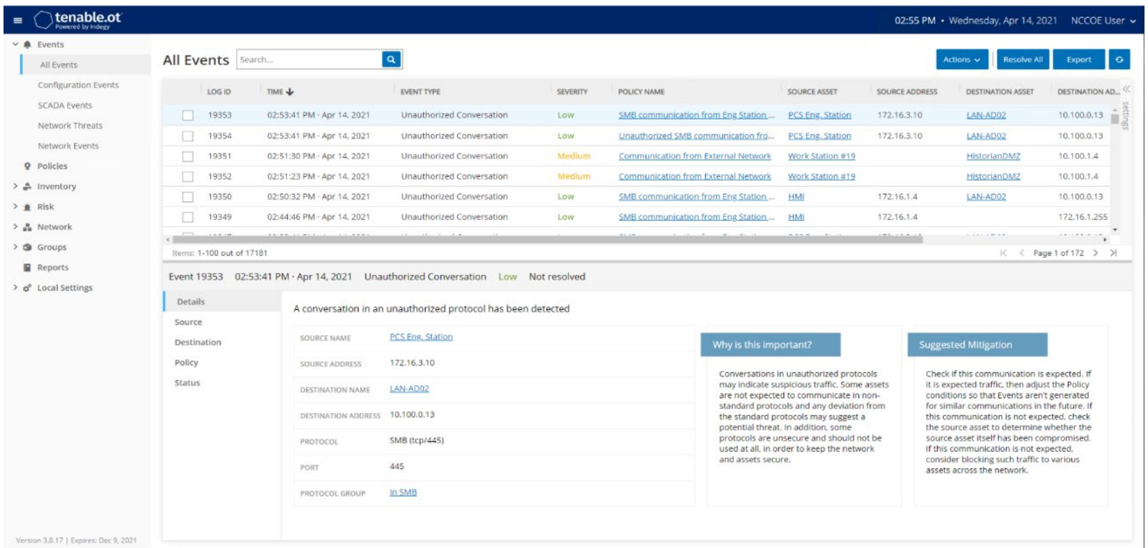
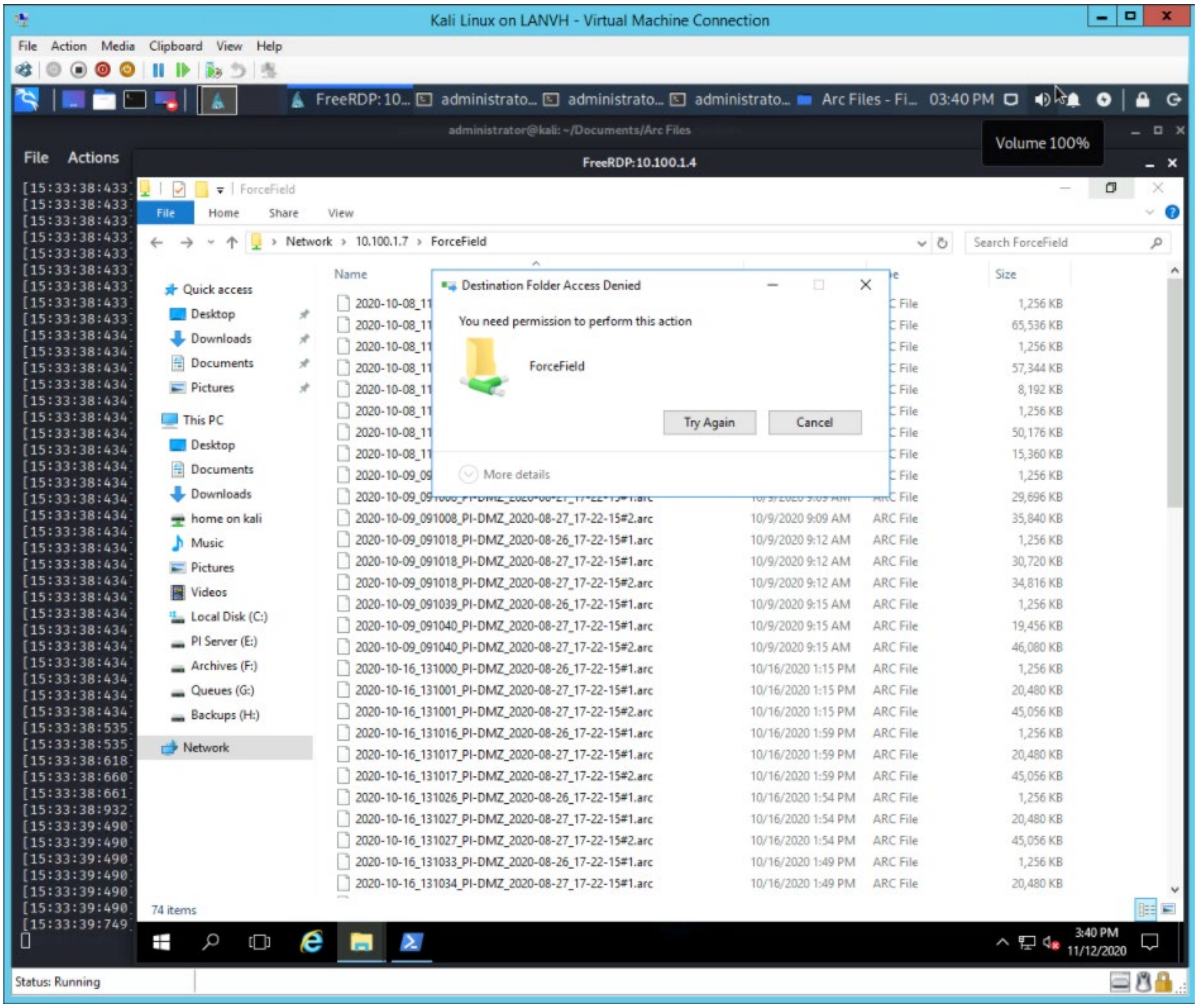


Figure D-94 GreenTec Denies Modification and Deletion File Operations in the Protected Drive



D.9.2 Build 2

D.9.2.1 Configuration

- Behavior Anomaly Detection: eyeInspect
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.
- File Integrity Checking: ForceField
  - PI Server is configured to use ForceField drive.

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

D.9.2.2 Test Results

Forescout detects the remote session as shown in Figure D-95. When the user attempts to alter a file on the protected drive, GreenTec denies the operation as shown in Figure D-96.

Figure D-95 Forescout Alert Shows Network Connection from Corporate Network to the Historian

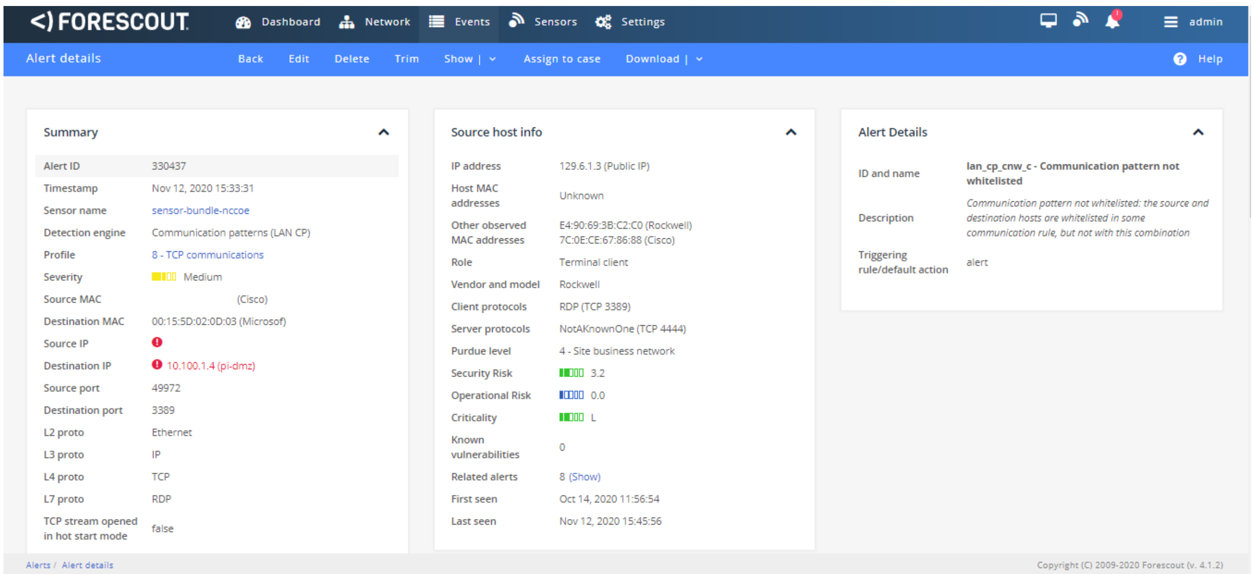
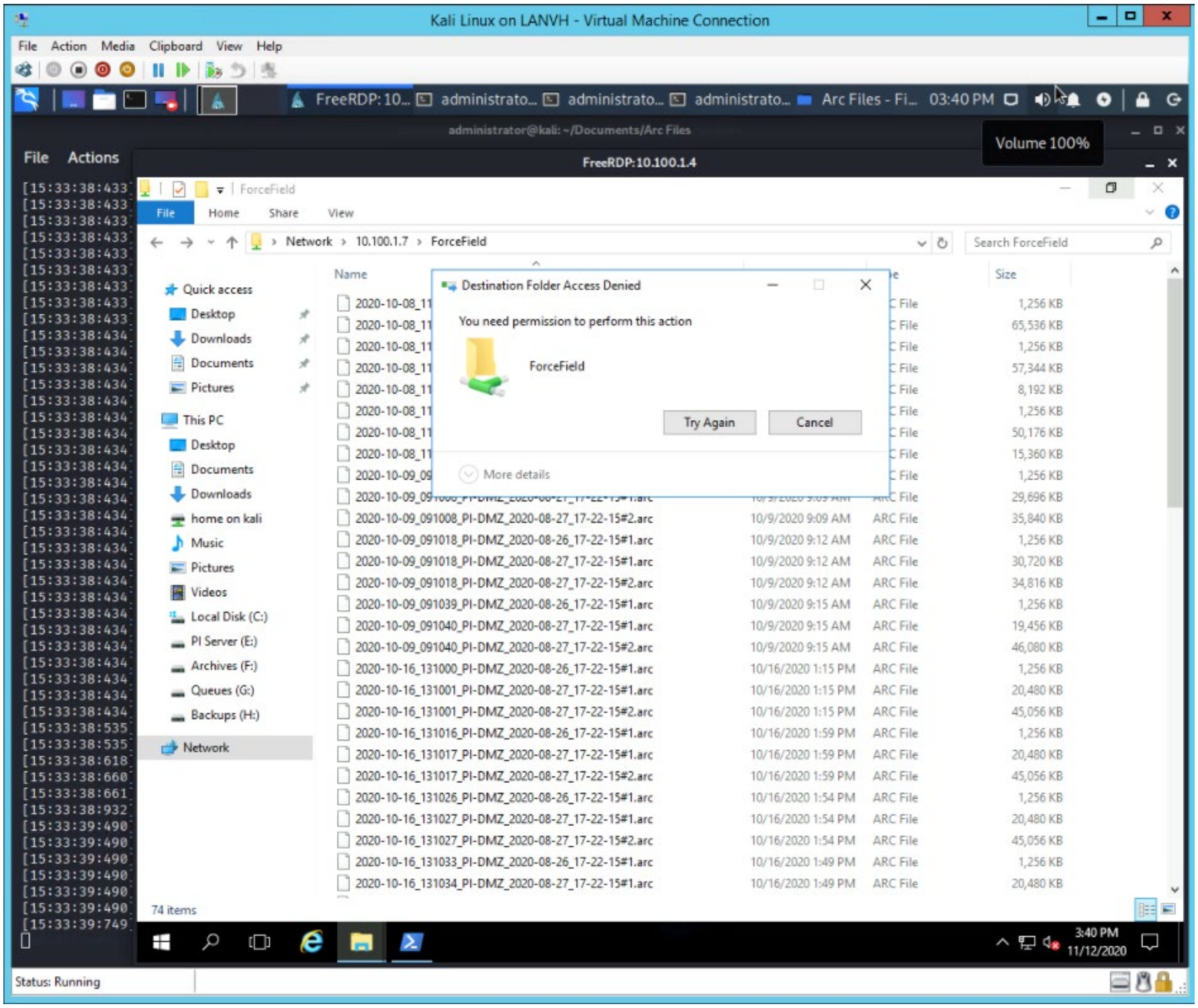


Figure D-96 GreenTec Denies Modification and Deletion File Operations in the Protected Drive



D.9.3 Build 3

D.9.3.1 Configuration

- Behavior Anomaly Detection: Dragos
    - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.
- File Integrity Checking: ForceField
    - PI Server is configured to use ForceField drive.

D.9.3.2 Test Results

Dragos detects the remote session as shown in Figure D-97. When the user attempts to alter a file on the protected drive, GreenTec denies the operation as shown in Figure D-98.

Figure D-97 Dragos Detection of RDP Session from an External Network to the Historian

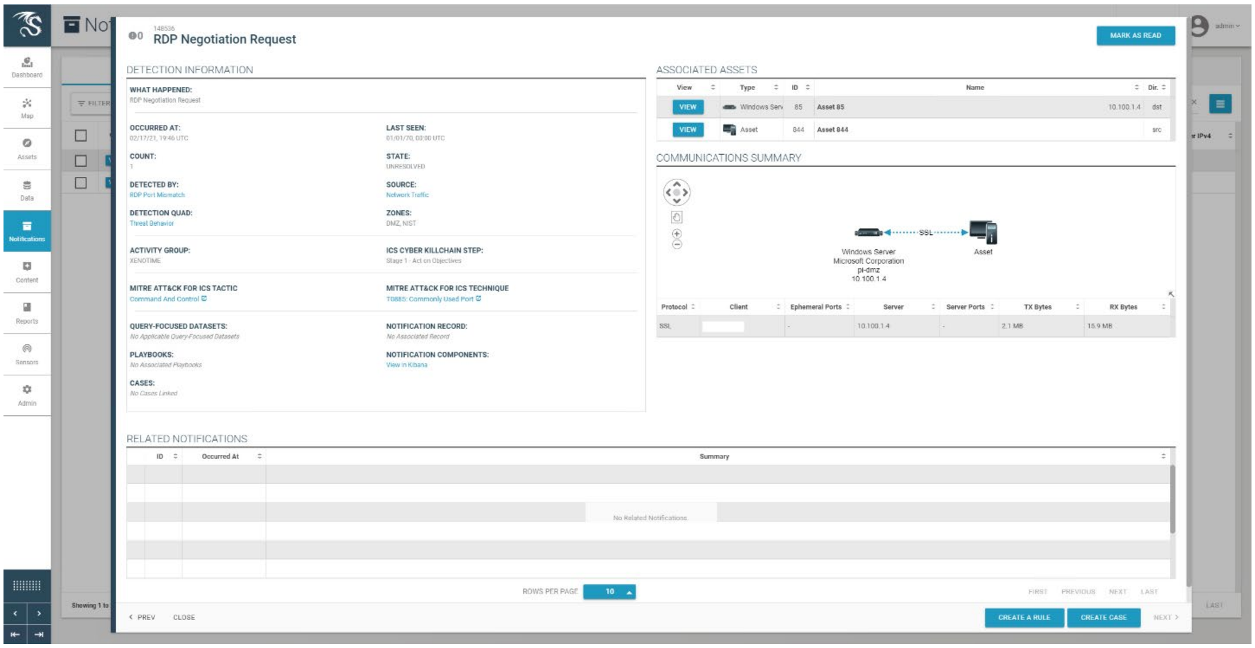
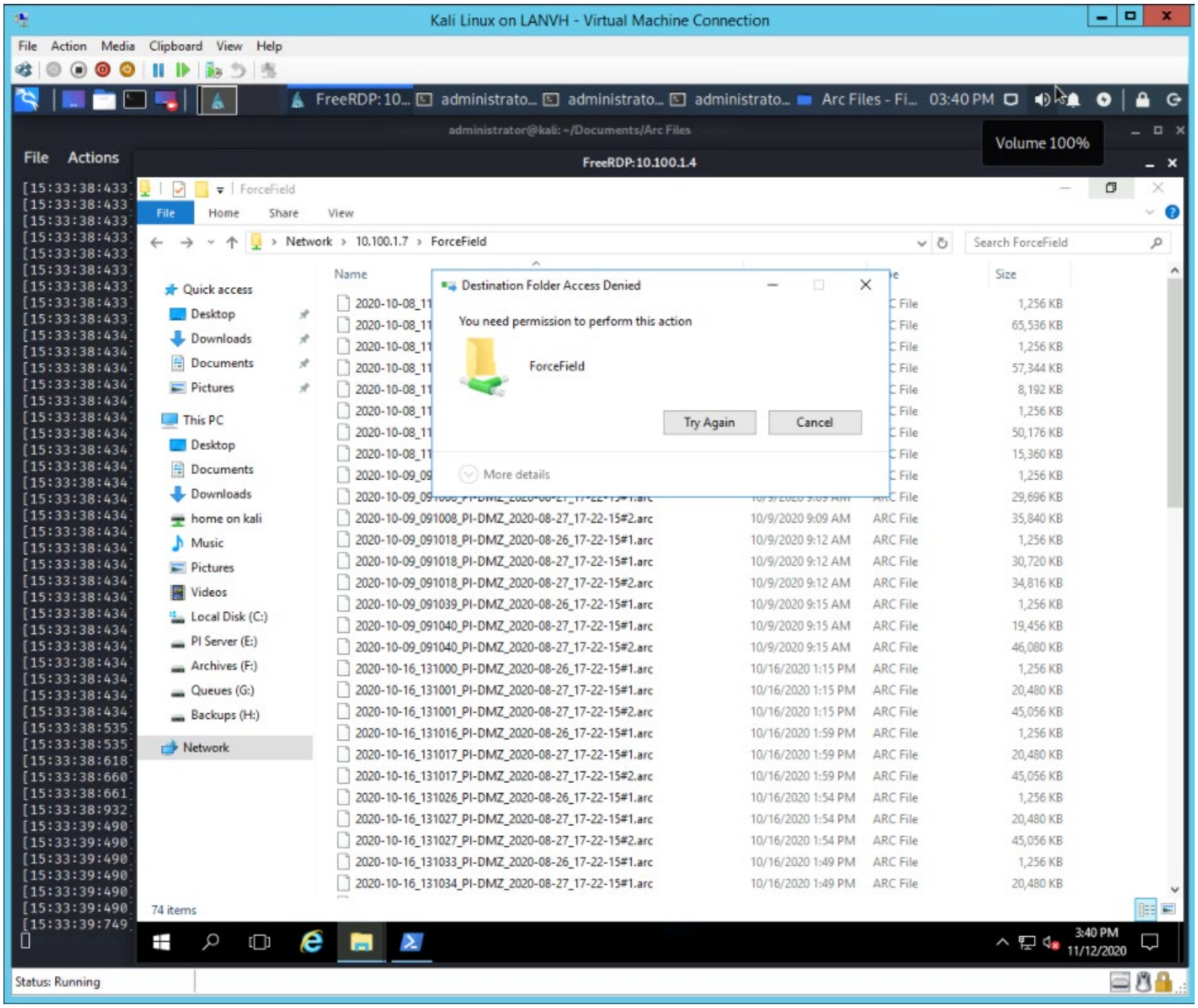


Figure D-98 GreenTec Denies Modification and Deletion File Operations in the Protected Drive



D.9.4 Build 4

D.9.4.1 Configuration

- Behavior Anomaly Detection: Azure Defender for IoT
  - Configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN.
- File Integrity Checking: ForceField
  - PI Server is configured to use ForceField drive.

D.9.4.2 Test Results

The connection to the Historian data storage was detected by Azure Defender for IoT as shown in Figure D-99. Figure D-100 shows a Windows error message after attempting to overwrite protected Historian files.

Figure D-99 Azure Defender for IoT Event Timeline Showing the Remote Access Connection to the Historian

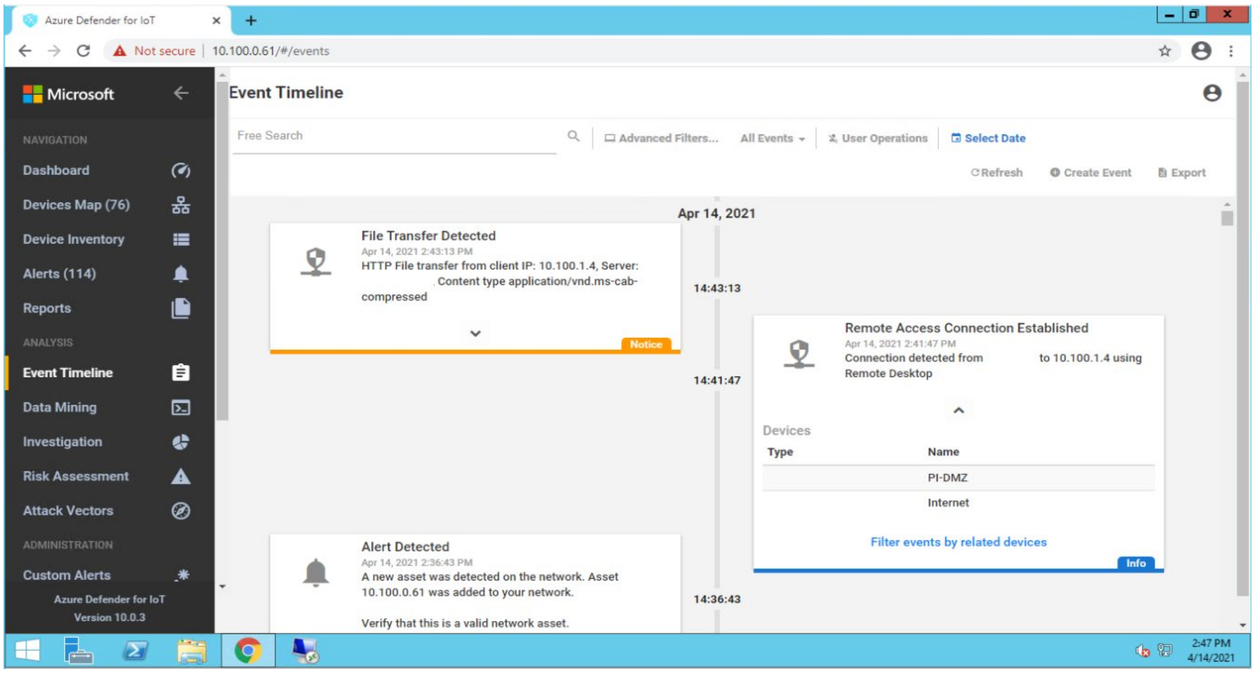
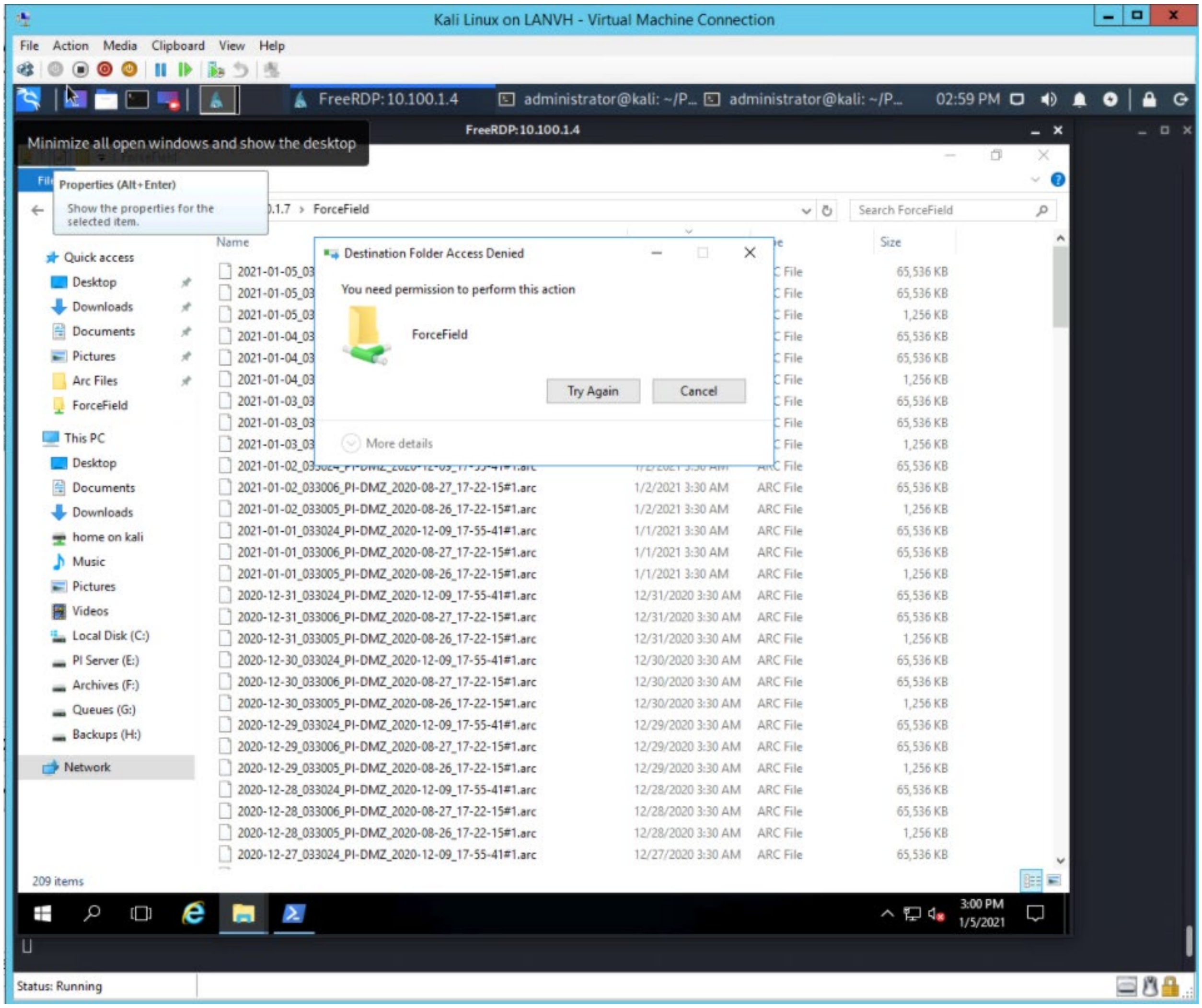


Figure D-100 GreenTec Denies Modification and Deletion File Operations in the Protected Drive



D.10 Executing Scenario 10: Detect Sensor Data Manipulation

A sensor in the manufacturing system sends out-of-range data values to the Historian. The expected result is the behavioral anomaly detection (data historian) capability alerts on out-of-range data.

D.10.1 All Builds

D.10.1.1 Configuration

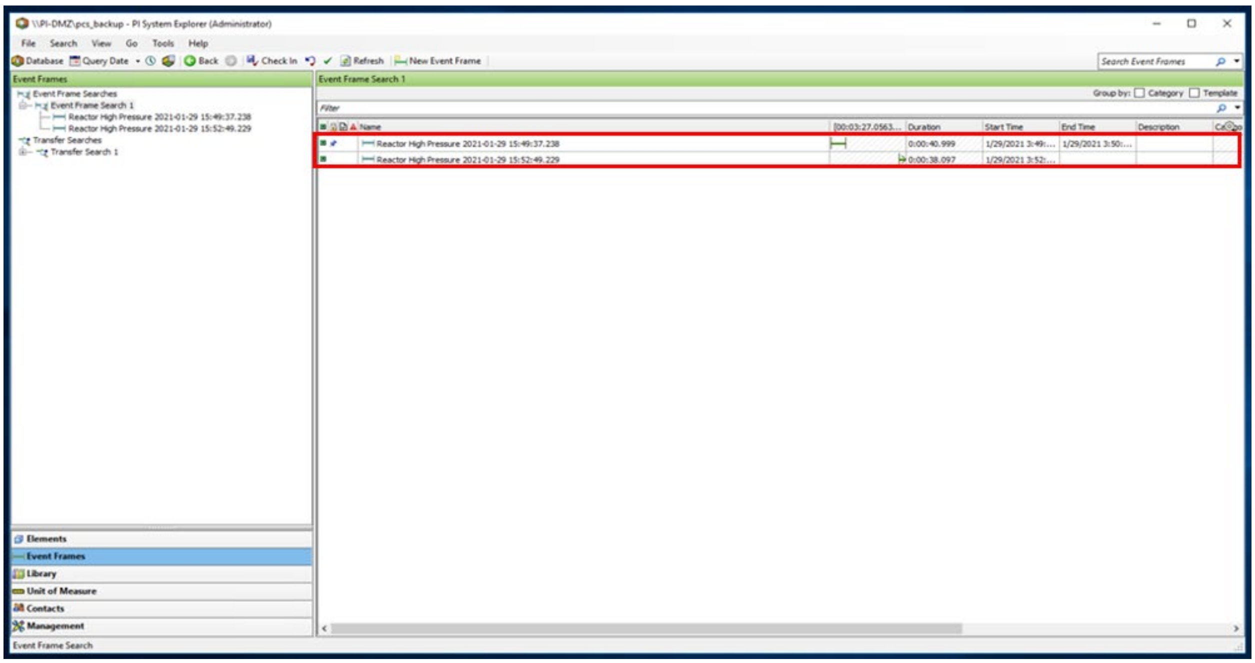
- Behavior Anomaly Detection: PI Server
    - Configured to receive process data from across the manufacturing system.

- Configured to perform analysis on incoming data points.

D.10.1.2 Test Results

The Historian process monitoring capabilities provided by the PI System are able to monitor out-of-range sensor readings and generate alerts. Figure D-101 shows the PI Server's event frame alerts on the out-of-range reactor pressure readings in the PCS.

Figure D-101 PI Server's Event Frames Showing Out-of-Range Sensor Readings for the Reactor Pressure



D.11 Executing Scenario 11: Detect Unauthorized Firmware Modification

An authorized user accesses the system remotely and performs an unauthorized firmware change on a PLC. The expected result is the behavioral anomaly detection tools will alert on the new firmware.

The behavior anomaly detection tools can detect changes to the firmware. Firmware change detection needs to be correlated with the maintenance management system to determine if the firmware change was authorized and approved. This was not demonstrated as part of this scenario.

D.11.1 Build 1

D.11.1.1 Configuration

- Behavior Anomaly Detection: Tenable.ot

- Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2.

- Remote Access: Cisco VPN
  - Configured to allow authorized VPN users access to ConsoleWorks web interface.
- User Authentication/User Authorization: ConsoleWorks
  - Configured for accessing the PCS environment.

D.11.1.2 Test Results

Figure D-102 depicts the list of the events detected by Tenable.ot resulting from the firmware change. The details of one of the alerts are shown in Figure D-103.

Figure D-102 Tenable.ot Detects a Collection of Events Generated by a Firmware Change

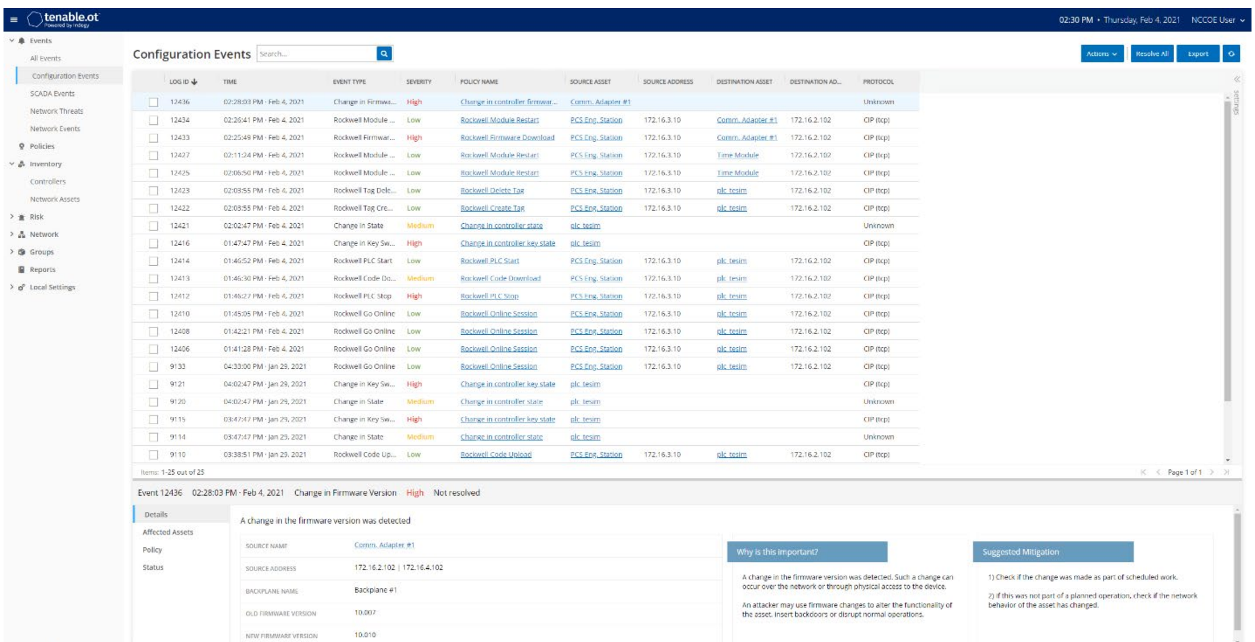
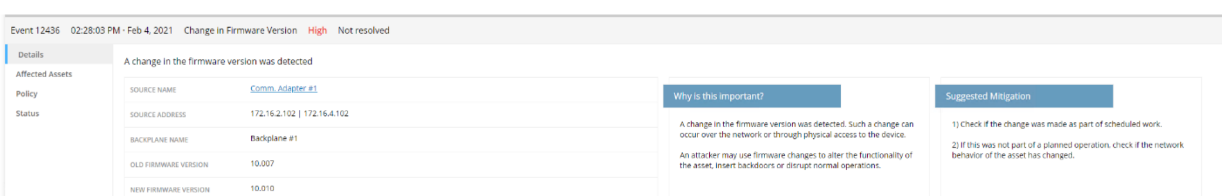


Figure D-103 Details for One of the Alerts Showing the Firmware Change



D.11.2 Build 2

D.11.2.1 Configuration

- Behavior Anomaly Detection: eyeInspect
  - Configured to receive packet streams from DMZ, Testbed LAN, and PCS VLAN 1 and 2
- Remote Access, User Authentication/User Authorization: Dispel
  - Dispel VDI is configured to allow authorized users to access the PCS environment through the Dispel Enclave to the Dispel Wicket.

D.11.2.2 Test Results

Figure D-104 shows the activities detected by Forescout as a result of firmware change. Figure D-104, Figure D-105 and Figure D-106 show more details on the alerts associated with the firmware update.

Figure D-104 Forescout Detects a Collection of Alerts Associated with the Firmware Change

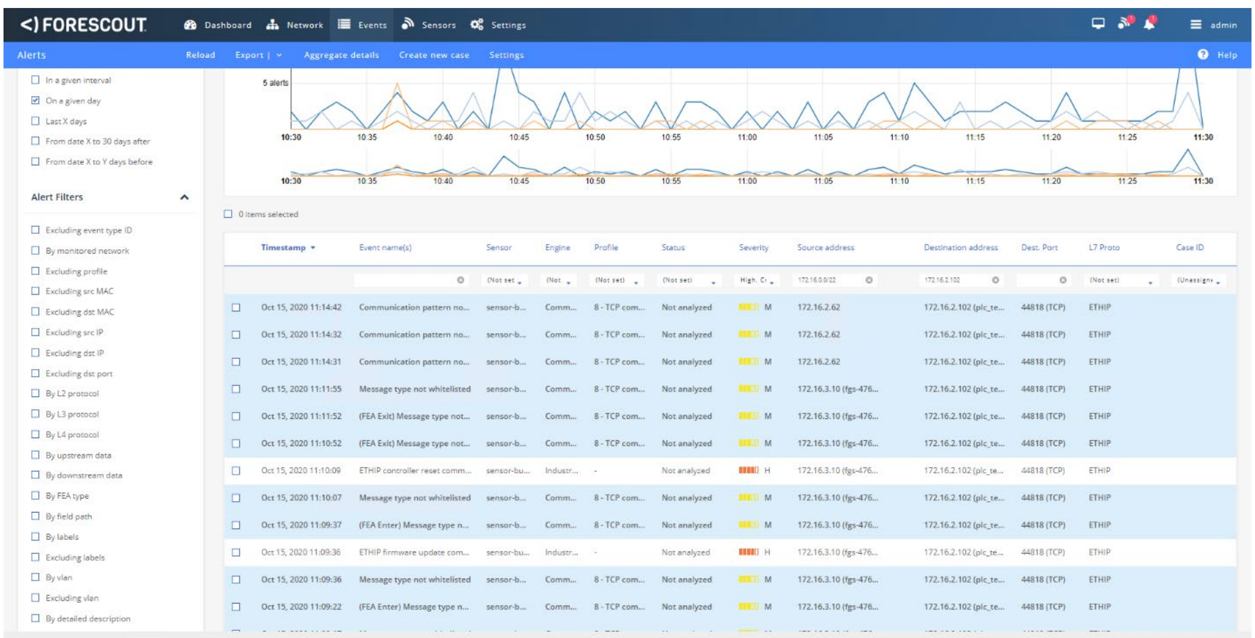


Figure D-105 Alert Details Detected by Forescout for the Firmware Change

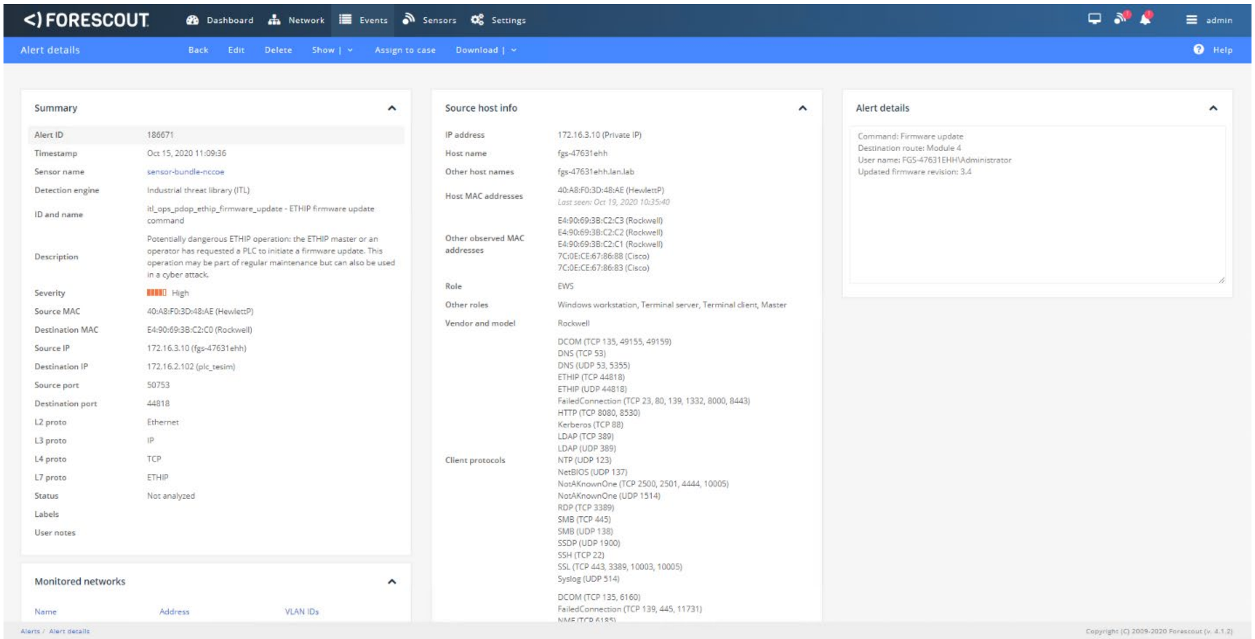
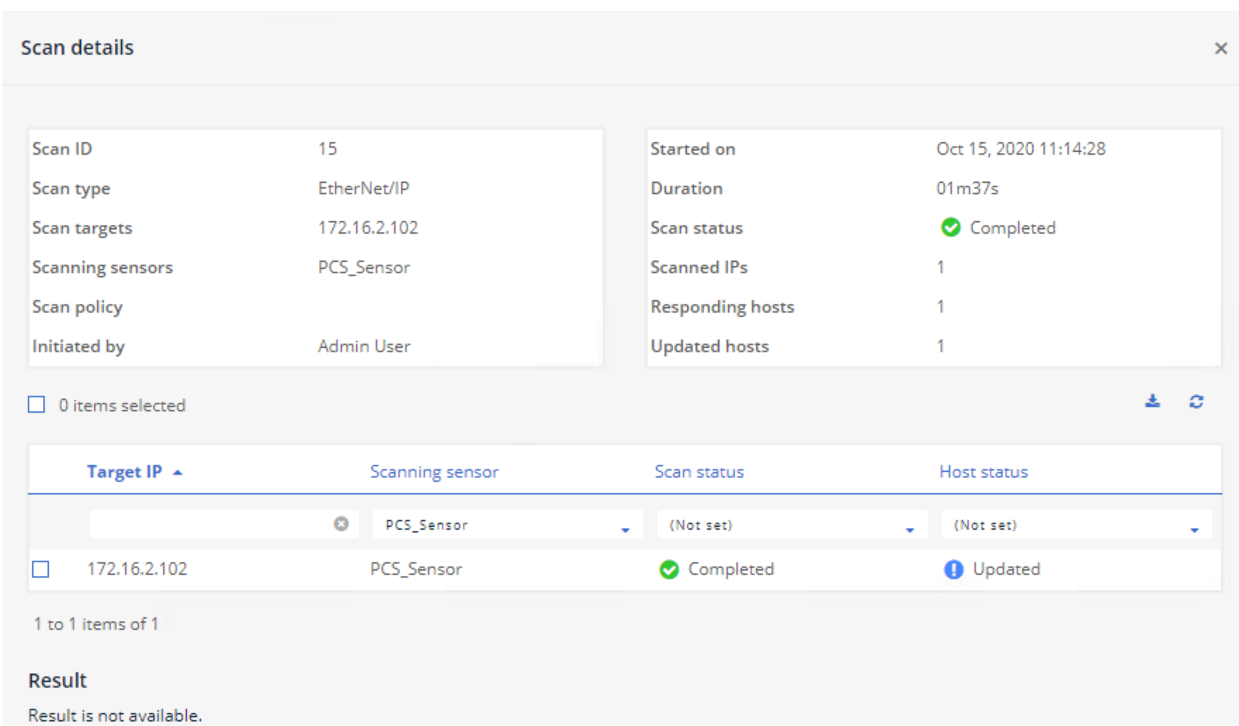


Figure D-106 ICS Patrol Scan Results Showing a Change Configuration was Made



D.11.3 Build 3

D.11.3.1 Configuration

- Remote Access: Cisco VPN
    - configured to allow authorized VPN users to access only the ConsoleWorks web interface
- User Authentication/User Authorization: ConsoleWorks
    - configured to allow remote access to hosts in manufacturing environment
- Behavior Anomaly Detection: Dragos
    - configured to receive packet streams from DMZ, Testbed LAN, Supervisory LAN, and Control LAN

D.11.3.2 Test Results

Dragos detects the change to the firmware as shown on the dashboard in Figure D-107with details shown in Figure D-108.

Figure D-107 Dragos Dashboard Showing an Alert for Firmware Change

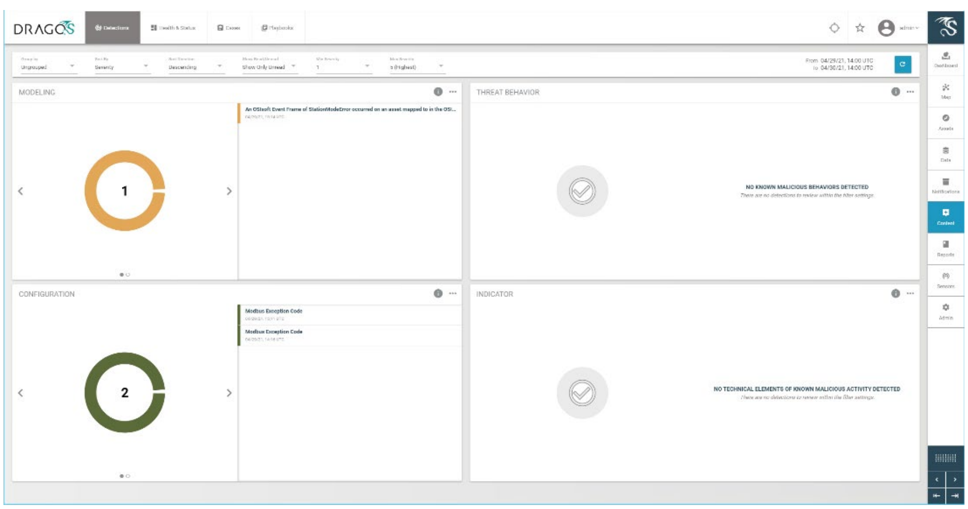
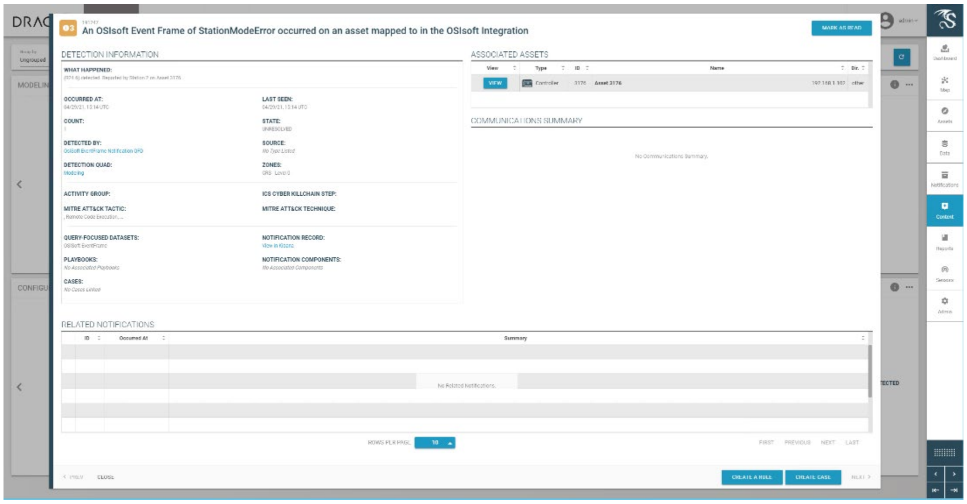


Figure D-108 Details for Firmware Change Alert



D.11.4 Build 4

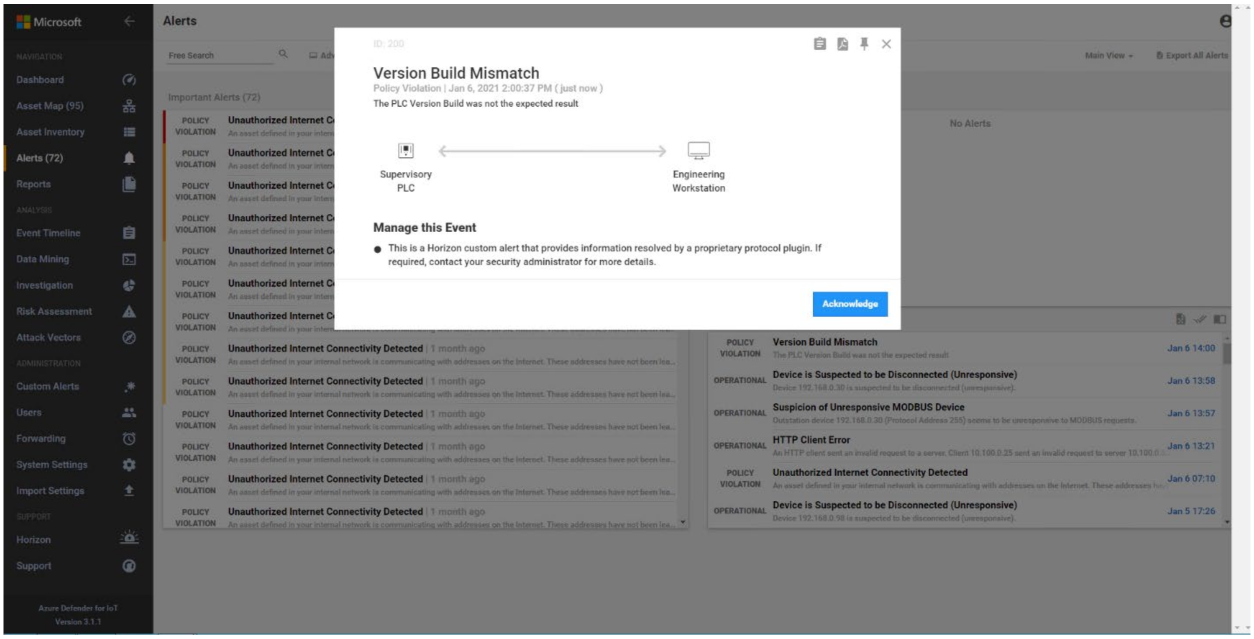
D.11.4.1 Configuration

- Behavior Anomaly Detection: Azure Defender for IoT
    - configured to receive packet streams from the DMZ, Testbed LAN, Supervisory LAN, and Control LAN
- Remote Access, User Authentication/User Authorization: Dispel
    - Dispel VDI is configured as the engineering workstation to connect through the Dispel Enclave to the Dispel Wicket to manage the Beckhoff PLC.

D.11.4.2 Test Results

Azure Defender for IoT alerts on the firmware update as shown below in Figure D-109.

Figure D-109 Azure Defender for IoT Alert Showing a Version Mismatch in the Firmware Build


Microsoft
Alerts
Dashboard
Asset Map (95)
Asset Inventory
Alerts (72)
Reports
Event Timeline
Data Mining
Investigation
Risk Assessment
Attack Vectors
Custom Alerts
Users
Forwarding
System Settings
Import Settings
Horizon
Support
Version Build Mismatch
Policy Violation | Jan 6, 2021 2:00:37 PM (just now)
The PLC Version Build was not the expected result
Supervisory PLC
Engineering Workstation
Manage this Event
This is a Horizon custom alert that provides information resolved by a proprietary protocol plugin. If required, contact your security administrator for more details.
Acknowledge
Unauthorized Internet Connectivity Detected
Device is Suspected to be Disconnected (Unresponsive)
Suspicion of Unresponsive MODBUS Device
HTTP Client Error

Appendix E Benefits of IoT Cybersecurity Capabilities

The National Institute of Standards and Technology's (NIST's) Cybersecurity for the Internet of Things (IoT) program supports development and application of standards, guidelines, and related tools to improve the cybersecurity of connected devices and the environments in which they are deployed. By collaborating with stakeholders across government, industry, international bodies, and academia, the program aims to cultivate trust and foster an environment that enables innovation on a global scale.

Cyber-physical components, including sensors and actuators, are being designed, developed, deployed, and integrated into networks at an ever-increasing pace. Many of these components are connected to the internet. IoT devices combine network connectivity with the ability to sense or affect the physical world. Stakeholders face additional challenges with applying cybersecurity controls as cyber-physical devices are further integrated.

NIST's Cybersecurity for IoT program has defined a set of device cybersecurity capabilities that device manufacturers should consider integrating into their IoT devices and that consumers should consider enabling/configuring in those devices. Device cybersecurity capabilities are cybersecurity features or functions that IoT devices or other system components (e.g., a gateway, proxy, IoT platform) provide through technical means (e.g., device hardware and software). Many IoT devices have limited processing and data storage capabilities and may not be able to provide these device cybersecurity capabilities on their own; instead, they may rely on other system components to provide these technical capabilities on their behalf. Nontechnical supporting capabilities are actions that a manufacturer or third-party organization performs in support of the cybersecurity of an IoT device. Examples of nontechnical support include providing information about software updates, instructions for configuration settings, and supply chain information.

Used together, device cybersecurity capabilities and nontechnical supporting capabilities can help mitigate cybersecurity risks related to the use of IoT devices while assisting customers in achieving their goals. If IoT devices are integrated into industrial control system (ICS) environments, device cybersecurity capabilities and nontechnical supporting capabilities can assist in securing the ICS environment.

E.1 Device Capabilities Mapping

Table E-1 lists device cybersecurity capabilities and nontechnical supporting capabilities as they map to the NIST *Cybersecurity Framework* Subcategories of particular importance to this project. It is acknowledged that IoT devices vary in their capabilities, and there may not be a clear delineation between the device cybersecurity capabilities that are provided by the IoT devices and those provided by another system component. It is also understood that the capabilities of cyber-physical components are evolving, so many of the mappings are not necessarily exact.

In this project, the focus was on the engineering workstations and not on the manufacturing components. The mapping presented in Table E-1 is a summary of both technical and nontechnical capabilities that would enhance the security of a manufacturing environment. It is acknowledged that many of the device cybersecurity capabilities may not be available in modern sensors and actuators and that other system elements (e.g., proxies, gateways) or other risk mitigation strategies (e.g., network segmentation) may be necessary.

Table E-1 Mapping of Device Cybersecurity Capabilities and Nontechnical Supporting Capabilities to NIST *Cybersecurity Framework* Subcategories of the ICS Project

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
PR.AC-1: Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes.	▪ Ability to uniquely identify the IoT device logically. ▪ Ability to uniquely identify a remote IoT device. ▪ Ability for the device to support a unique device ID. ▪ Ability to configure IoT device access control policies using IoT device identity. ▪ Ability to verify the identity of an IoT device. ▪ Ability to add a unique physical identifier at an external or internal location on the device authorized entities can access. ▪ Ability to set and change authentication configurations, policies, and limitations settings for the IoT device. ▪ Ability to create unique IoT device user accounts. ▪ Ability to identify unique IoT device user accounts. ▪ Ability to create organizationally defined	▪ Providing details for how to establish unique identification for each IoT device associated with the system and critical system components within which it is used. ▪ Providing communications and documentation detailing how to perform account management activities, using the technical IoT device capabilities, or through supporting systems and/or tools. ▪ Providing the details necessary to establish and implement unique identification for each IoT device associated with the system and critical system components within which it is used. ▪ Providing the details necessary to require unique identifiers for each IoT device associated with the system and critical system	AC-2 IA-2 IA-4 IA-5 IA-8 IA-12

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	accounts that support privileged roles with automated expiration conditions. ▪ Ability to establish organizationally defined user actions for accessing the IoT device and/or device interface. ▪ Ability to enable automation and reporting of account management activities. ▪ Ability to establish conditions for shared/group accounts on the IoT device. ▪ Ability to administer conditions for shared/group accounts on the IoT device. ▪ Ability to restrict the use of shared/group accounts on the IoT device according to organizationally defined conditions.	components within which it is used. ▪ Providing education explaining how to establish and enforce approved authorizations for logical access to IoT device information and system resources. ▪ Providing education explaining how to control access to IoT devices implemented within IoT device customer information systems. ▪ Providing education explaining how to enforce authorized access at the system level.	
PR.AC-3: Remote access is managed.	▪ Ability to configure IoT device access control policies using IoT device identity. ▪ Ability for the IoT device to differentiate between authorized and unauthorized remote users. ▪ Ability to authenticate external users and systems. ▪ Ability to securely interact with authorized external, third-party systems.	N/A	AC-17 AC-19 AC-20

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	▪ Ability to identify when an external system meets the required security requirements for a connection. ▪ Ability to establish secure communications with internal systems when the device is operating on external networks. ▪ Ability to establish requirements for remote access to the IoT device and/or IoT device interface, including: • usage restrictions • configuration requirements • connection requirements • manufacturer established requirement ▪ Ability to enforce the established local and remote access requirements. ▪ Ability to prevent external access to the IoT device management interface. ▪ Ability to control the IoT device's logical interface (e.g., locally or remotely). ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors.		

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
PR.AC-4: Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties.	▪ Ability to assign roles to IoT device user accounts. ▪ Ability to support a hierarchy of logical access privileges for the IoT device based on roles (e.g., admin, emergency, user, local, temporary). • Ability to establish user accounts to support role-based logical access privileges. • Ability to administer user accounts to support role-based logical access privileges. • Ability to use organizationally defined roles to define each user account's access and permitted device actions. • Ability to support multiple levels of user/process account functionality and roles for the IoT device. ▪ Ability to apply least privilege to user accounts. • Ability to create additional processes, roles (e.g., admin, emergency, temporary) and accounts as necessary to achieve least privilege. • Ability to apply least privilege settings within	▪ Providing the tools, assistance, instructions, and other types of information to support establishing a hierarchy of role-based privileges within the IoT device. ▪ Providing details about the specific types of manufacturer's needs to access the IoT device interfaces, such as for specific support, updates, ongoing maintenance, and other purposes. ▪ Providing documentation with instructions for the IoT device customer to follow for how to restrict interface connections that enable specific activities. ▪ Providing descriptions of the types of access to the IoT device that the manufacturer will require on an ongoing or regular basis. ▪ Providing detailed instructions for how to implement management and operational controls based on the role of the IoT device user, and not on an individual basis. ▪ Providing documentation and/or other communications describing how to implement management and operational	AC-2 AC-3 AC-5 AC-6 AC-14 AC-16 AC-24

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	the device (i.e., to ensure that the processes operate at privilege levels no higher than necessary to accomplish required functions). • Ability to limit access to privileged device settings that are used to establish and administer authorization requirements. • Ability for authorized users to access privileged settings. ▪ Ability to create organizationally defined accounts that support privileged roles with automated expiration conditions. ▪ Ability to enable automation and reporting of account management activities. ▪ Ability to establish conditions for shared/group accounts on the IoT device. ▪ Ability to administer conditions for shared/group accounts on the IoT device. ▪ Ability to restrict the use of shared/group accounts on the IoT device according to organizationally defined conditions.	controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing a detailed description of the other types of devices and systems that will access the IoT device during customer use of the device, and how they will access it. ▪ Providing communications and detailed instructions for implementing a hierarchy of privilege levels to use with the IoT device and/or necessary associated information systems. ▪ Providing communications and documentation detailing how to perform account management activities, using the technical IoT device capabilities, or through supporting systems and/or tools. ▪ Providing education explaining how to establish and enforce approved authorizations for logical access to IoT device information and system resources. ▪ Providing education explaining how to control access to IoT devices	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	▪ Ability to implement dynamic access control approaches (e.g., service-oriented architectures) that rely on: • run-time access control decisions facilitated by dynamic privilege management. • organizationally defined actions to access/use device. ▪ Ability to allow information sharing capabilities based upon the type and/or role of the user attempting to share the information. ▪ Ability to restrict access to IoT device software, hardware, and data based on user account roles, used with proper authentication of the identity of the user to determine type of authorization. ▪ Ability to establish limits on authorized concurrent device sessions. ▪ Ability to restrict updating actions to authorized entities. ▪ Ability to restrict access to the cybersecurity state indicator to authorized entities. ▪ Ability to revoke access to the IoT device.	implemented within IoT device customer information systems. ▪ Providing education explaining how to enforce authorized access at the system level. ▪ Providing education and supporting materials explaining how to establish roles and responsibilities for IoT device data security, using the device capabilities and/or other services that communicate or interface with the device. ▪ Providing education and supporting materials describing the IoT device capabilities for role-based controls, and how to establish different roles within the IoT device. ▪ Providing education and supporting materials for how to establish roles to support IoT device policies, procedures, and associated documentation.	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks).	▪ Ability for the IoT device to require authentication prior to connecting to the device. ▪ Ability for the IoT device to support a second, or more, authentication method(s) such as: • temporary passwords or other one-use log-on credentials • third-party credential checks • biometrics • hard tokens ▪ Ability to authenticate external users and systems. ▪ Ability to verify and authenticate any update before installing it.	▪ Providing detailed instructions and guidance for establishing activities performed by the IoT device that do not require identification or authentication. ▪ Providing documentation describing the specific IoT platforms used with the device to support required IoT authentication control techniques. ▪ Providing documentation with details describing external authentication by IoT platforms and associated authentication methods that can be used with the IoT device.	AC-7 AC-8 AC-9 AC-12 AC-14 IA-2 IA-3 IA-4 IA-5 IA-8 IA-11
PR.DS-1: Data-at-rest is protected.	▪ Ability to execute cryptographic mechanisms of appropriate strength and performance. ▪ Ability to obtain and validate certificates. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to change keys securely. ▪ Ability to generate key pairs.	▪ Providing detailed instructions for how to implement management and operational controls for securely handling and retaining IoT device data, associated systems data, and data output from the IoT device. ▪ Providing education describing how to securely handle and retain IoT device data, associated systems data, and data output from the IoT device to meet	SC-28 MP-2 MP-4 MP-5

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	▪ Ability to store encryption keys securely. ▪ Ability to cryptographically store passwords at rest, as well as device identity and other authentication data. ▪ Ability to support data encryption and signing to prevent data from being altered in device storage. ▪ Ability to secure data stored locally on the device. ▪ Ability to secure data stored in remote storage areas (e.g., cloud, server). ▪ Ability to utilize separate storage partitions for system and user data. ▪ Ability to protect the audit information through mechanisms such as: • encryption • digitally signing audit files • securely sending audit files to another device • other protections created by the device manufacturer	requirements of the IoT device customers' organizational security policies, contractual requirements, applicable Federal laws, Executive Orders, directives, policies, regulations, standards, and other legal requirements.	
PR.DS-6: Integrity checking mechanisms are used to verify software, firm-	▪ Ability to identify software loaded on the IoT device based on IoT device identity.	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data	SC-16 SI-7 MP-4

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
ware, and information integrity.	▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory).	obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity.	MP-5

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
PR.IP-4: Backups of information are conducted, maintained, and tested.	N/A	▪ Providing education to IoT device customers covering the instructions and details necessary for them to create accurate backups and to recover the backups when necessary. ▪ Providing education to IoT device customers that includes instructions describing how to back up data from systems where IoT device data is stored. ▪ Providing awareness reminders and tips to IoT device customers (e.g., directly in person, in videos, in an online webinar) for various aspects involved with backing up the IoT device data.	CP-4 CP-9
PR.MA-1: Maintenance and repair of organizational assets are performed and logged, with approved and controlled tools.	N/A	▪ Providing details about the types of, and situations that trigger, local and/or remote maintenance activities required once the device is purchased and deployed in the organization's digital ecosystem or within an individual consumer's home. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity.	MA-2 MA-3 MA-5 MA-6

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		▪ Providing other information and actions as necessary for physically securing, and securely using, the IoT device based upon the IoT device use, purpose, and other contextual factors related to the digital ecosystem(s) within which they are intended to be used. ▪ Providing the details necessary for IoT device customers to implement only organizationally approved IoT device diagnostic tools within their system. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		maintenance operations that the IoT device customer is required to perform. If such comprehensive IoT device maintenance operations documentation does not exist, the manufacturer should clearly communicate to IoT device customers that the user must perform these operations themselves. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing the details necessary to enable IoT device customers to monitor onsite and offsite IoT device maintenance activities. ▪ Providing the details necessary to implement management and operational controls for IoT device maintenance personnel and associated authorizations, and record-keeping of maintenance organizations and personnel.	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		▪ Providing communications describing the type and nature of the local and/or remote maintenance activities that will involve and require manufacturer personnel, or their contractors, once the device is purchased and deployed in the IoT device customer's organization. ▪ Providing IoT device customers with the details necessary to implement management and operational controls in support of their security policies and legal requirements for IoT device maintenance for assigned organizationally defined personnel or roles to follow. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing the details necessary for customers to document attempts to obtain IoT device components or IoT device information system service documentation when such documentation is either unavailable or nonexistent, and documenting the appropriate response for manufacturer employees, or	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		supporting entities, to follow. ▪ Providing a process for IoT device customers to contact the manufacturer to ask questions or obtain help related to the IoT device configuration settings. ▪ Providing information to allow for in-house support from within the IoT device customer organization. ▪ Providing education explaining how to inspect IoT device and/or use maintenance tools to ensure the latest software updates and patches are installed. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing education that explains the legal requirements governing IoT device maintenance responsibilities or how to meet specific types of legal requirements when using the IoT device.	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
PR.MA-2: Remote maintenance of organizational assets is approved, logged, and performed in a manner that prevents unauthorized access.	N/A	▪ Providing details about the types of, and situations that trigger, local and/or remote maintenance activities required once the device is purchased and deployed in the organization's digital ecosystem or within an individual consumer's home. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing other information and actions as necessary for physically securing, and securely using, the IoT device based upon the IoT device use, purpose, and other contextual factors related to the digital ecosystem(s) within which they are intended to be used. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the	MA-4

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		manufacturer's supporting entities. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing the details necessary to enable IoT device customers to monitor onsite and offsite IoT device maintenance activities. ▪ Providing the details necessary for maintaining records for nonlocal IoT device maintenance and diagnostic activities. ▪ Providing the details necessary to implement management and operational controls for IoT device maintenance personnel and associated authorizations, and record-keeping of maintenance organizations and personnel. ▪ Providing communications describing the type and nature of the local and/or remote maintenance activities that will involve and require manufacturer personnel, or their contractors, once the device is purchased and deployed in	

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
		the IoT device customer's organization. ▪ Providing IoT device customers with the details necessary to implement management and operational controls in support of their security policies and legal requirements for IoT device maintenance for assigned organizationally defined personnel or roles to follow. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks.	
DE.AE-1: A baseline of network operations and expected data flows for users and systems is established and managed.	N/A	▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems.	AC-4 CA-3 CM-2 SI-4
DE.AE-2: Detected events are analyzed to understand attack targets and methods.	N/A	▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched.	AU-6 CA-7 IR-4 SI-4
DE.AE-3: Event data are collected and correlated from multiple sources and sensors.	▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing	▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device.	AU-6 AU-12 CA-7 IR-4 IR-5 SI-4

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
	information can be checked to allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time.		
DE.CM-1: The information system and assets are monitored to identify cybersecurity events and verify the effectiveness of protective measures.	▪ Ability to monitor specific actions based on the IoT device identity. ▪ Ability to access information about the IoT device's cybersecurity state and other necessary data. ▪ Ability to monitor for organizationally defined cybersecurity events (e.g., expected state change) that may occur on or involving the IoT device. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. (The device may be able to perform this check itself or provide the information necessary for an external process to check). ▪ Ability to monitor communications traffic.	▪ Providing information that describes the types of system monitoring information generated from, or associated with, the IoT device and instructions for obtaining that information. ▪ Providing documentation describing the types of monitoring tools with which the IoT device is compatible, and recommendations for how to configure the IoT device to best work with such monitoring tools. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing how to perform monitoring activities.	AU-12 CA-7 CM-3 SC-7 SI-4
DE.CM-3: Personnel activity is monitored to detect potential cybersecurity events.	N/A	N/A	AC-2 AU-12 CA-7 CM-3 SC-5 SC-7

Cybersecurity Framework v1.1 Subcategory	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities	NIST SP 800-53 Rev. 5
			SI-4
DE.CM-7: Monitoring for unauthorized personnel, connections, devices, and software is performed.	▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. (The device may be able to perform this check itself or provide the information necessary for an external process to check). ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.	AC-2 AU-12 AU-13 CA-7 CM-10 CM-11

E.2 Device Capabilities Supporting Functional Test Scenarios

In this project, the focus was on the engineering workstations and not on the manufacturing components. It is acknowledged that many of the **device cybersecurity capabilities** may not be available in modern sensors and actuators and that other system elements (e.g., proxies, gateways) or other risk mitigation strategies (e.g., network segmentation) may be necessary.

Table E-2 builds on the functional test scenarios included in Section 5 of this document. The table lists both **device cybersecurity capabilities** and **nontechnical supporting capabilities** that map to relevant *Cybersecurity Framework* Subcategories for each of the functional test scenarios. If IoT devices are integrated into future efforts or a production ICS environment, selecting devices and/or third parties that provide these capabilities can help achieve the respective functional requirements.

It is acknowledged that IoT devices vary in their capabilities, and there may not be a clear delineation between **the device cybersecurity capabilities** that are provided by the IoT devices and those provided by another system component. It is also understood that the capabilities of cyber-physical components are evolving, so many of the mappings are not necessarily exact.

In this project, the focus was on the engineering workstations and not on the manufacturing components. It is acknowledged that many of the **device cybersecurity capabilities** may not be available in modern sensors and actuators and that other system elements (e.g., proxies, gateways) or other risk mitigation strategies (e.g., network segmentation) may be necessary.

Table E-2 Device Cybersecurity Capabilities and Nontechnical Supporting Capabilities that Map to Each of the Functional Test Scenarios

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
Scenario 1: Protect Host from Mal-ware via USB: This test will demon-strate blocking the introduction of malware through physical access to a workstation within the manufacturing system. **PR.DS-6** **PR.MA-2** **DE.AE-2**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures,	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory).	▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing the details necessary to enable IoT device customers to monitor onsite and offsite IoT device maintenance activities. ▪ Providing communications describing the type and nature of the local and/or remote maintenance activities that will involve and require manufacturer personnel, or their contractors, once the device is purchased and deployed in the IoT device customer's organization. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched.
Scenario 2: Protect Host from Mal-ware via Network Vector This test will demonstrate the	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures.	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
detection of mal-ware introduction from the network. **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications)	▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when	operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.
Scenario 3: Protect Host from Malware via Remote Access Connections: This test will demonstrate blocking malware attempting to infect manufacturing system through authorized remote access connections.	▪ Ability to uniquely identify the IoT device logically. ▪ Ability to uniquely identify a remote IoT device. ▪ Ability for the device to support a unique device ID. ▪ Ability to configure IoT device access control policies using IoT device identity. ▪ Ability to verify the identity of an IoT device.	▪ Providing details for how to establish unique identification for each IoT device associated with the system and critical system components within which it is used. ▪ Providing communications and documentation detailing how to perform account management activities, using the technical IoT device capabilities, or through supporting systems and/or tools. ▪ Providing the details necessary to establish and implement unique identification for each IoT device associated with the system and critical system components within which it is used.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
PR.AC-1 **PR.AC-3** **PR.AC-4** **PR.AC-7** **PR.MA-1** **PR.MA-2** **DE.CM-3** **DE.CM-7**	▪ Ability to add a unique physical identifier at an external or internal location on the device authorized entities can access. ▪ Ability to set and change authentication configurations, policies, and limitations settings for the IoT device. ▪ Ability to revoke access to the device. ▪ Ability to create unique IoT device user accounts. ▪ Ability to identify unique IoT device user accounts. ▪ Ability to create organizationally defined accounts that support privileged roles with automated expiration conditions. ▪ Ability to configure IoT device access control policies using IoT device identity. ▪ Ability to authenticate external users and systems.	▪ Providing the tools, assistance, instructions, and other types of information to support establishing a hierarchy of role-based privileges within the IoT device. ▪ Providing details about the specific types of manufacturer's needs to access the IoT device interfaces, such as for specific support, updates, ongoing maintenance, and other purposes. ▪ Providing education explaining how to control access to IoT devices implemented within IoT device customer information systems. ▪ Providing education explaining how to enforce authorized access at the system level. ▪ Providing detailed instructions and guidance for establishing activities performed by the IoT device that do not require identification or authentication. ▪ Providing documentation describing the specific IoT platforms used with the device to support required IoT authentication control techniques. ▪ Providing documentation with details describing external authentication by IoT platforms and associated authentication methods that can be used with the IoT device. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to securely interact with authorized external, third-party systems. ▪ Ability to identify when an external system meets the required security requirements for a connection. ▪ Ability to establish secure communications with internal systems when the device is operating on external networks. ▪ Ability to establish requirements for remote access to the IoT device and/or IoT device interface. ▪ Ability to enforce the established local and remote access requirements. ▪ Ability to prevent external access to the IoT device management interface. ▪ Ability to assign roles to IoT device user accounts.	▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing details about the types of, and situations that trigger, local and/or remote maintenance activities required once the device is purchased and deployed in the organization's digital ecosystem or within an individual consumer's home. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to support a hierarchy of logical access privileges for the IoT device based on roles. ▪ Ability to apply least privilege to user accounts. ▪ Ability to enable automation and reporting of account management activities. ▪ Ability for the IoT device to require authentication prior to connecting to the device. ▪ Ability for the IoT device to support a second, or more, authentication method(s). ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors.	

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	
Scenario 4: Protect Host from Unauthorized Application Installation: This test will demonstrate blocking the installation and execution of unauthorized applications on workstation in the manufacturing system. **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification.	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting).	▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
Scenario 5: Protect from Unauthorized Addition of a Device: This test will demonstrate the detection of an unauthorized device connecting to the manufacturing system. **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation).	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings.	▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.
Scenario 6: Detect Unauthorized Device-to-Device Communications: This test will demonstrate the detection of unau-	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures.	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
thorized communi-cations between devices. **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications)	controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities.

<table>
<tr><th>Scenario ID and Description with Cybersecurity Framework Sub-categories</th><th>Device Cybersecurity Capabilities</th><th>Manufacturer Nontechnical Supporting Capabilities</th></tr>
<tr><td></td><td>in read-only media (e.g., Read Only Memory).
Ability to provide a physical indicator of sensor use.
Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting).
Ability to keep an accurate internal system time.
Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities.
Ability to monitor changes to the configuration settings.
Ability to detect remote activation attempts.
Ability to detect remote activation of sensors.
Ability to take organizationally defined actions when</td><td>Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform.
Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer.
Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities.
Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks.
Providing education for how to scan for critical software updates and patches.
Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems.
Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched.
Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device.
Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report</td></tr>
</table>

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.
Scenario 7: Protect from Unauthorized Modification and Deletion of Files: This test will demonstrate protection of files from unauthorized deletion both locally and on network file share. **PR.DS-1** **PR.DS-6** **PR.IP-4** **PR.MA-1** **DE.AE-2**	▪ Ability to execute cryptographic mechanisms of appropriate strength and performance. ▪ Ability to obtain and validate certificates. ▪ Ability to change keys securely. ▪ Ability to generate key pairs. ▪ Ability to store encryption keys securely. ▪ Ability to cryptographically store passwords at rest, as well as device identity and other authentication data.	▪ Providing detailed instructions for how to implement management and operational controls for securely handling and retaining IoT device data, associated systems data, and data output from the IoT device. ▪ Providing education describing how to securely handle and retain IoT device data, associated systems data, and data output from the IoT device to meet requirements of the IoT device customers' organizational security policies, contractual requirements, applicable Federal laws, Executive Orders, directives, policies, regulations, standards, and other legal requirements. ▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion.

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to support data encryption and signing to prevent data from being altered in device storage. ▪ Ability to secure data stored locally on the device. ▪ Ability to secure data stored in remote storage areas (e.g., cloud, server). ▪ Ability to utilize separate storage partitions for system and user data. ▪ Ability to protect the audit information through mechanisms such as: ▪ encryption ▪ digitally signing audit files ▪ securely sending audit files to another device ▪ other protections created by the device manufacturer ▪ Ability to identify software loaded on the IoT device based on IoT device identity.	▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing education to IoT device customers covering the instructions and details necessary for them to create accurate backups and to recover the backups when necessary. ▪ Providing education to IoT device customers that includes instructions describing how to back up data from systems where IoT device data is stored. ▪ Providing awareness reminders and tips to IoT device customers (e.g., directly in person, in videos, in an online

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it.	webinar) for various aspects involved with backing up the IoT device data. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory).	▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched.
Scenario 8: Detect Unauthorized Modification of PLC Logic: This test will demonstrate the detection of PLC logic modification. **PR.AC-3** **PR.AC-7** **PR.DS-6** **PR.MA-1** **PR.MA-2** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to configure IoT device access control policies using IoT device identity. ▪ Ability to authenticate external users and systems. ▪ Ability to securely interact with authorized external, third-party systems. ▪ Ability to identify when an external system meets the required security requirements for a connection. ▪ Ability to establish secure communications with internal systems when the device is operating on external networks. ▪ Ability to establish requirements for remote	▪ Providing detailed instructions and guidance for establishing activities performed by the IoT device that do not require identification or authentication. ▪ Providing documentation describing the specific IoT platforms used with the device to support required IoT authentication control techniques. ▪ Providing documentation with details describing external authentication by IoT platforms and associated authentication methods that can be used with the IoT device. ▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	access to the IoT device and/or IoT device interface. ▪ Ability to enforce the established local and remote access requirements. ▪ Ability to prevent external access to the IoT device management interface. ▪ Ability for the IoT device to require authentication prior to connecting to the device. ▪ Ability for the IoT device to support a second, or more, authentication method(s). ▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms.	▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer’s supporting entities.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use.	▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing the details necessary to enable IoT device customers to monitor onsite and offsite IoT device maintenance activities. ▪ Providing communications describing the type and nature of the local and/or remote maintenance activities that will involve and require manufacturer personnel, or their contractors, once the device is purchased and deployed in the IoT device customer's organization.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a	▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	Universal Serial Bus [USB] port is present).	
Scenario 9: Protect from Modification of Historian Data: This test will demonstrate the blocking of modification of historian archive data. **PR.DS-6** **PR.MA-1** **DE.AE-2**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures,	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory).	▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
		▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched.
Scenario 10: Detect Sensor Data Manipulation: This test will demonstrate detection of atypical data reported to the historian. **PR.IP-4** **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3** **DE.CM-7**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective	▪ Providing education to IoT device customers covering the instructions and details necessary for them to create accurate backups and to recover the backups when necessary. ▪ Providing education to IoT device customers that includes instructions describing how to back up data from systems where IoT device data is stored. ▪ Providing awareness reminders and tips to IoT device customers (e.g., directly in person, in videos, in an online webinar) for various aspects involved with backing up the IoT device data. ▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities.	▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer's supporting entities.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities. ▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
		actions to the monitoring service of the manufacturer's supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.
Scenario 11: Detect Unauthorized Firmware Modification: This test will demonstrate the detection of device firmware modification **PR.DS-6** **PR.MA-1** **DE.AE-1** **DE.AE-2** **DE.AE-3** **DE.CM-1** **DE.CM-3**	▪ Ability to identify software loaded on the IoT device based on IoT device identity. ▪ Ability to verify digital signatures. ▪ Ability to run hashing algorithms. ▪ Ability to perform authenticated encryption algorithms. ▪ Ability to compute and compare hashes. ▪ Ability to utilize one or more capabilities to protect transmitted data from	▪ Providing documentation and/or other communications describing how to implement management and operational controls to protect data obtained from IoT devices and associated systems from unauthorized access, modification, and deletion. ▪ Providing communications to IoT device customers describing how to implement management and operational controls to protect IoT device data integrity and associated systems data integrity. ▪ Providing IoT device customers with the details necessary to support secure implementation of the IoT device and associated systems data integrity controls. ▪ Providing IoT device customers with documentation describing the data integrity controls built into the IoT device and how to use them. If there are no data integrity controls built into the IoT device, include documentation

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
DE.CM-7	unauthorized access and modification. ▪ Ability to validate the integrity of data transmitted. ▪ Ability to verify software updates come from valid sources by using an effective method (e.g., digital signatures, checksums, certificate validation). ▪ Ability to verify and authenticate any update before installing it. ▪ Ability to store the operating environment (e.g., firmware image, software, applications) in read-only media (e.g., Read Only Memory). ▪ Ability to provide a physical indicator of sensor use. ▪ Ability to send requested audit logs to an external audit process or information system (e.g., where its auditing information can be checked to	explaining to IoT device customers the ways to achieve IoT device data integrity. ▪ Providing details for how to review and update the IoT device and associated systems while preserving data integrity. ▪ Providing instructions and documentation describing the physical and logical access capabilities necessary to the IoT device to perform each type of maintenance activity. ▪ Providing detailed documentation describing the tools manufacturers require for IoT device diagnostics activities. ▪ Providing the details and instructions to perform necessary IoT device maintenance activities and repairs. ▪ Providing communications and comprehensive documentation describing the IoT device maintenance operations performed by the manufacturer and the manufacturer’s supporting entities. ▪ Providing communications and comprehensive documentation describing maintenance operations that the IoT device customer is required to perform. ▪ Providing communications that include details for the recommended events that will trigger IoT device system reviews and/or maintenance by the manufacturer. ▪ Providing communications and documentation detailing how to perform recommended local and/or remote maintenance activities.

Scenario ID and Description with *Cybersecurity Framework* Sub-categories	Device Cybersecurity Capabilities	Manufacturer Nontechnical Supporting Capabilities
	allow for review, analysis, and reporting). ▪ Ability to keep an accurate internal system time. ▪ Ability to support a monitoring process to check for disclosure of organizational information to unauthorized entities. ▪ Ability to monitor changes to the configuration settings. ▪ Ability to detect remote activation attempts. ▪ Ability to detect remote activation of sensors. ▪ Ability to take organizationally defined actions when unauthorized hardware and software components are detected (e.g., disallow a flash drive to be connected even if a Universal Serial Bus [USB] port is present).	▪ Providing documented descriptions of the specific maintenance procedures for defined maintenance tasks. ▪ Providing education for how to scan for critical software updates and patches. ▪ Providing documentation describing how to implement and securely deploy monitoring devices and tools for IoT devices and associated systems. ▪ Providing documentation describing IoT device behavior indicators that could occur when an attack is being launched. ▪ Providing documentation describing the types of usage and environmental systems data that can be collected from the IoT device. ▪ Providing appropriate tools, assistance, instructions, or other details describing the capabilities for monitoring the IoT device and/or for the IoT device customer to report actions to the monitoring service of the manufacturer’s supporting entity. ▪ Providing the details necessary to monitor IoT devices and associated systems. ▪ Providing documentation describing details necessary to identify unauthorized use of IoT devices and their associated systems. ▪ Providing documentation that describes indicators of unauthorized use of the IoT device.

NIST SPECIAL PUBLICATION 1800-10C

Protecting Information and System Integrity in Industrial Control System Environments:

Cybersecurity for the Manufacturing Sector

Volume C:
How-To Guides

Michael Powell
National Cybersecurity Center of Excellence
National Institute of Standards and Technology

Michael Pease
Keith Stouffer
CheeYee Tang
Timothy Zimmerman
Engineering Laboratory
National Institute of Standards and Technology

Matthew Zopf
Strativia
Largo, Maryland

Joseph Brule
Chelsea Deane
John Hoyt
Mary Raguso
Aslam Sherule
Kangmin Zheng
The MITRE Corporation
McLean, Virginia

FINAL

March 2022

This publication is available free of charge from
https://doi.org/10.6028/NIST.SP.1800-10

The first draft of this publication is available free of charge from
https://www.nccoe.nist.gov/publications/practice-guide/protecting-information-and-system-integrity-industrial-control-system-draft

DISCLAIMER

Certain commercial entities, equipment, products, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that entities, equipment, products, or materials are necessarily the best available for the purpose.

While NIST and the NCCoE address goals of improving management of cybersecurity and privacy risk through outreach and application of standards and best practices, it is the stakeholder's responsibility to fully perform a risk assessment to include the current threat, vulnerabilities, likelihood of a compromise, and the impact should the threat be realized before adopting cybersecurity measures such as this recommendation.

Domain name and IP addresses shown in this guide represent an example domain and network environment to demonstrate the NCCoE project use case scenarios and the security capabilities.

National Institute of Standards and Technology Special Publication 1800-10C, Natl. Inst. Stand. Technol. Spec. Publ. 1800-10C, 128 pages, March 2022, CODEN: NSPUE2.

FEEDBACK

As a private-public partnership, we are always seeking feedback on our practice guides. We are particularly interested in seeing how businesses apply NCCoE reference designs in the real world. If you have implemented the reference design, or have questions about applying it in your environment, please email us at manufacturing_nccoe@nist.gov.

All comments are subject to release under the Freedom of Information Act (FOIA).

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, easily adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align more easily with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Today's manufacturing organizations rely on industrial control systems (ICS) to conduct their operations. Increasingly, ICS are facing more frequent, sophisticated cyber attacks—making manufacturing the second-most targeted industry (C. Singleton et al., X-Force Threat Intelligence Index 2021, IBM, February 2021, https://www.ibm.com/security/data-breach/threat-intelligence). Cyber attacks against ICS threaten operations and worker safety, resulting in financial loss and harm to the organization's reputation.

The architecture and solutions presented in this guide are built upon standards-based, commercially available products, and represent some of the possible solutions. The solutions implement standard cybersecurity capabilities, such as behavioral anomaly detection, application allowlisting, file integrity-checking, change control management, and user authentication and authorization. The solution was tested in two distinct lab settings: a discrete manufacturing work cell, which represents an assembly line

production, and a continuous process control system (PCS), which represents chemical manufacturing industries.

Organizations that are interested in protecting the integrity of the manufacturing system and information from destructive malware, insider threats, and unauthorized software should first conduct a risk assessment and determine the appropriate security capabilities required to mitigate those risks. Once the security capabilities are identified, the sample architecture and solution presented in this document may be used.

The security capabilities of the example solution are mapped to NIST's Cybersecurity Framework, the National Initiative for Cybersecurity Education Framework, and NIST Special Publication 800-53.

KEYWORDS

Application allowlisting; behavioral anomaly detection; file integrity checking; firmware modification; industrial control systems; manufacturing; remote access; software modification; user authentication; user authorization.

ACKNOWLEDGEMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Dan Frechette	Microsoft
Ian Schmertzler	Dispel
Ben Burke	Dispel
Chris Jensen	Tenable
Bethany Brower	VMWare
Dennis Hui	OSIsoft (now part of AVEVA)
John Matranga	OSIsoft (now part of AVEVA)
Michael A. Piccalo	Forescout
Tim Jones	Forescout
Yejin Jang	Forescout
Samantha Pelletier	TDI Technologies
Rusty Hale	TDI Technologies
Steve Petruzzo	GreenTec-USA
Josh Carlson	Dragos
Alex Baretta	Dragos

The Technology Partners/Collaborators who participated in this build submitted their capabilities in response to a notice in the Federal Register. Respondents with relevant capabilities or product

components were invited to sign a Cooperative Research and Development Agreement (CRADA) with NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Product
Carbon Black (VMware)	Carbon Black App Control
Microsoft	Azure Defender for the internet of things (IoT) (incorporating technology from the acquisition of CyberX)
Dispel	Dispel Wicket ESI Dispel Enclave Dispel VDI (Virtual Desktop Interface)
Dragos	Dragos Platform
Forescout	eyeInspect (Formerly SilentDefense) ICS Patrol EyeSight
GreenTec	WORMdisk and ForceField
OSIsoft (now part of AVEVA)	PI System (which comprises products such as PI Server, PI Vision and others)
TDi Technologies	ConsoleWorks
Tenable	Tenable.ot

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly to conform to the publication and from which no deviation is permitted. The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited. The terms “may” and “need not” indicate a course of action permissible within the limits of the publication. The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

PATENT DISCLOSURE NOTICE

NOTICE: The Information Technology Laboratory (ITL) has requested that holders of patent claims whose use may be required for compliance with the guidance or requirements of this publication disclose such patent claims to ITL. However, holders of patents are not obligated to respond to ITL calls for patents and ITL has not undertaken a patent search in order to identify which, if any, patents may apply to this publication.

As of the date of publication and following call(s) for the identification of patent claims whose use may be required for compliance with the guidance or requirements of this publication, no such patent claims have been identified to ITL.

No representation is made or implied by ITL that licenses are not required to avoid patent infringement in the use of this publication.

Contents

List of Figures

List of Tables

1 Introduction

The following volume of this guide shows information technology (IT) professionals and security engineers how we implemented this example solution. We cover all the products employed in this reference design. We do not re-create the product manufacturers' documentation, which is presumed to be widely available. Rather, these volumes show how we incorporated the products together in our environment.

Note: These are not comprehensive tutorials. There are many possible service and security configurations for these products that are out of scope for this reference design.

1.1 How to Use this Guide

This NIST Cybersecurity Practice Guide demonstrates a modular design and provides users with the information they need to replicate the described manufacturing industrial control system (ICS) security solutions, specifically focusing on information and system integrity. This reference design is modular and can be deployed in whole or in part.

This guide contains three volumes:

- NIST SP 1800-10A: *Executive Summary*
- NIST SP 1800-10B: *Approach, Architecture, and Security Characteristics* – what we built and why
- NIST SP 1800-10C: *How-To Guides* – instructions for building the example solution (**this document**)

Depending on your role in your organization, you might use this guide in different ways:

Senior information technology (IT) executives, including chief information security and technology officers, will be interested in the Executive Summary, NIST SP 1800-10A, which describes the following topics:

- challenges that enterprises face in ICS environments in the manufacturing sector
- example solution built at the NCCoE
- benefits of adopting the example solution

Technology or security program managers might share the *Executive Summary*, NIST SP 1800-10A, with your leadership to help them understand the importance of adopting a standards-based solution. Doing so can strengthen their information and system integrity practices by leveraging capabilities that may already exist within their operating environment or by implementing new capabilities.

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in *NIST SP 1800-10B*, which describes what we did and why. The following sections will be of particular interest:

- Section 3.4.4, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices.

IT professionals who want to implement an approach like this will find this whole practice guide useful. You can use this How-To portion of the guide, *NIST SP 1800-10C*, to replicate all or parts of the build created in our lab. This How-To portion of the guide provides specific product installation, configuration, and integration instructions for implementing the example solution. We do not recreate the product manufacturers' documentation, which is generally widely available. Rather, we show how we incorporated the products together in our environment to create an example solution.

This guide assumes that IT professionals have experience implementing security products within the enterprise. While we have used a suite of commercial products to address this challenge, this guide does not endorse any products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of this manufacturing ICS solution. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and best practices. Section 3.5, Technologies, in *NIST SP 1800-10B,* lists the products that we used and maps them to the cybersecurity controls provided by this reference solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution, but a possible solution. This is a draft guide. We seek feedback on its contents and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to manufacturing_nccoe@nist.gov.

1.1 Build Overview

The NCCoE partnered with NIST's Engineering Laboratory (EL) to provide real-world scenarios that could happen in ICS in the manufacturing sector. This collaboration spawned four unique builds: two builds within the Collaborative Robotics (CRS) environment and two builds within the Process Control System (PCS) environment. For each build, the NCCoE and the EL performed eleven scenarios. The step-by-step instructions on how each product was installed and configured in this lab environment are outlined in this document. For more information on the two environments refer to Section 4.5 in *NIST SP 1800-10B*. Additionally, Appendix B of this Volume contains the four build architecture diagrams for reference.

1.2 Typographic Conventions

The following table presents typographic conventions used in this volume.

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For language use and style guidance, see the *NCCoE Style Guide*.
Bold	names of menus, options, command buttons, and fields	Choose **File** > **Edit**.
Monospace	command-line input, on-screen computer output, sample code examples, and status codes	`mkdir`

Typeface/Symbol	Meaning	Example
Monospace Bold	command-line user input contrasted with computer output	**`service sshd start`**
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

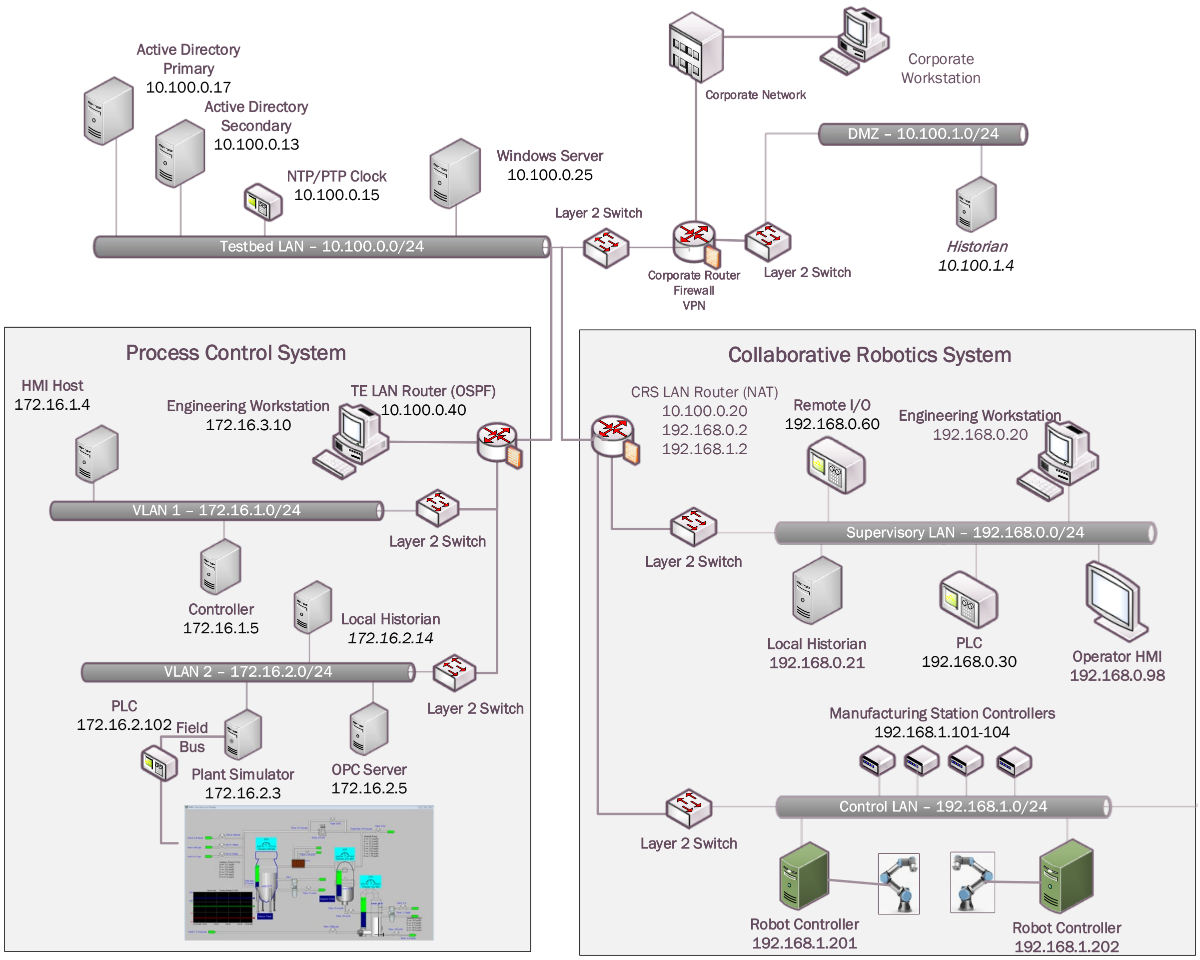
1.3 Logical Architecture Summary

The security mechanisms and technologies were integrated into the existing NIST Cybersecurity for Smart Manufacturing Systems (CSMS) lab environment. This cybersecurity performance testbed for ICS is comprised of the PCS and the CRS environments along with additional networking capabilities to emulate common manufacturing environments. For more information see An *Industrial Control System Cybersecurity Performance Testbed*, NISTIR 8089, http://nvlpubs.nist.gov/nistpubs/ir/2015/NIST.IR.8089.pdf.

Typically, manufacturing organizations have unique cyber-ecosystems and specific needs for their operations. To demonstrate the modularity and interoperability of the provided solutions, this project used available Cooperative Research and Development Agreement (CRADA) partner technologies to assemble four "builds" deployed across both the PCS and CRS. Additionally, to increase the diversity of technologies between builds, two of the builds also utilized open source solutions (Security Onion Wazuh), native operating system features (Windows Software Restriction Policies [SRP]), and a Cisco Adaptive Security Appliance (ASA) device configured with the AnyConnect virtual private network (VPN) client.

Figure 1-1 depicts a high-level architecture for the demonstration environment consisting of a Testbed Local Area Network (LAN), a demilitarized zone (DMZ), the PCS, and the CRS. The environment utilizes a combination of physical and virtual systems and maintains a local network time protocol (NTP) server for time synchronization. Additionally, the environment utilizes virtualized Active Directory (AD) servers for domain services. The tools used to support information and system integrity are deployed and integrated in the DMZ, Testbed LAN, PCS, and CRS per vendor recommendations and standard practices as described in the detailed sections for each build.

Figure 1-1: CSMS Network Architecture



In summary, there are six networks within the CSMS architecture:

Testbed LAN: This network is where the majority of the collaborators' products are installed. This LAN has access to the PCS and CRS environments. Other systems, such as AD, an NTP server, and a Windows server, are also located on this LAN. The Testbed LAN has three gateways to other network segments, including 10.100.0.1 to reach the DMZ and the corporate network, 10.100.0.20 as a network address translation (NAT) interface to the CRS environment, and 10.100.0.40 as the gateway to the PCS environment.

DMZ: A demilitarized zone that separates the corporate network from the operational technology (OT) network. Many of the collaborators' products are also installed in the DMZ. The DMZ is used across the PCS and CRS environments.

PCS Virtual Local Area Network (VLAN) 1: This is the operations LAN within the PCS environment. This LAN simulates a central control room environment. The gateway interface for this network segment is 172.16.1.1

PCS VLAN 2: This is the supervisory LAN within the PCS environment. This LAN simulates the process operation/manufacturing environment, which consists of the operating plant, programmable logic controller (PLC)s, object linking and embedding for process control (OPC) server, and data historian. The gateway interface for this network segment is 172.16.2.1

CRS Supervisory LAN: This LAN is within the CRS environment. The historian, PLCs, operating human machine interface (HMI), Engineering workstation, and remote input/output devices are connected to this network. The gateway interface for this network segment is 192.168.0.2

CRS Control LAN: This LAN is within the CRS environment. The robot controllers and manufacturing station controllers are connected to this network. The gateway interface for this network segment is 192.168.1.2

The test bed networks used static IPv4 addresses exclusively, and the subnet masks were set to 255.255.255.0. No IPv6 addresses were used. This setup is consistent with industry practice. Specific Internet Protocol (IP) addresses are listed for each component in the following sections.

For an in-depth view of the architectures PCS and CRS builds, specific build architecture diagrams can be found in Volume B of this practice guide, Section 4.3, Process Control System, and Section 4.4, Collaborative Robotics System.

2 Product Installation Guides

This section of the practice guide contains detailed instructions for installing and configuring all the products used to build the example solutions.

2.1 Dispel Remote Access

Dispel is a remote access tool for OT environments that provides secure remote access to the industrial networks. Dispel, implemented in Build 2 and Build 4, uses cloud-based virtual desktop interfaces (VDIs) that traverse a cloud-based Enclave to reach a Wicket ESI device that is deployed within the local OT network. Dispel supports both user authentication and authorization, and remote access for Builds 2 and 4.

Virtual Desktop Interfaces (VDIs)

VDIs are Virtual Machines (VMs) that reside in the cloud and allow users to connect using Remote Desktop Protocol (RDP). The VDIs establish a secure connection to the Wicket ESI located in the OT network to provide network access to the OT devices.

Enclave

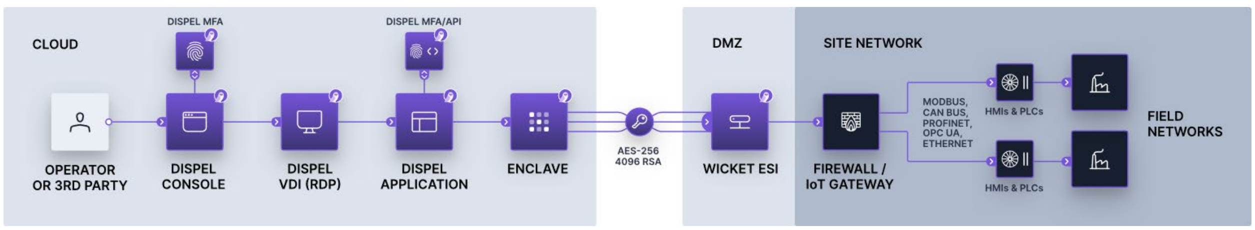
Enclaves are single-tenanted, colorless core, moving target defense (MTD) networks. Enclaves are composed of VMs that act as traffic nodes. To create a shifting target profile, these VMs are steadily replaced by new VMs launched on different hypervisors, in different geographic regions, and/or on altogether different public or private clouds. In the case of Builds 2 and 4, the Enclaves were launched exclusively on public clouds. To provide a static set of IP addresses throughout the builds, the MTD characteristic was disabled.

Wicket ESI

Wicket ESIs are on-premise components, shown in Figure 2-1, that allow users to connect to the OT network remotely. These devices establish encrypted connections from the local OT network up to an Enclave which, in turn, is connected to the VDI, allowing a remote user to access the OT devices.

Additional information is available in *Remote Access for Industrial Control Systems* from Dispel.io at: https://s3.amazonaws.com/downloads.dispel.io/resources/One+Pager/dispel-ics-brochure_20190529.pdf

Figure 2-1 Dispel High-level Implementation, from Remote Access for ICS



2.1.1 Host and Network Configuration

The Wicket ESI is connected to two ports within the DMZ, one for supporting outbound communications to the Dispel Enclave (labeled “WAN”) and one for supporting communication through the local firewall to the ICS environment (labeled “LAN”). The items listed in Table 2-1 are the Wicket ESI specific device and network settings for the hardware provided to support Build 2 Figure B-2 and 4 Figure B-4.

Table 2-1 Dispel Deployment

Name	System	OS	CPU	Memory	Storage	Network
Dispel Wicket ESI	ONLOGIC, ML340G-51	Ubuntu 16.04	Intel i5-6300U	16GB	120GB	Wicket WAN Interface 10.100.1.60 Wicket LAN Interface 10.100.1.61 DMZ
Dispel Enclave	Cloud Virtual Machines	Ubuntu 16.04	Variable	Variable	Variable	N/A
Dispel VDI	Cloud Virtual Machine	Windows Server 2016	Intel Xeon Platinum 8171M	8GB	120GB	N/A

2.1.2 Installation

Installation involves establishing an account on the Dispel cloud-infrastructure and deploying the preconfigured Wicket ESI device within the OT environment. Detailed installation information, customized to the end user's deployment, is provided by Dispel.

After connecting the WAN and LAN network cables, configuring the Wicket ESI required connecting a monitor, keyboard, and mouse to the unit using the available VGA and USB ports. Logging into the unit locally using the credentials provided by Dispel enabled configuration of the network connections using the following procedure (note: these procedures were executed using root privileges and can also be performed using Sudo).

1. Update the network interfaces with the IP configuration information:

   #> vi /etc/network/interfaces

```
source-directory /etc/network/interfaces.d
# LAN
auto enp4s0
allow-hotplug enp4s0
iface enp4s0 inet static
      address 10.100.1.61
      netmask 255.255.255.0
      #gateway
      up route add -net 10.100.0.0 netmask 255.255.255.0 gw 10.100.1.1 dev
enp4s0
      up route add -net 172.16.0.0 netmask 255.255.252.0 gw 10.100.1.1 dev
enp4s0

# WAN
auto enp0s31f6
allow-hotplug enp0s31f6
iface enp0s31f6 inet static
      address 10.100.1.60
      netmask 255.255.255.0
      gateway 10.100.1.1
      dns-nameservers <ip address>
```

2. Update the Wicket ESI netcutter.cfg file to include the local subnet information (toward the bottom of the file):

   #> vi /home/ubuntu/wicket/netcutter.cfg

```
…
subnets = (
  {
    name = "Default";
    value = "10.100.0.0/24";
    advertise = "false";
  },
  {
    name = "PCS";
    value = "172.16.0.0/22";
    advertise = "false";
```

```
  },
  {
    name = "DMZ";
    value = "10.100.1.0/24";
    advertise = "false";
  });
```

3. Restart the Wicket services with the following command:

   #> service wicket restart

4. Check the log for errors and test connectivity to the Dispel environment (note: IP address will be account specific):

   #> tail -f /home/ubuntu/wicket/wicket.log

2.1.3 Configuration

With the Wicket ESI connected to the lab environment, the solution may be configured by establishing an account and configuring the cloud infrastructure, configuring the corporate router/firewall to allow authorized connections to and from the Wicket ESI, and configuring the VDI environment to support the remote access to the ICS environments.

For full documentation and configuration instructions, see the Dispel documentation at https://intercom.help/dispel/en/.

Dispel created an organization named "NCCOE" with an Enclave name "NCCoE-Manufacturing" in their pre-production staging environment. A single "user" account was created for accessing the cloud infrastructure environment named nccoe-m-user@dispel.io. Organizations will need to plan for implementing multiple accounts for supporting the "owner" and "admin" roles in addition to the "user" roles. The "owner" and "admin" roles are for monitoring and managing the cloud infrastructure and are separate from the user accounts used to login to the VDI environment.

The staging environment was configured without the Dispel multifactor authentication (MFA) settings because personal identity verification (PIV) cards were not available as a supported mechanism, and the lab environment did not support authenticator application or security keys. However, MFA is very important for implementation and is strongly encouraged when planning the implementation. For this effort, to reduce the risk of not having the MFA implementation, NCCoE worked with Dispel to limit access to the cloud infrastructure and the VDI instances to only approved source IP addresses. *The additional protection of restricting access to the cloud infrastructure and VDI instances is also encouraged to reduce the risks associated with the internet-accessible web and RDP service*s.

Configure Firewall Settings:

The Wicket ESI needs access to the internet and to the internal OT environment. Table 2-2 below describes the firewall rules implemented on the corporate router/firewall for communications on the internet-facing firewall and internal network zone firewall.

Table 2-2 Firewall Rules for Dispel

Rule Type	Source	Destination	Protocol:Port(s)	Purpose
Allow	10.100.1.60	IdAM: 159.65.111.193 Entry Node: 52.162.177.202	TCP/UDP:1194, HTTPS	Outbound Secure Web to Dispel Environment on the Internet
Allow	10.100.1.61	10.100.1.0/24	ICMP TCP/UDP:RDP, SSH, HTTP/HTTPS, SMB, NTP	PLC Controller Scans
Allow	10.100.1.61	Security Onion 10.100.0.26	TCP:1515 UDP:1514	Build 2: Communication between Wazuh Agent and the server
Allow	10.100.1.61	172.16.0.0/22	TCP:RDP, HTTP/HTTPS	Build 2: Authorized Inbound Communications to PCS Environment
Allow	10.100.1.61	Carbon Black 10.100.0.52	TCP:41002	Build 4: Communication port used between Carbon Black Agent and the server
Allow	10.100.1.61	CRS NAT 10.100.0.20	TCP:48898 UDP:48899	Build 4: Inbound Automation Device Specification (ADS) Protocol for Communication with PLC Device

Notes:

- Dispel's recommended rule for allowing secure shell (SSH)for installation and remote support from the Dispel environment was not enabled for this effort.
- The rules implemented include restricting these outbound ports to Enclave specific IP addresses.
- The Enclave's MTD characteristics were disabled to keep the Enclave's IP addresses static for the duration of the project.

Configure Virtual Desktop Infrastructure (VDI):

The VDI instance is a fully functional workstation/server within the cloud environment. From the VDI instance, authorized users establish a VPN tunnel to the Wicket ESI within the OT environment and then have the access to the environment configured by the device and firewall configurations. In this effort, NCCoE implanted the VDI configuration to support Build 2 and Build 4. The configuration supports the OT environment's jump server configuration (allowing RDP and SSH access to systems within the PCS and CRS environment) and remote engineering workstation (configuring the VDI with the tools needed to support the ICS environment). The configuration for each build is detailed in the following sections:

1. Build 2: PCS Configuration

   a. For the PCS setup, the Dispel VDI was used in a jump server configuration. No additional software was installed. The firewall and Wicket ESI configuration allowed RDP and SSH connections to the PCS ICS environment. Additionally, RDP, SSH, and HTTP/HTTPS access to the Cybersecurity LAN environment was authorized for the remote sessions as defined in the previously described firewall settings, Table 2-2.

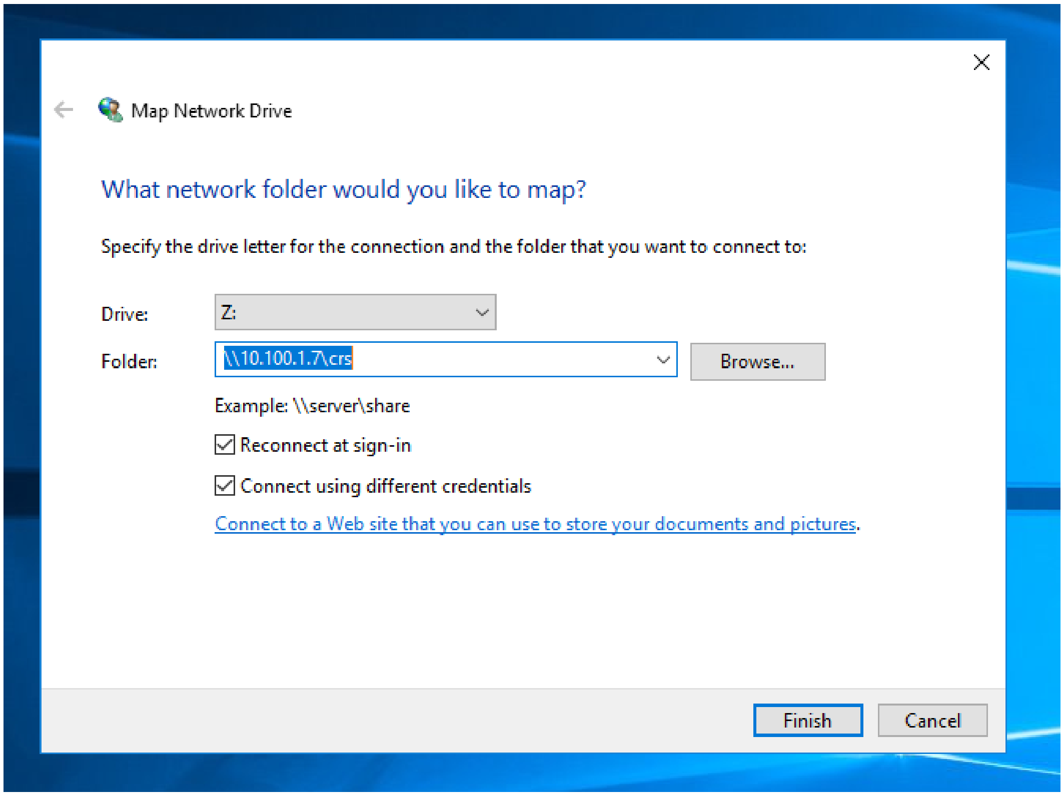
2. Build 4: CRS Configuration
   a. For the CRS setup, the Dispel VDI was configured as a remote engineering workstation. To support the Beckhoff PLC, the TwinCAT 3 XAE software was installed on a VDI, and the network drive provided by the GreenTec-USA solution and hosted in the DMZ environment that contained the PLC code was mapped to the VDI. Additionally, RDP, SSH, and HTTP/HTTPS access to the Cybersecurity LAN environment was authorized for the remote sessions as defined in the previously described firewall settings, Table 2-2.
   b. For the interaction with the Beckhoff PLC, the TwinCAT 3 XAE software (TC31-FULL-Setup.3.1.4024.10.exe) was installed on the VDI.
   c. The Dispel VPN connection does not allow split-tunneling so, once the VPN connection is established from the VDI to the Wicket ESI, the VDI is disconnected from the internet. Therefore, download and installation of software occurred prior to connecting to the Wicket ESI.
   d. Due to the NAT configuration of the RUGGEDCOM RX1510 router between the Cybersecurity LAN and the CRS environment, port forwarding rules were configured to allow external traffic to reach the Beckhoff CX9020 PLC.
   e. The following rules (Table 2-3) were created in the RX1510 firewall to enable destination network address translation (DNAT) from the firewall WAN interface (10.100.0.20) to the CRS PLC (192.168.0.30)

Table 2-3 Firewall Rules

Rule Type	Source	Destination	Destination Port(s)	Purpose
DNAT	10.100.1.61	192.168.0.30	UDP:48899	DNAT (10.100.0.20) - Beckhoff ADS discovery protocol used by the TwinCAT 3 software to discover ADS devices.
DNAT	10.100.1.61	192.168.0.30	TCP:48898	DNAT (10.100.0.20) - Beckhoff ADS protocol used by the TwinCAT 3 software to com-municate with the PLC.

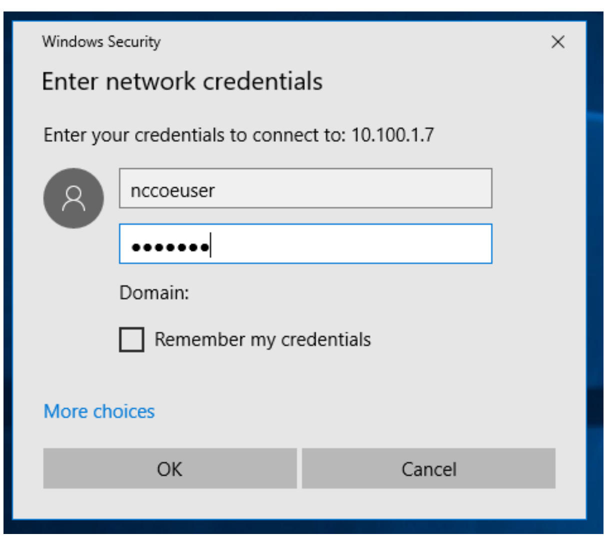
3. As described in 2.i above, the GreenTec WORMdisk (\\10.100.1.7\crs) was mapped to the VDI to access the PLC code. The configuration to map Windows is shown in Figure 2-2 below:

Figure 2-2 Mapping a Network Drive



4. After clicking **Finish**, the user is prompted for credentials, as shown in Figure 2-3. An account authorized to access the network drive must be used. This is separate from the Dispel VDI credentials.

Figure 2-3 Authentication to File Server



2.2 Dragos

The Dragos platform implementation in Build 3 consists of two physical servers hosting the Dragos SiteStore and the Dragos sensor to meet the behavioral anomaly detection (BAD), hardware modification, firmware modification, and software modification capabilities. Dragos utilizes a combination of a passive sensor and integration with the OSIsoft PI Server to monitor critical networks for anomalies. OSIsoft PI performs active querying to retrieve information about endpoints in the CRS environment, which is shared with Dragos.

2.2.1 Host and Network Configuration

Dragos is installed and configured to support the CRS Environment in Build 3. The overall build architecture is shown in Figure B-3, and the Dragos specific components are listed in Table 2-4.

Table 2-4 Dragos Deployment

Name	System	OS	CPU	Memory	Storage	Network
VMware Server	Dell OEMR R740	VMware 6.7.0 Update 3	2x Intel 6130 CPU	384 GB	2x 1.5TB Mirror 6x 8TB RAID 10	Testbed LAN 10.100.0.62/24
Dragos Server	VMware	CentOS 7	48x vCPU	192 GB	215 GB 10 GB 1.5 TB 1.5 TB	Testbed LAN 10.100.0.63/24
Dragos Sensor	Dell OEM	CentOS 7	64x vCPU	128 GB	240 GB 1 TB	Testbed LAN 10.100.0.64/24

2.2.2 Installation

The Dragos platform, which includes the SiteStore server and the Dragos sensor, was delivered as pre-configured hardware appliance by Dragos with the required IP addresses already assigned. The only installation step was correctly connecting the server and the sensor management ports to the Testbed LAN and adding the switch port analyzer (SPAN) port connection to the sensor.

The Dragos Platform Administrator Guide and Dragos Platform User Guide for Release 1.7 were used to guide the installation. Customers can obtain these guides from Dragos.

2.2.3 Configuration

In addition to the standard configuration preset by Dragos, the Dragos Platform was configured to work with OSIsoft PI for alerting on certain conditions.

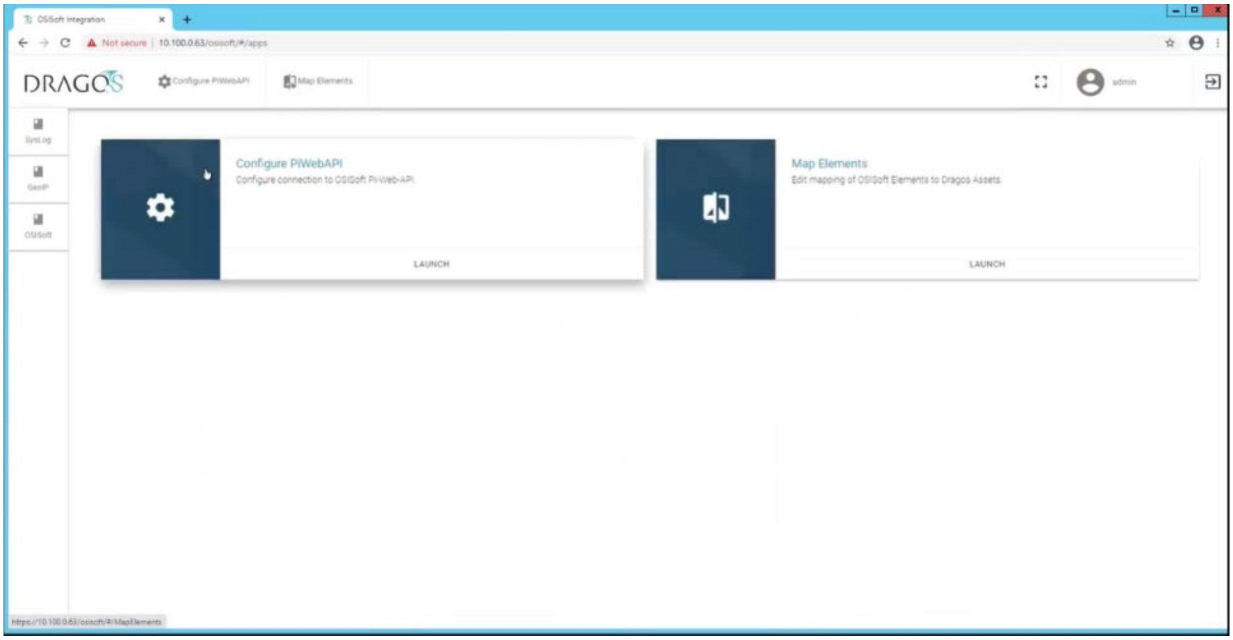
Configure the Dragos SiteStore Server:

1. Configure the data connection between Dragos SiteStore and OSIsoft PI Server:

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

a. Once installation is successful, open a browser to access the configuration screen by using the URL **https://<SiteStore ip address>/osisoft/#/apps**. (Figure 2-4)

Figure 2-4 Dragos OSIsoft PI Server Integration



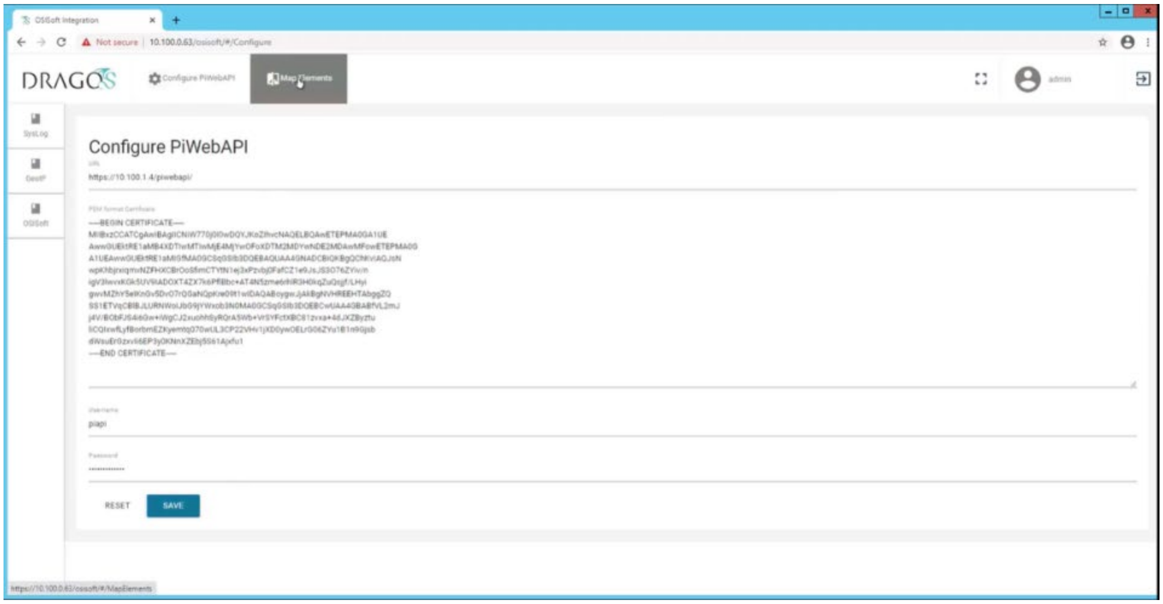
b. Click **Configuration Pi Web API** to open a screen for filling out the required information, including privacy enhanced mail (PEM) format certificate and password for secure authentication (Figure 2-5).

   i. Upload the server public key for the HTTPS certificate.

   ii. Specify the user credentials for the OSIsoft PI Web API interface.

   iii. Click **Save**.

Figure 2-5 Dragos PI Web API Configuration



c. Click **Map Elements** to access the interface to pair elements between OSIsoft PI Server and the Dragos Platform assets. Here, the PLC in **OSIsoft Elements** panel is paired with Beckhoff asset in the Dragos Platform asset (Figure 2-6).

   i. Select the OSIsoft Database **CRS-backup** on the left side to access the devices list from the Historian Database.

   ii. Select the **Default NetworkID RFC 1918** and use the Filer options to find specific assets.

   iii. For each asset in the OSIsoft Database, select the corresponding asset in the Dragos asset repository and click **Pair Selected**.

   iv. Repeat this process for each asset until all paired assets are listed in the **Paired Data** table (Figure 2-7).

      1) PLC paired to 192.168.0.30

      2) Station 1 paired to 192.168.1.101

      3) Station 2 paired to 192.168.1.102

      4) Station 3 paired to 192.168.1.103

      5) Station 4 paired to 192.168.1.104

Figure 2-6 OSIsoft PI Server to Dragos Asset and Data Pairing

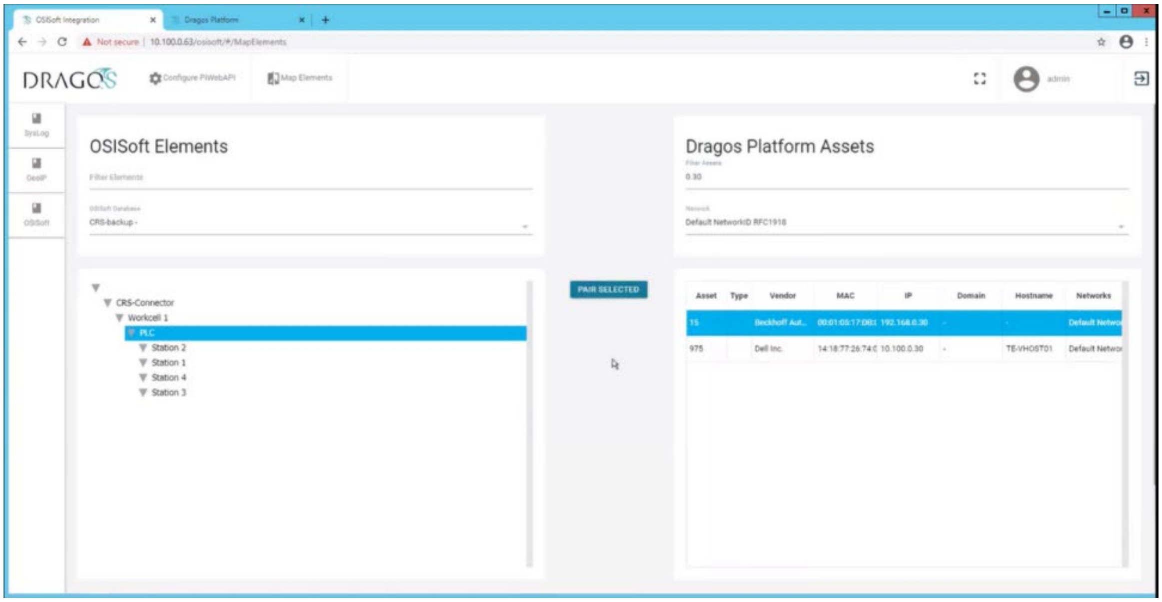


Figure 2-7 OSIsoft PI Server and Dragos Paired Data Elements

Paired Data

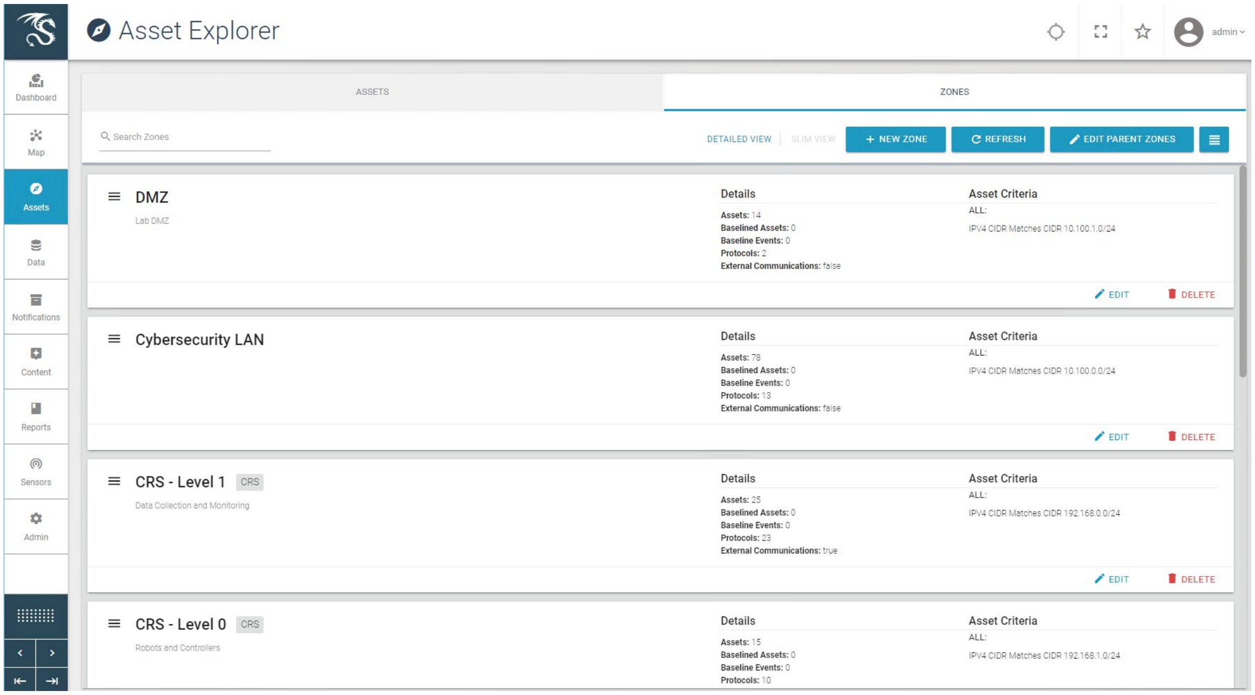
Delete	Asset	OSIsoft Name	Type	Vendor	MAC	IP	Domain
	15	PLC		Beckhoff Automation GmbH	-	192.168.0.30	-
	3176	Station 2			B0:D5:CC:FE:6E:B1	(2) 192.168.1.102, FE80::B2D5:CCFF:FEFE:6EB1	(2) machining-station-2.local, _tcp.local
	3186	Station 1			B0:D5:CC:FA:70:C9	(2) 192.168.1.101, FE80::B2D5:CCFF:FEFA:70C9	(2) machining-station-1.local, _tcp.local
	3180	Station 3			B0:D5:CC:FA:7A:43	(2) 192.168.1.103, FE80::B2D5:CCFF:FEFA:7A43	(2) machining-station-3.local, _tcp.local
	3177	Station 4			B0:D5:CC:F4:26:EC	(2) 192.168.1.104, FE80::B2D5:CCFF:FEF4:26EC	(2) _tcp.local, machining-station-4.local

2. Configure Zones

   NOTE: Zones are ordered in a similar manner to firewall rules. In other words, higher rules have priority over lower rules.

   a. Click **Assets** and select the **Zones** tab (Figure 2-8).

Figure 2-8 Dragos Zone Administration Page



b. Click **+ New Zone** (Figure 2-9) and define the following zones:

i. Name: **DMZ:**

1) Description: Lab DMZ
2) Zone Criteria (Match ALL):
a) IPV4 CIDR Matches CIDR 10.100.1.0/24

ii. Name: Testbed LAN:

1) Description: Lab Testbed LAN

2) Auto Zone Criteria (Match ALL):

a) IPV4 CIDR Matches CIDR 10.100.0.0/24

iii. Name: CRS:

1) Description: **Parent CRS**

2) No Criteria

iv. Name: CRS – Level 0:

1) Description: Robots and Controllers

2) Parent Zone: **CRS**

3) Auto Zone Criteria (Match **ALL**):

a) IPV4 CIDR Matches CIDR 192.168.1.0/24

v. Name: CRS – Level 1:

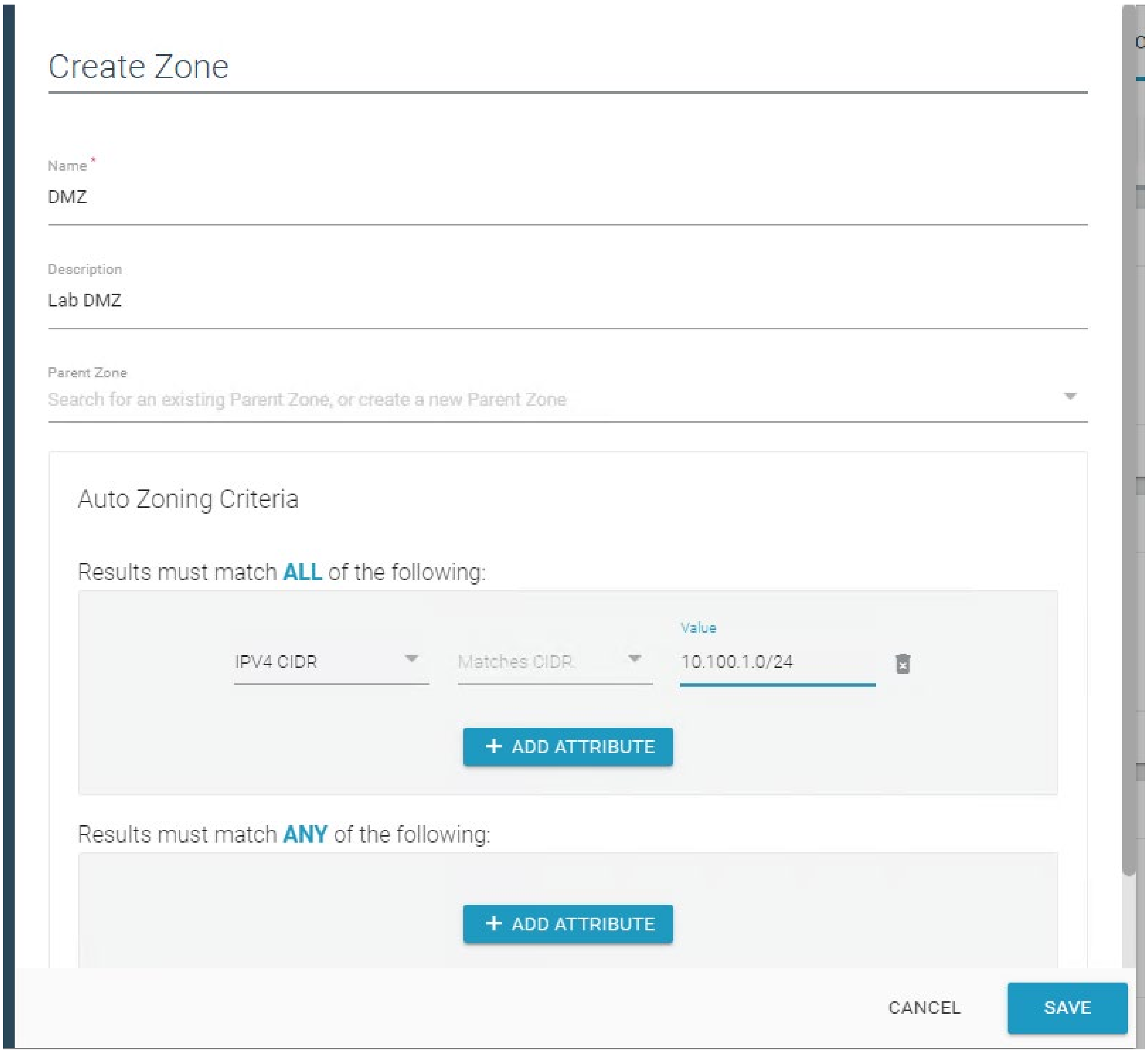
1) Description: **Lab DMZ**

2) Parent Zone: **CRS**

3) Auto Zone Criteria (Match **ALL**):

a) IPV4 CIDR Matches CIDR 192.168.0.0/24

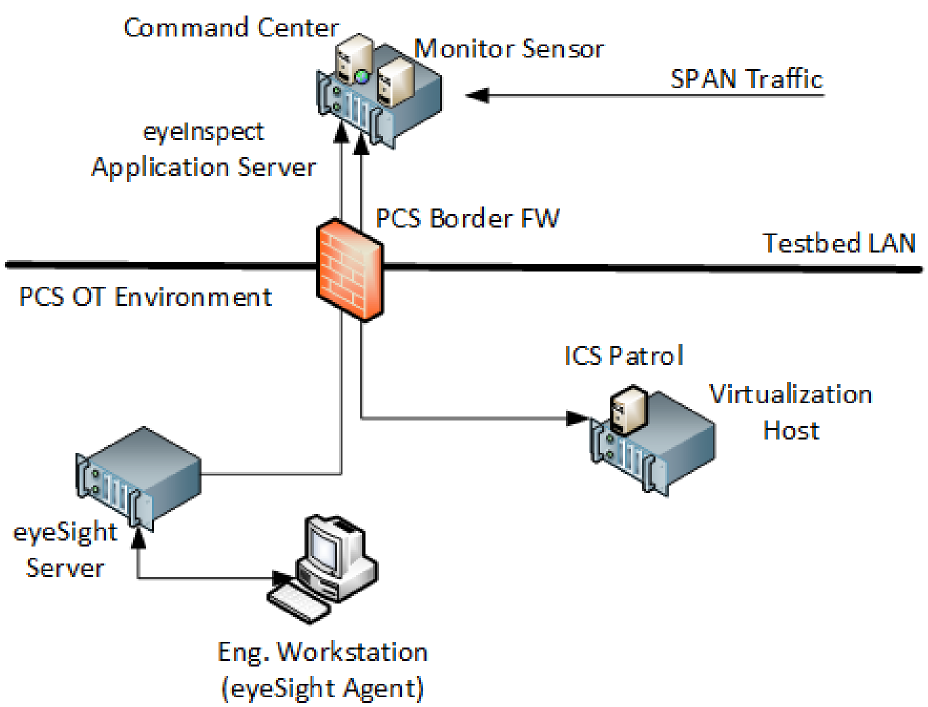
Figure 2-9 Dragos Create Zone Pop-up



2.3 Forescout Platform

The Forescout products included in the practice guide are eyeInspect (formally SilentDefense), eyeSight, ICS Patrol, and Forescout Console. These products are utilized in Build 2 to meet the BAD, hardware modification, firmware modification, and software modification capabilities. The Forescout implementation utilizes different components and modules installed on different devices to monitor critical networks for anomalies and active query capabilities to retrieve information about endpoints in the PCS environment. A high-level of the key server and agent components is presented in Figure 2-10.

Figure 2-10 Forescout High-Level Components and Dataflows



eyeInspect (formally SilentDefense)

The eyeInspect (Version 4.1.2) control server and monitoring sensor are installed on a single appliance with a management interface on the Testbed VLAN and network monitoring capabilities through a dedicated SPAN port. The SPAN port provides passive monitoring for network-based anomalies and retrieves information about endpoints within the network. The eyeInspect appliance also serves as the command center for supporting the ICS Patrol and eyeSight components.

eyeSight

Forescout eyeSight (Version 8.2.1) provides enhanced network monitoring and response using an agent installed on endpoints. In this build, eyeSight instances are configured through the Forescout Console to provide additional monitoring and reporting information to eyeInspect.

ICS Patrol

Forescout ICS Patrol (Version 1.1.2-4.a826b94) is a sensor that supports active queries for ICS devices to obtain status and other information such as hardware configuration and firmware version. ICS Patrol queries and reporting results are managed through eyeInspect.

Forescout Console

The Forescout Console (Version 8.2.1) is a Java-based application for configuring and managing eyeSight and eyeSight agents. The Forescout Console is installed on a computer with network access to the eyeSight server.

2.3.1 Host and Network Configuration

Forescout was installed and configured to support the PCS Environment as part of Build 2. The overall build architecture is provided in Figure B-2 with the Forescout specific components in Table 2-5 and the eyeSight agents in Table 2-6.

Table 2-5 Forescout Deployment

Name	System	OS	CPU	Memory	Storage	Network
eyeInspect control server	Dell Embedded Box PC 5000	Ubuntu 16.04	Intel i7-6820EQ	32 GB	250 GB	Testbed LAN 10.100.0.65
Forescout Console	Hyper-V VM	Windows 2012R2	2x vCPU	6 GB	65 GB	Testbed LAN 10.100.0.25
eyeSight Server	Dell R640	Ubuntu 16.04.06	Intel Xeon Silver 4110	32	600 GB	PCS VLAN 2 172.16.2.61
ICS Patrol	VirtualBox VM	Ubuntu 16.04.06	2x vCPU	2 GB	40 GB	PCS VLAN 2 172.16.2.62

For the lab environment, network connectivity between the components in the Testbed LAN and the components in the PCS environment required the following persistent route configured on Testbed LAN systems:

```
route -p ADD 172.16.0.0 MASK 255.255.252.0 10.100.0.40
```

The following systems were configured to utilize the eyeSight Agents.

Table 2-6 eyeSight Agent Deployment

Name	System	OS	CPU	Memory	Storage	Network
Engineering Workstation	Dell T5610	Windows 7	Intel i5-4570	16 GB	465 GB	PCS VLAN 3 172.16.3.10
HMI Host	Generic	Windows 7	Intel i5-4590	8 GB	233 GB	PCS VLAN 1 172.16.1.4

Additional details for Build 2 are available in Section 4.5 of Volume B.

2.3.2 Installation

The Forescout products included in the practice guide are eyeInspect, Forescout Console, ICS Patrol, and eyeSight. These products are installed as indicated in the appropriate subsection below. To support these components, the PCS Gateway/Firewall rules were updated as follows (Table 2-7).

Table 2-7 Firewall Rules for Forescout

Rule Type	Source	Destination	Port(s)	Purpose
Allow	10.100.0.65	172.16.2.61	22 (ssh) 9999 9092	System Management eyeInspect Data eyeInspect Data
Allow	10.100.0.65	172.16.2.62	22 (ssh) 9001	System Management eyeInspect Data

2.3.2.1 eyeInspect

eyeInspect is an appliance hosted on a Dell Embedded Box PC 5000. The unit was placed within a standard datacenter rack unit with the eyeSight appliance and connected to the network as described in Section 2.3.1. SPAN ports from the DMZ, Testbed LAN, and PCS VLAN 1, 2, and 3 switches were routed to the appliance for passive network monitoring. Installation also required uploading the license file after successfully logging onto the appliance.

2.3.2.2 Forescout Console

Forescout Console was installed following the standard installation procedures. Instructions can be found in the Forescout Installation Guide Version 8.2.1 available at https://docs.forescout.com. The software is available from https://forescout.force.com/support/s/downloads, where current and past versions are available. Login credentials were provided by Forescout.

2.3.2.3 eyeSight

Forescout eyeSight is an appliance hosted on a 1U Dell R640 that is installed within a standard datacenter rack and connected to the network as described in the previous section.

2.3.2.4 eyeSight SecureConnector Agent

1. In a browser on a system with web connectivity to the eyeSight server, navigate to https://172.16.2.61/sc.jsp to access the SecureConnector download page (Figure 2-11) and follow these steps:

    a. Select Create SecureConnector for: **Windows**.

    b. Enable **Show the SecureConnector icon on the endpoint systray**.

    c. Select **Install Permanent As Service.**

    d. Click **Submit.**

2. Download the Forescout Agent (Figure 2-12):

    a. Select Version **Win64.**

    b. Click **Download.**

3. Install the downloaded agent on the target systems using an administrator account.

Figure 2-11 Forescout SecureConnector Distribution Tool

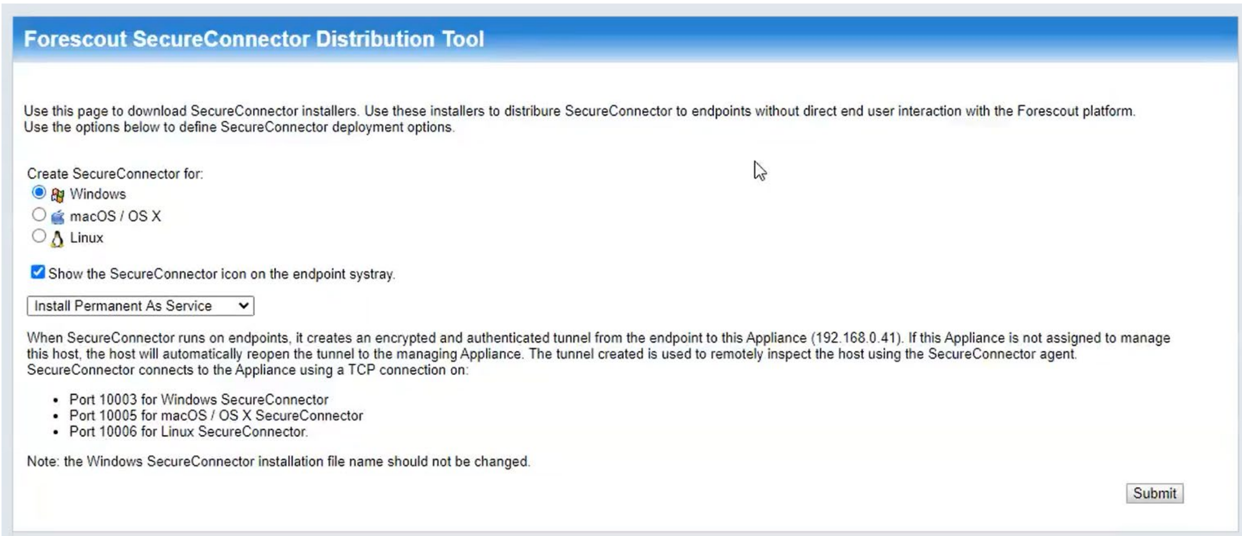
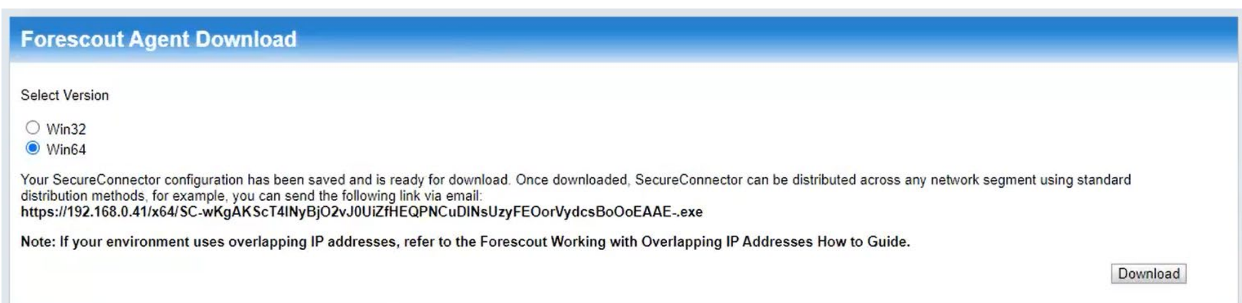


Figure 2-12 Forescout Agent Download



2.3.2.5 ICS Patrol

Forescout ICS Patrol (Version 1.1.2-4.a826b94) is a sensor that is deployed on an existing VirtualBox host in the PCS environment. Ubuntu 16.04.06 is required for proper installation and can be downloaded from http://old-releases.ubuntu.com/releases/xenial/ubuntu-16.04.6-server-amd64.iso. Install the operating system on a VM connected to PCS VLAN 2 following the procedures from the Silent Defense Installation and Configuration Guide 4.1.2 document Section 2.2.2, Installing the Linux Ubuntu OS.

1. Install the ICS Patrol Component from the Silent Defense Installation and Configuration Guide 4.1.2 document Sections 2.2.4 and 2.2.5 following these steps:

    a. Establish an SSH session to the eyeInspect appliance.

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

b. Copy the components to the ICS Patrol VM:

```
$ scp os_provisioning_4.1.1_install.run \
main_configuration_4.1.1_install.run \
silentdefense@172.16.2.62:/home/silentdefense
```

c. SSH to the ICS Patrol VM and execute the installation components:

```
$ chmod a+x *.run
$ sudo ./os provisioning 4.1.1 install.run
$ sudo ./main_configuration_4.1.1_install.run
$ sudo reboot
```

2.3.3 Configuration

The eyeSight agents and ICS Patrol do not require specific configurations.

2.3.3.1 eyeInspect

1. Access the eyeInspect web interface and log in with an administrator account.
2. Register the local sensor for SPAN traffic monitoring:
   a. Click the **Sensors** tab to access the Sensor Admin/Overview Page (Figure 2-13).
   b. Click **Add > SilentDefense sensor**.
   c. Specify the sensor parameters in the dialog box (Figure 2-14).

Figure 2-13 eyeInspect Sensor Admin/Overview Page – Add Sensor

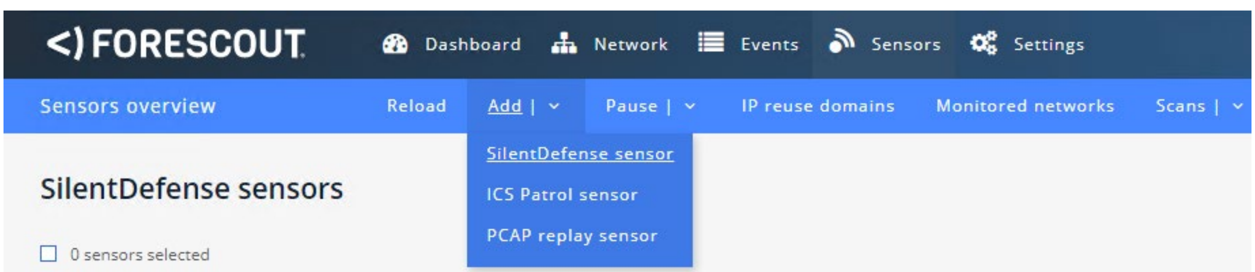
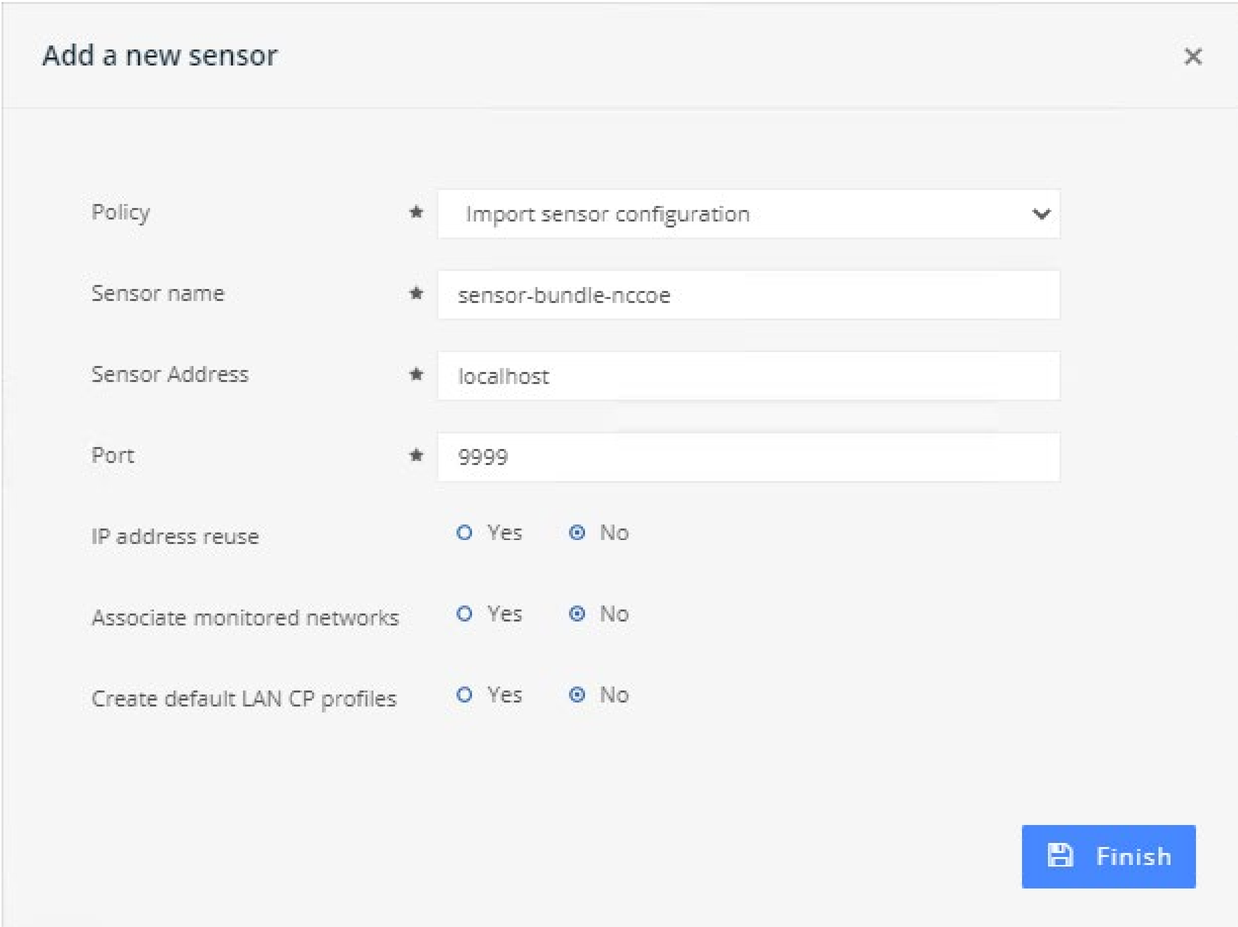
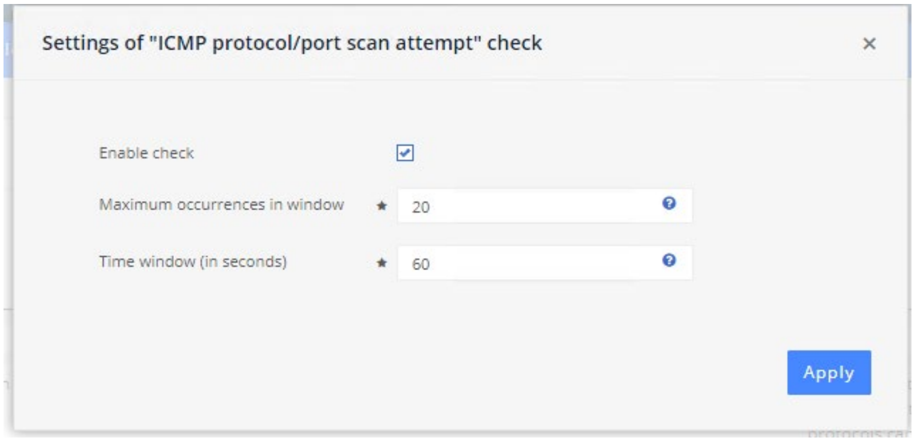


Figure 2-14 Adding a New SilentDefense Sensor Dialog



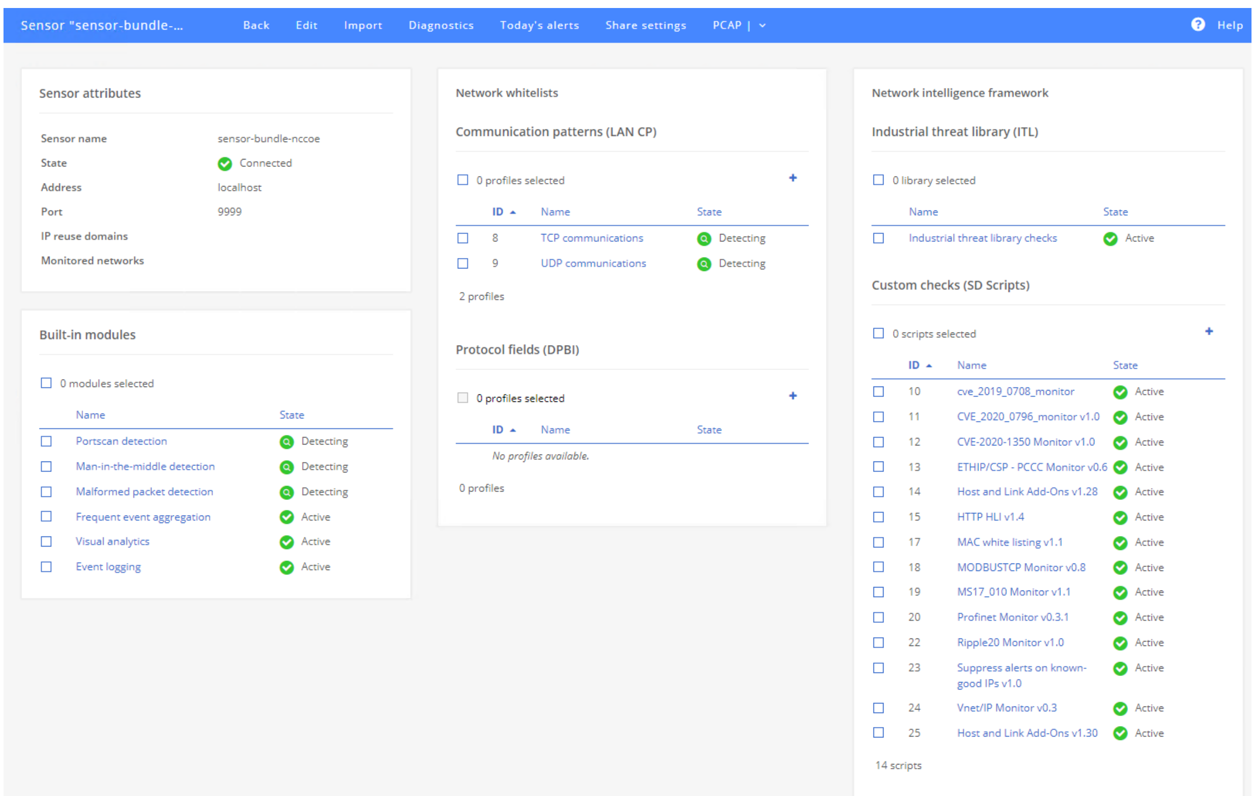
3. Adjust Passive Monitoring settings:
    a. From the Dashboard, click **Sensors**.
    b. Select the **SilentDefense Sensor** from the list of available sensors.
    c. Click the **Industrial Threat Library Overview** option in the upper right corner.
    d. Click the **Security** menu option on the left under **Checks by Category**.
    e. Enter "ICMP" in the Search field to reduce the list of available options.
    f. Click the **ICMP** protocol/port scan attempt to open the settings dialog box (Figure 2-15) and verify the following settings:
        i. Verify **Enable Check** is selected.
        ii. Verify **Maximum occurrences in window** is set to **20**.
        iii. Verify **Time Window (in seconds)** is set to **60**.

Figure 2-15 eyeInspect ICMP Protocol/Port Scan Attempt Settings



g. Select **Portscan Detection** under Built-in Modules (Figure 2-16).

Figure 2-16 eyeInspect Sensor Configuration Options



h. Click the **Settings** tab and set the following parameters (Figure 2-17):

i. **Sensitivity level:** User defined

ii. **Number of Hosts with failed connections to make a distributed scan:** 10

iii. **Detect SYN scans**: Checked

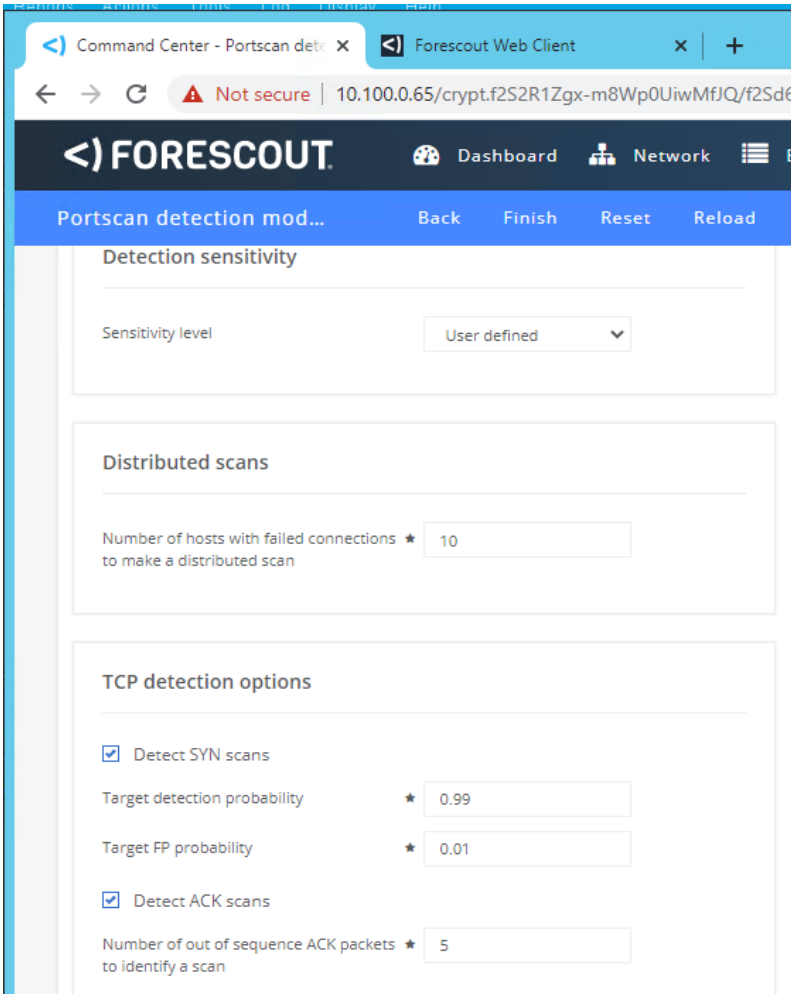
iv. **Target detection probability**: 0.99

v. **Target FP probability**: 0.01

vi. **Detect ACK scans**: Checked

vii. **Number of out of sequence ACK packets**: 5

Figure 2-17 eyeInspect Portscan Detection Settings

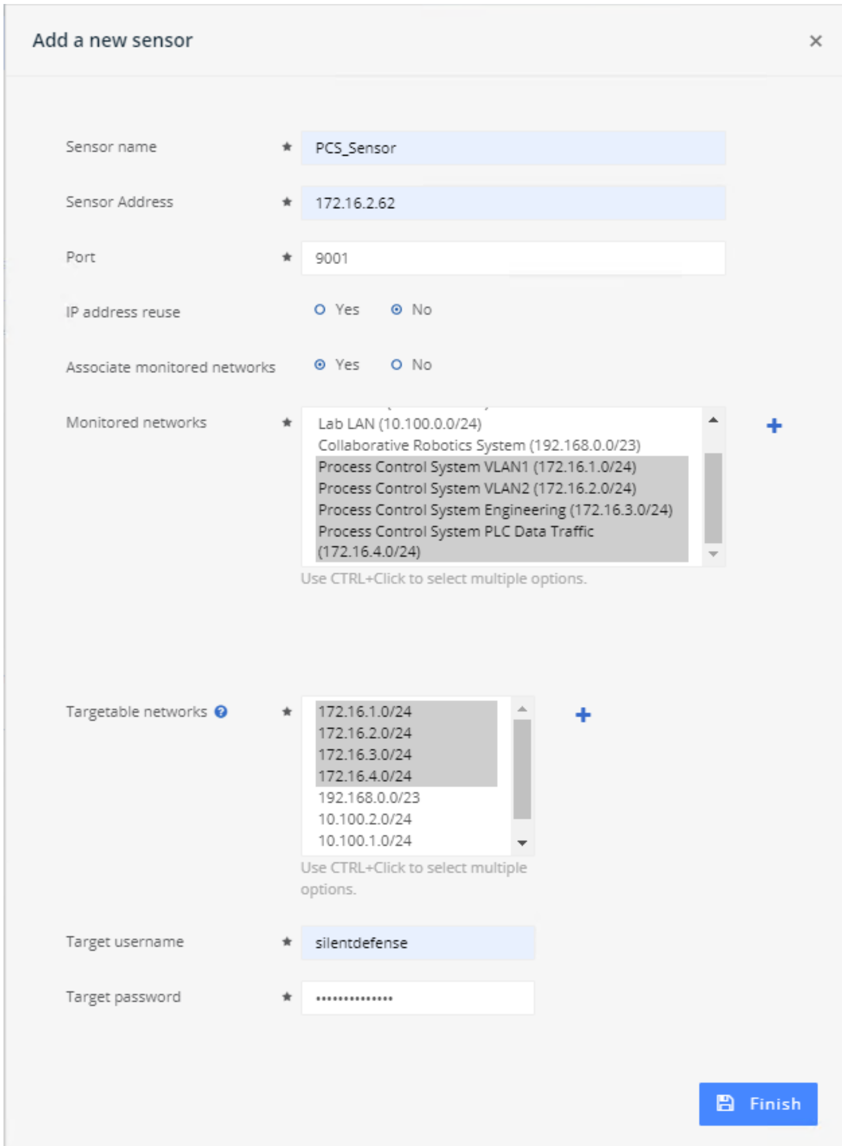


4. Register the ICS Patrol Sensor:

   a. From the Sensor admin page, click **Add > ICS Patrol sensor**.

   b. Specify the sensor parameters in the dialog box (Figure 2-18).

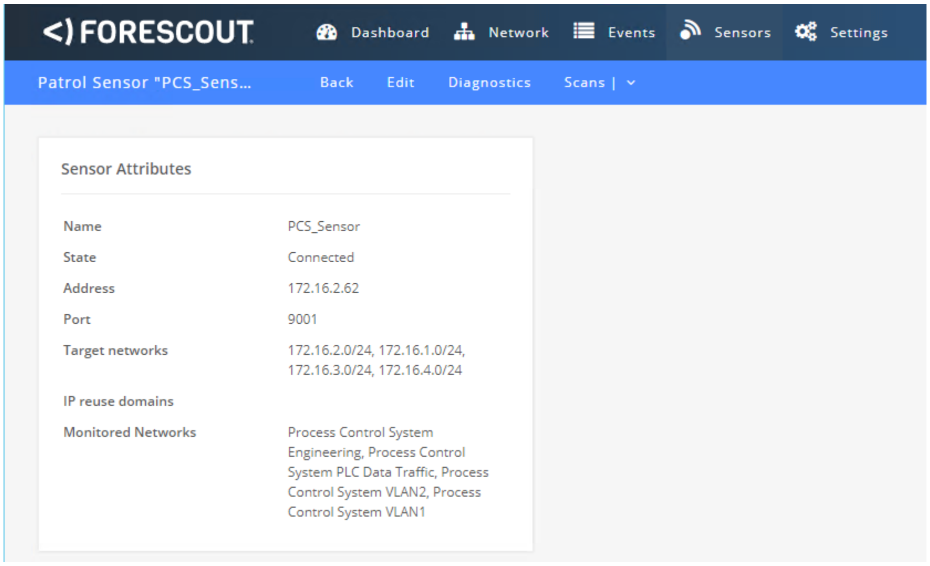
Figure 2-18 Add ICS Patrol Sensor Dialog



c. Define a scan policy to periodically check the PCS PLC to monitor for changes.

i. Click the PCS Sensor created in the previous step to open the sensor admin page (Figure 2-19).

Figure 2-19 ICS Patrol Sensor Admin Page



ii. Click **Scans > Scan Policies.**

iii. In the dialog option (Figure 2-20) enter the scanning parameters:

1) **Name**: PCS PLC

2) **Scan Type**: EtherNet/IP

3) **Target Type**: Custom target

4) **IP address reuse**: No

5) **Network Address**: 172.16.2.102

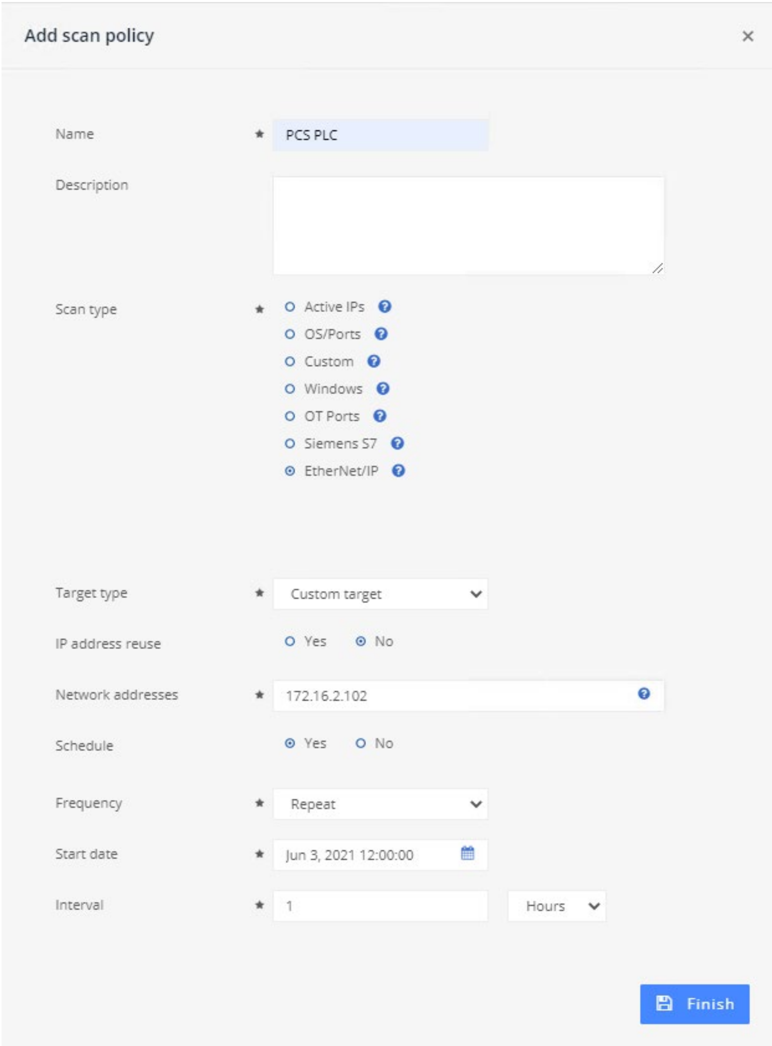
6) **Schedule**: Yes

7) **Frequency**: Repeat

8) **Interval:** 1 . Select "Hours" from the drop-down menu.

9) Click **Finish**.

Figure 2-20 Add an ICS Patrol Scan Policy



2.3.3.2 eyeSight

Using the Forescout Console application, users may configure, monitor, and manage the eyeSight appliance and agents. The Forescout Console is also used to test and verify connectivity to the eyeInspect server.

1. Login to the Forescout Console.
2. Select the Gear Icon in the upper right corner or the **Tools > Option** menu item to bring up the Options display.
3. Enter "Operational" in the search bar.
4. Select the **Operational Technology** tab on the left side of the screen to display the current settings.
5. Select the IP entry for the Command Center and select **Add** to start the workflow process.

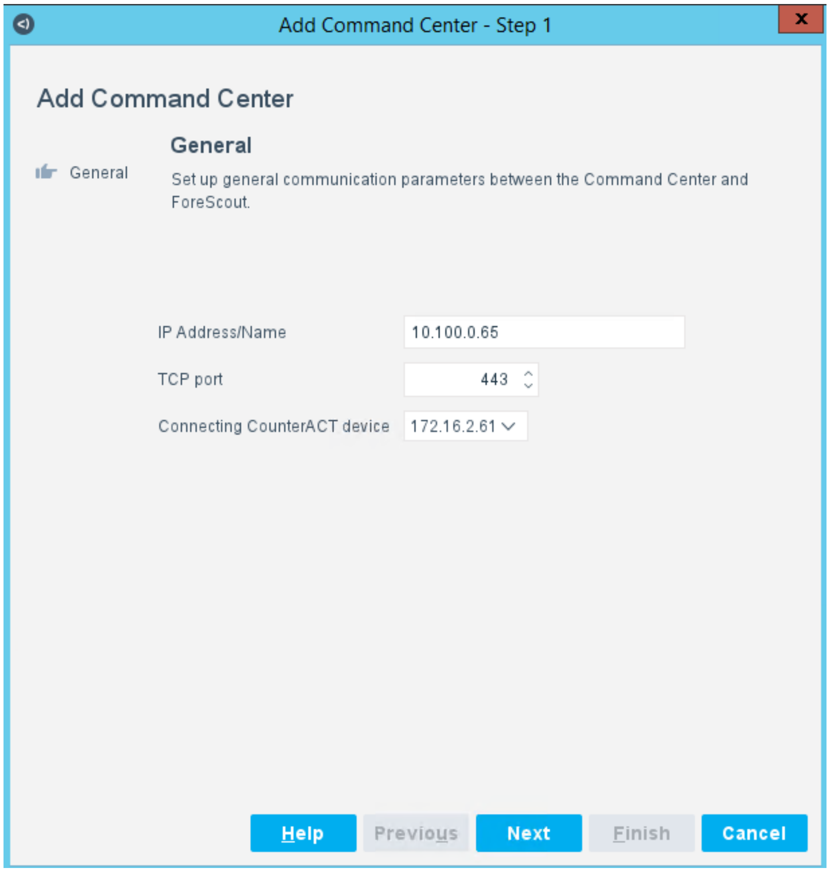
a. Specify General Information (Figure 2-21):

   i. Enter the Command Center IP Address "10.100.0.65" for IP Address/Name.

   ii. Select "172.16.2.61" from **the Connecting CounterAct device** drop-down menu.

   iii. Select "443" from the TCP Port drop-down menu.

Figure 2-21 eyeSight Add Dialog – General Information

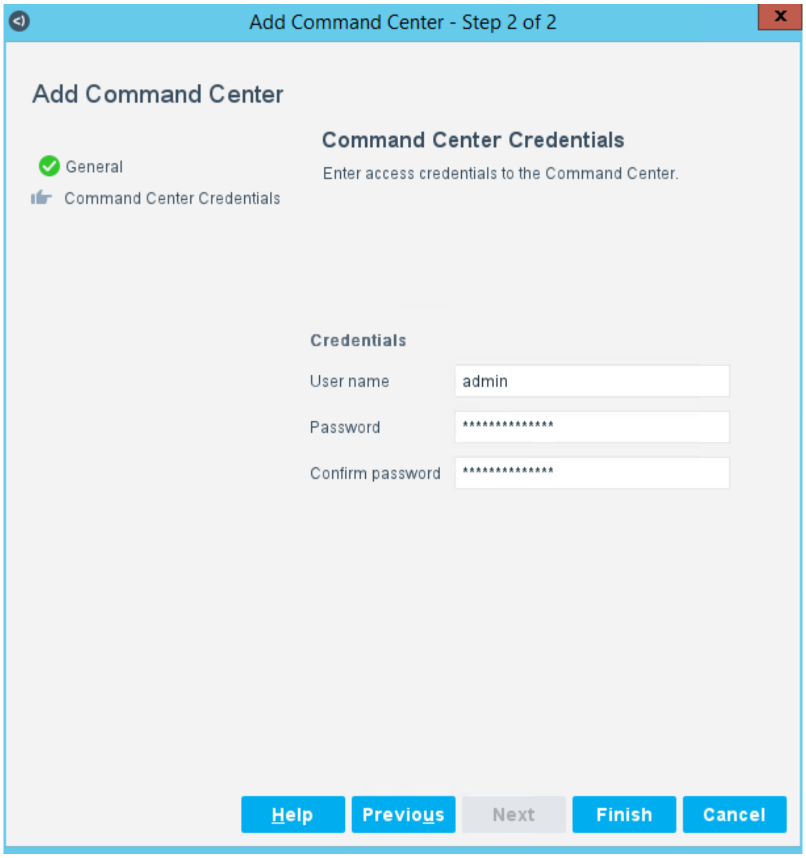


b. Click **Next**.

c. Enter the command center credentials (Figure 2-22).

d. Click **Finish**.

Figure 2-22 eyeSight Add – Command Center Credentials



6. Select the IP address for the Command Center and Click **Test** (Figure 2-23). If the connection is successful, a message like the one shown in Figure 2-24 displays.
7. Click **Apply** to save the changes.
8. Click **Close** to close the message.

Figure 2-23 eyeSight OT Settings

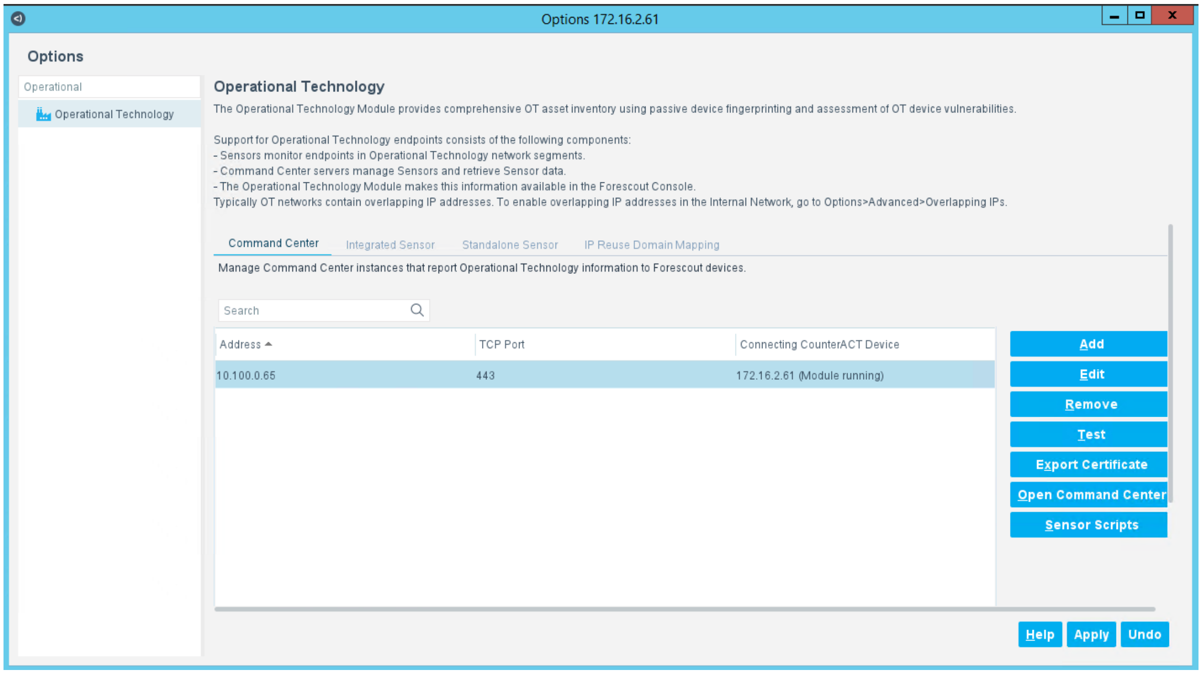
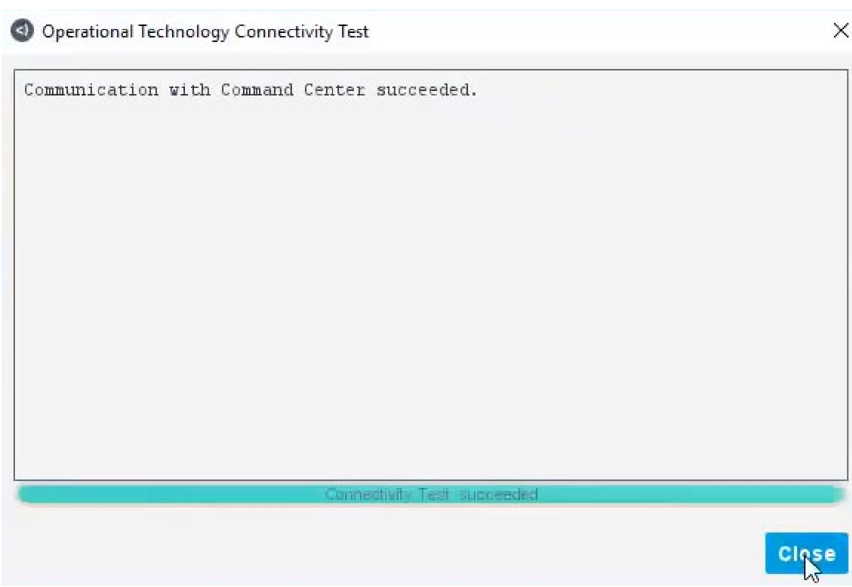


Figure 2-24 eyeSight Test Connection Successful Message



2.4 GreenTec-USA

The GreenTec-USA products included in this practice guide are the ForceField and WORMdisk zero trust storage devices. These products were utilized in Builds 1, 2, 3, and 4 to meet the File Integrity Checking capability by storing and protecting critical PCS and CRS data from modification and deletion.

ForceField

A ForceField hard disk drive (HDD) provides a protected write-once-read-many data storage location for historian data backups and database backups. Data is immediately protected as it is written to the HDD in real time, permanently preventing the data from modification and deletion.

WORMdisk

A WORMdisk HDD provides a protected data storage location for PLC logic, device firmware, and approved software applications for use in the manufacturing environment. Data is protected by "locking" individual partitions of the HDD using a software utility, permanently preventing the data from modification and deletion.

2.4.1 Host and Network Configuration

The WORMdisk and ForceField HDDs were installed in a rack-mount server appliance provided by GreenTec-USA and described in Table 2-8. The overall build architectures utilizing this appliance and devices are described in Section 4.5 in Volume B.

Table 2-8 GreenTec-USA WORMdrive and ForceField Deployment

Name	System	OS	CPU	Memory	Storage	Network
GreenTec-USA Server	Supermicro x8 Series Server	Ubuntu 18.04	2x Intel Xeon E5620	16 GB	750 GB OS 1.0 TB WORMdisk 1.0 TB ForceField	DMZ 10.100.1.7

2.4.2 Installation

The ForceField and WORMdisk HDDs were hosted on a hardware appliance provided by GreenTec-USA. The unit was placed within a standard datacenter rack unit and connected to the network as shown in Figure B-1, Figure B-2, Figure B-3, and Figure B-4.

Full documentation and installation guides are provided to customers by GreenTec-USA.

NIST chose to utilize Samba as the network file sharing protocol due to the prevalence of Windows and Linux workstations within the testbed. The GreenTec-USA appliance did not come with Samba pre-installed, so installation was performed via the Ubuntu Advanced Packaging Tool and the Ubuntu package repository.

NOTE: GreenTec-USA typically provides turnkey server storage solutions. Installation and configuration of file sharing packages and other software will likely not be required.

NOTE: Many of the commands used to manage the ForceField and WORMdisk HDDs must be executed by a user with superuser privileges or as the root user.

1. Add the default gateway so the appliance can communicate to other devices on the network using the following command:

   ```
   $ sudo route add default gw 10.100.1.1
   ```

2. In a terminal window on the GreenTec-USA appliance, execute these commands:

```
$ sudo apt update
$ sudo apt -y install samba
$ sudo ufw allow samba
```

2.4.3 Configuration

The appliance provided by GreenTec-USA for this project was preconfigured with the ForceField HDD as device `/dev/sdc` and the WORMdisk HDD as device `/dev/sdb`.

2.4.3.1 ForceField HDD

The ForceField HDD is configured as a mounted volume, allowing the drive to be used as a typical HDD by using native operating system commands.

1. Create a mount point (empty directory) for the ForceField HDD using the following command:

```
$ sudo mkdir /mnt/forcefield
```

2. Start the ForceField WFS volume manager to mount the drive using the following command:

```
$ sudo /opt/greentec/forcefield/bin/wfs /dev/sdc /mnt/forcefield/
```

2.4.3.2 WORMdisk HDD

The WORMdisk is divided into 120 partitions to enable periodic updates and revisions to the protected data (i.e., data in the "golden" directory). Once a partition is locked it cannot be modified, so the next sequential partition on the drive is used as the new "golden" directory.

1. Format the WORMdisk with 120 partitions (NOTE: this operation must be performed from the command line as administrator on a computer with the Microsoft Windows OS) using the following command:

```
> gt_format.exe 1 /parts:120
```

2. In the Ubuntu OS, create the mountpoint for the WORMdisk HDD partition using the following command:

```
$ sudo mkdir /mnt/golden
```

3. Add a persistent mount to the /etc/fstab file:

```
$ sudo echo "/dev/sdb2 /mnt/golden fuseblk
rw,nosuid,nodev,relatime,user_id=0,group_id=0,allow_other,blksize
=4096 0 0" >> /etc/fstab
```

4. Create a directory structure within the "golden" directory and copy approved files into those directories (e.g., PLC logic, device firmware, approved software).

5. Once all files have been copied and verified, lock the partition to protect the data:

```
$ sudo /greentec/Ubuntu/wvenf /dev/sdb2
```

When it is time to create a new "golden" partition, the partition names in the /etc/fstab file must be updated to point to the correct partition. The following instructions provide an example process to update the files and increment the golden partition from `/dev/sdb2 to /dev/sdb3.`

1. On the GreenTec-USA appliance, create a temporary directory, mount the folder to the next unlocked WORMdisk partition, and copy existing "golden" files to the temporary directory:

```
$ sudo mkdir /mnt/tmp
$ sudo mount /dev/sdb3 /mnt/tmp
$ sudo cp -R /mnt/golden /mnt/tmp
```

2. Update the files and folders in the temporary directory, `/mnt/tmp,` as desired.

3. Unmount the temporary directory and lock the partition:

```
$ sudo umount /mnt/tmp
$ sudo /greentec/Ubuntu/wvenf /dev/sdb3
```

4. Stop the Samba service:

```
$ sudo systemctl stop smb.service
```

5. Unmount the golden partition:

```
$ sudo umount /mnt/golden
```

6. Modify the /etc/fstab file with the new partition name and save the file:

```
/dev/sdb3 /mnt/golden fuseblk
rw,nosuid,nodev,relatime,user_id=0,group_id=0,allow_other,blksize
=4096 0 0"
```

7. Re-mount all partitions, start the Samba service, and remove the temporary directory:

```
$ sudo mount -a
$ sudo systemctl stop smb.service
$ sudo rmdir -r /mnt/tmp
```

2.4.3.3 Samba

1. Add local user accounts to the appliance for accessing the network file shares and create a password:

```
$ sudo adduser nccoeuser
$ sudo smbpasswd -a nccoeuser
```

2. Open the `file /etc/samba/smb.conf` and add the following content to the end of the file to create the individual shares:

```
# GreenTec-USA ForceField Share
strict sync=no

# OSIsoft PI historian and database backups
   [ForceField]
```

```
 browsable = yes
 guest ok = no
 path = /mnt/forcefield
 read only = no
 writeable = yes
 case sensitive = yes

# GreenTec-USA Golden WORMDisk Share
[golden]
 browsable = yes
 guest ok = no
 path = /mnt/golden
 read only = no
 writeable = yes
 case sensitive = yes
```

3. Restart Samba:

```
$ sudo systemctl restart smbd.service
```

2.4.3.4 OSIsoft PI Server and Database Backups

Create the scheduled backup task to backup PI Data Archive files. The script automatically inserts the current datetime stamp into the filename of each file copied to the ForceField drive. Follow these steps:

1. On the server containing the PI Data Archive, open a command prompt with Administrator privileges.

2. Change to the PI\adm directory:

```
> cd /d "%piserver%adm"
```

3. Create the backup directory, and start the Windows scheduled task to perform the backup:

```
> pibackup h:\PIBackup -install
```

Create a scheduled task to copy the backup files to the ForceField HDD. Follow these steps:

1. Open the Task Scheduler and create a new scheduled task to rename, timestamp, and copy the backup files to the ForceField HDD:

   Trigger: At 3:30 AM every day

   Action: Start a Program

   Program/script:
   `C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe`

   Add arguments (optional): `-Command { Get-ChildItem -Path "h:\PIBackup\arc\" | foreach { copy-item -path $($_.FullName) -destination "\\10.100.1.7\ForceField\$(Get-Date -f yyyy-MM-dd_HHMMss)_$($_.name)" } }`

2.5 Microsoft Azure Defender for IoT

Microsoft Azure Defender for IoT, based on technology acquired via CyberX, consists of a single appliance containing the sensor and application interface integrated into Build 4 to meet BAD, hardware modification, firmware modification, and software modification capabilities. The Microsoft Azure Defender for IoT implementation utilizes passive monitoring and protocol analysis to support cybersecurity monitoring and threat detection.

2.5.1 Host and Network Configuration

Microsoft Azure Defender for IoT was installed and configured to support the CRS environment as part of Build 4. The overall build architecture is provided in Figure B-4. The Microsoft Azure Defender for IoT specific components are in Table 2-9.

Table 2-9 Microsoft Azure Defender IoT Deployment

Name	System	OS	CPU	Memory	Storage	Network
Azure Defender for IoT	Dell OEMR XL R340	Ubuntu 18.04	Intel Xeon E-2144G	32 GB	3x 2 TB Drives RAID-5	Testbed LAN 10.100.0.61

2.5.2 Installation

The Microsoft Azure Defender for IoT (Version 10.0.3) appliance was preinstalled with the operating system and application. The appliance is mounted in a rack with power and network interfaces connected to the Testbed LAN on the Eth0 port along with the SPAN connection on the expansion network interface board.

2.5.3 Configuration

To configure the Microsoft Azure Defender for IoT platform, follow these steps:

1. Set the Network Configuration:

   a. Using either SSH, iDRAC, or the KVM Console connections on the appliance, establish shell access to the appliance.

   b. From the console, enter the following command:

   ```
   $sudo cyberx-xsense-network-reconfigure
   ```

   c. The system will walk through a series of network options (Figure 2-25) that are set as follows:

      i. **IP Address**: "10.100.0.61"

      ii. **Subnet Mask:** "255.255.255.0"

      iii. **DNS**: "10.100.0.17"

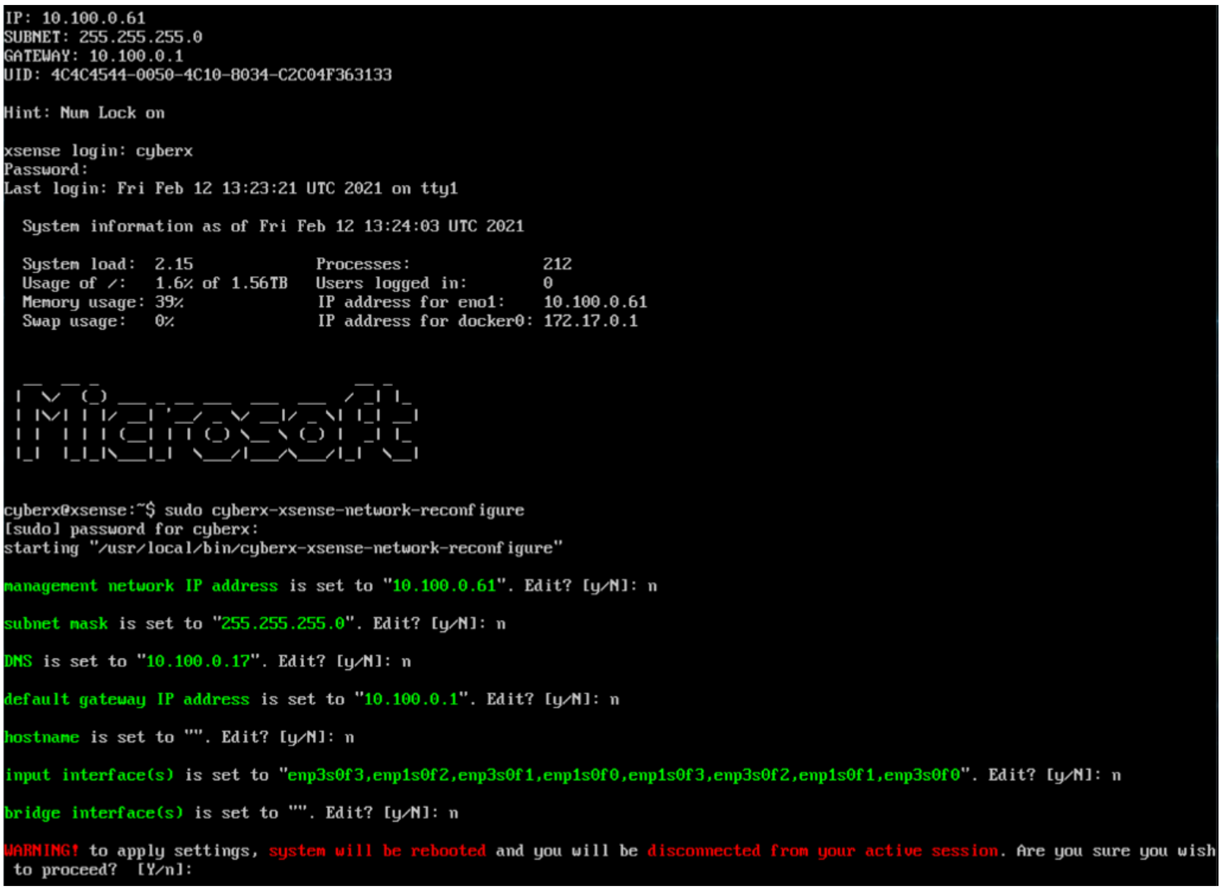
iv. **Default Gateway**: "10.100.0.1"

v. **Hostname:** *Not set*

vi. **Input Interface(s):** "enp3s0f3, enp1s0f2, enp3s0f1, enp1s0f0, enp1s0f3, enp3s0f2, enp1s0f1, enp3s0f0"

vii. **Bridge Interface(s):** *Not Set*

Figure 2-25 Azure Defender for IoT SSH Session for Network Configuration



2. Create AMS Protocol report as a data mining tool:

   a. Login to the application web interface and click **Data Mining** in the left menu navigation.

   b. Click the **+** sign and click **New Report**. In the **Create New Report** panel set the following settings (Figure 2-26):

      i. Under Categories select **AMS** to automatically select the sub-elements, including:

         1) AMS Firmware Information

         2) AMS Index Group

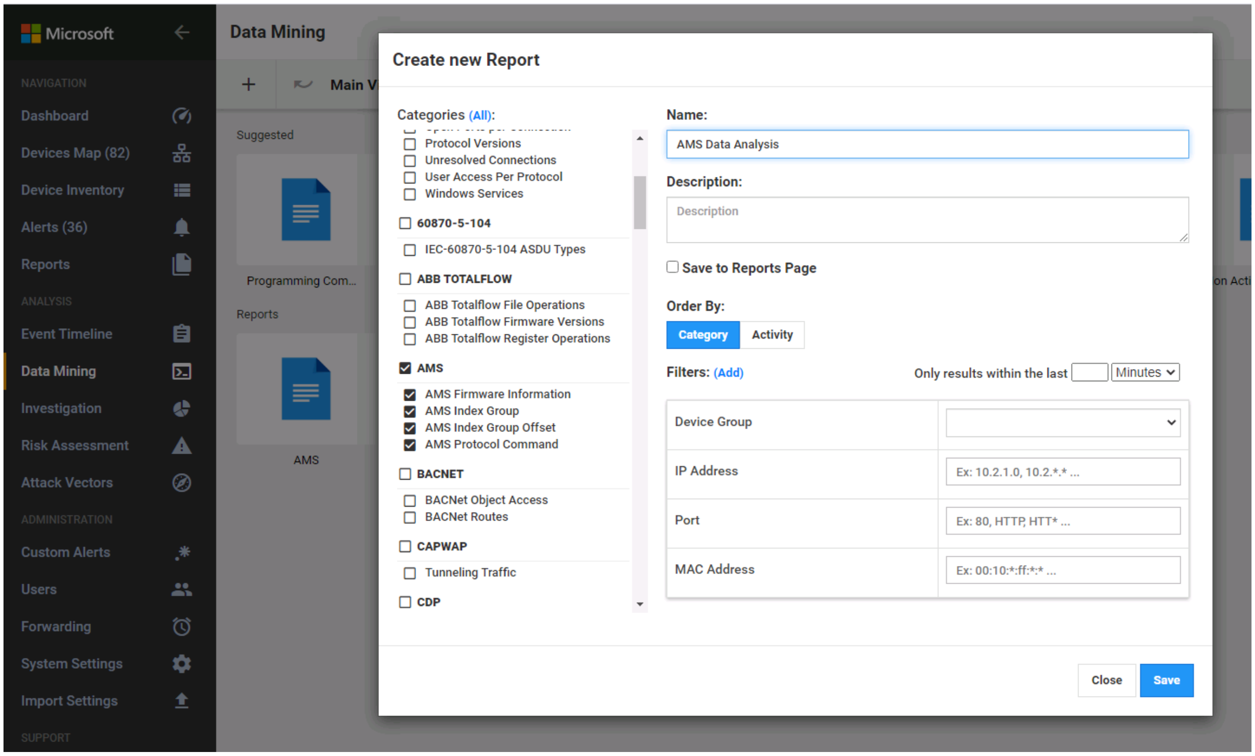
         3) AMS Index Group Offset

4) AMS Protocol Command

ii. Enter "AMS Data Analysis" as the name for the report.

iii. Click **Save.**

Figure 2-26 Azure Defender for IoT Create New Data Mining Report for AMS Protocol Information



3. Create AMS – Custom Alert Rules

For this effort, the CRS PLC is configured to run using firmware version 3.1.4022 as the approved production firmware version. To detect changes to the approved version, custom alert rules are created to monitor for deviations from the approved version numbers through the AMS protocol messages over the network.

a. Click **Horizon** on the left menu navigation.

b. Select **AMS > Horizon Customer Alert** under the Plugin Options on the left menu.

c. Create Custom Alert to Detect Change in PLC Firmware Major Build Number (Figure 2-27):

i. Enter "PLC Firmware Major Build Mismatch" as the title for the custom alert.

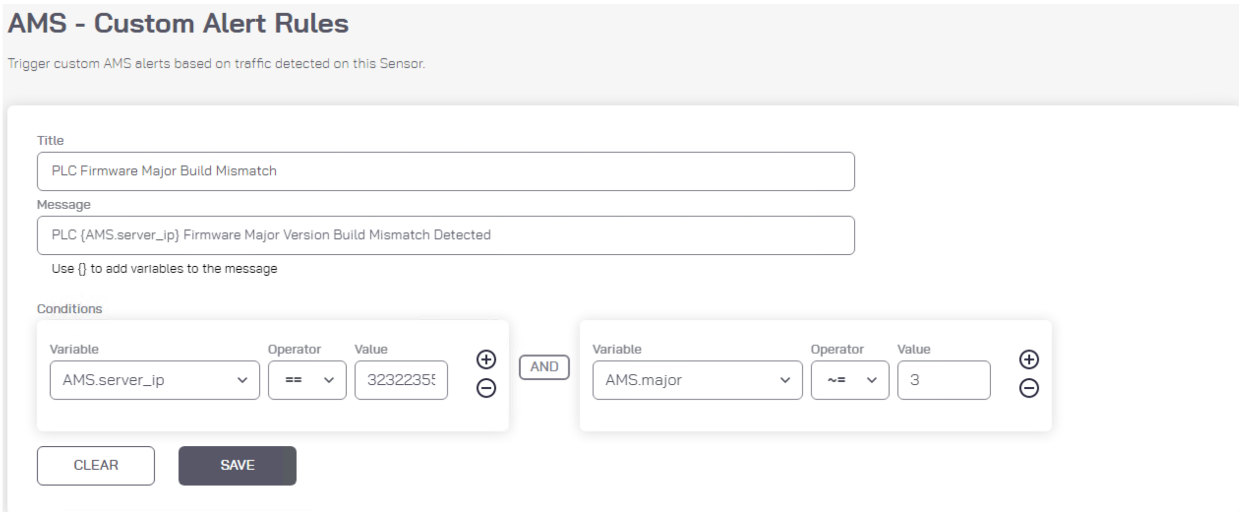
ii. Enter "PLC {AMS_server_ip} Firmware Major Version Build Mismatch Detected" as the message to display with the alert.

iii. Set the following conditions:

1) **AMS_server_ip == 3232235550 (**Note: this is the PLC IP address 192.168.0.30 in Integer format).

2) **AND AMS_major ~= 3**

Figure 2-27 Azure Defender for IoT Custom Alert for Firmware Major Version Number Change



d. Create the custom alert to detect change in PLC firmware minor build number (Figure 2-28):

i. Enter "PLC Firmware Minor Build Mismatch" as the title for the custom alert. PLC Firmware Minor Build Mismatch

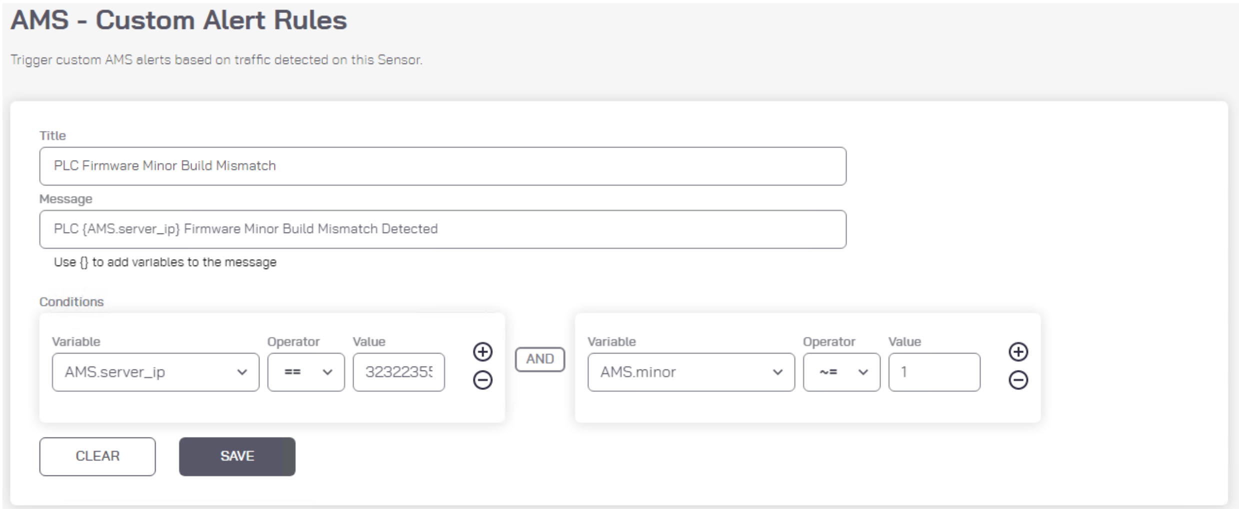
ii. Enter "PLC {AMS_server_ip} Firmware Minor Version Build Mismatch Detected" as the message to display with the alert.

iii. Set the following conditions:

1) **AMS_server_ip == 3232235550** (Note: this is the PLC IP address 192.168.0.30 in Integer format).

2) **AND AMS_minor ~= 1**

Figure 2-28 Azure Defender for IoT Custom Alert for Firmware Minor Version Number Change



e. Create the custom alert to detect change in the PLC Firmware Build Version (Figure 2-29):

   i. Enter "PLC Firmware Build Version Mismatch" as the Title for the custom alert.

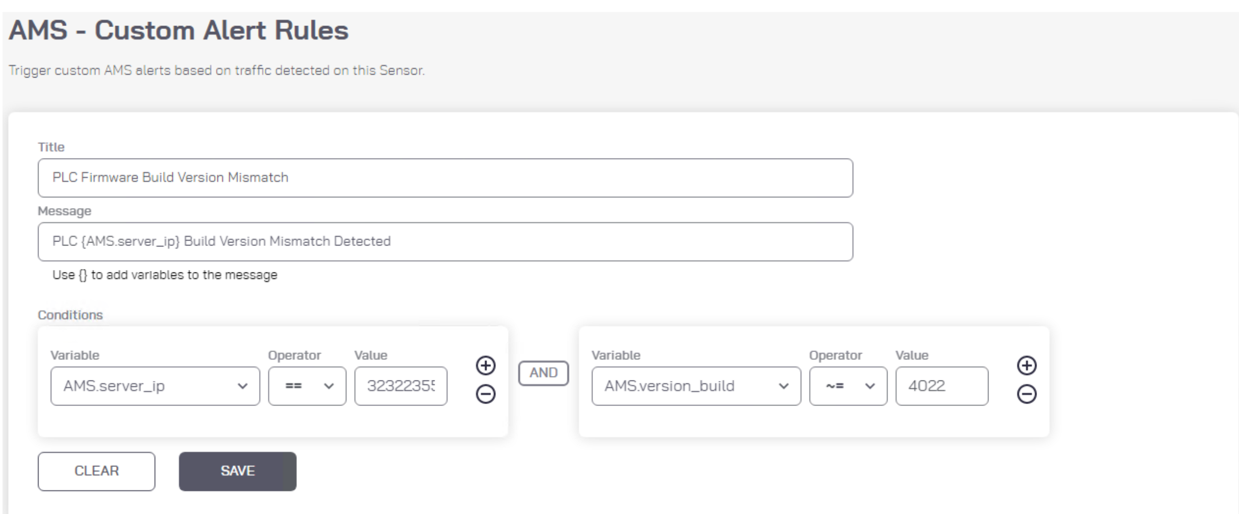
   ii. Enter "PLC {AMS_server_ip} Build Version Mismatch Detected" as the message to display with the alert:

   iii. Set the following conditions:

      1) **AMS_server_ip == 3232235550** (Note: this is the PLC IP address 192.168.0.30 in Integer format).

      2) **AND AMS_version_build ~= 4022**

Figure 2-29 Azure Defender for IoT Custom Alert for Firmware Build Version Number Change



2.6 OSIsoft PI Data Archive

The OSIsoft product included in this practice guide is Process Information (PI), which is used to collect, store, analyze, and visualize testbed data. The product was utilized in Builds 1, 2, 3, and 4 to meet the historian capability by collecting and storing testbed data and the BAD capability by alerting when activity deviates from a baseline.

OSIsoft PI is a suite of software applications for capturing, analyzing, and storing real-time data for industrial processes. Although the PI System is typically utilized as a process historian, the PI System is also utilized to collect, store, and manage data in real time. Interface nodes retrieve data from disparate sources to the PI Server, where the PI Data Archive resides. Data is stored in the data archive and is accessible in the assets defined in the Asset Framework (AF). Data is accessed either directly from the data archive or from the AF Server by using tools in the PI visualization suite.

2.6.1 Host and Network Configuration

PI was installed on virtual machines hosted on hypervisors located in the DMZ and CRS networks. The virtual machine details and resources are provided in Table 2-10, Table 2-11 and, Table 2-12. The overall build architectures utilizing PI are described in Section 4.5 in Volume B.

Table 2-10 OSIsoft PI Domain Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
DMZ Historian	Virtual Machine	Microsoft Windows Server 2016	4x Intel Xeon E3-1240	8 GB	Boot: 80 GB PI Data: 170 GB	DMZ 10.100.1.4

Table 2-11 OSIsoft PI CRS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
CRS Local Historian	Virtual Machine	Microsoft Windows Server 2016	4x Intel Xeon E5-2407	16 GB	Boot: 80 GB PI Data: 170 GB	CRS Supervisory LAN 192.168.0.21

Table 2-12 OSIsoft PI PCS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
PCS Local Historian	Virtual Machine	Microsoft Windows Server 2008 R2	1x Intel i5-4590	2 GB	50 GB	PCS VLAN 2 172.16.2.14

2.6.2 Installation

PI was previously installed in the testbed as part of the *NISTIR 8219: Securing Manufacturing Industrial Control Systems: Behavioral Anomaly Detection*, https://www.nccoe.nist.gov/sites/default/files/library/mf-ics-nistir-8219.pdf. The installation for this project involved upgrading the existing CRS Local Historian and DMZ Historian VMs to Microsoft Windows Server 2016, and subsequently upgrading all the PI software components. Step-by-step instructions for each PI component installation are not included for brevity. Detailed instructions provided by the vendor can be found on the OSIsoft Live Library: https://livelibrary.osisoft.com/.

DMZ Historian Server

The following software is installed on the DMZ Historian server:

- Microsoft SQL Server 2019 Express 15.0.2080.9
- PI Server 2018 (Data Archive Server, Asset Framework Server)
- PI Server 2018 SP3 Patch 1
- PI Interface Configuration Utility version 1.5.1.10
- PI to PI Interface version 3.10.1.10
- PI Interface for Ramp Soak Simulator Data 3.5.1.12
- PI Interface for Random Simulator Data 3.5.1.10
- PI Connector Relay version 2.6.0.0
- PI Data Collection Manager version 2.6.0.0
- PI Web API 2019 SP1 version 1.13.0.6518

CRS Local Historian Server (Collaborative Robotics System)

The following software is installed on the CRS Local Historian server:

- Microsoft SQL Server 2019 Express 15.0.2080.9
- PI Asset Framework Service 2017 R2 Update 1
- PI Data Archive 2017 R2A
- PI Server 2018 SP3 Patch 1
- PI Interface Configuration Utility version 1.5.1.10
- PI to PI Interface version 3.10.1.10
- PI Interface for Ramp Soak Simulator Data 3.5.1.12
- PI Interface for Random Simulator Data version 3.5.1.10
- PI Interface for Performance Monitor version 2.2.0.38
- PI Ping Interface version 2.1.2.49
- PI Interface for Modbus ReadWrite version 4.3.1.24
- PI Interface for SNMP ReadOnly version 1.7.0.37

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

- PI TCP Response Interface version 1.3.0.47
- PI Processbook 2015 R3 Patch 1 version 3.7.1.249
- PI Vision 2019 Patch 1 version 3.4.1.10
- PI System Connector version 2.2.0.1

PCS Local Historian (Process Control System Historian)

- Rockwell FactoryTalk Historian SE version 1.00

2.6.3 Configuration

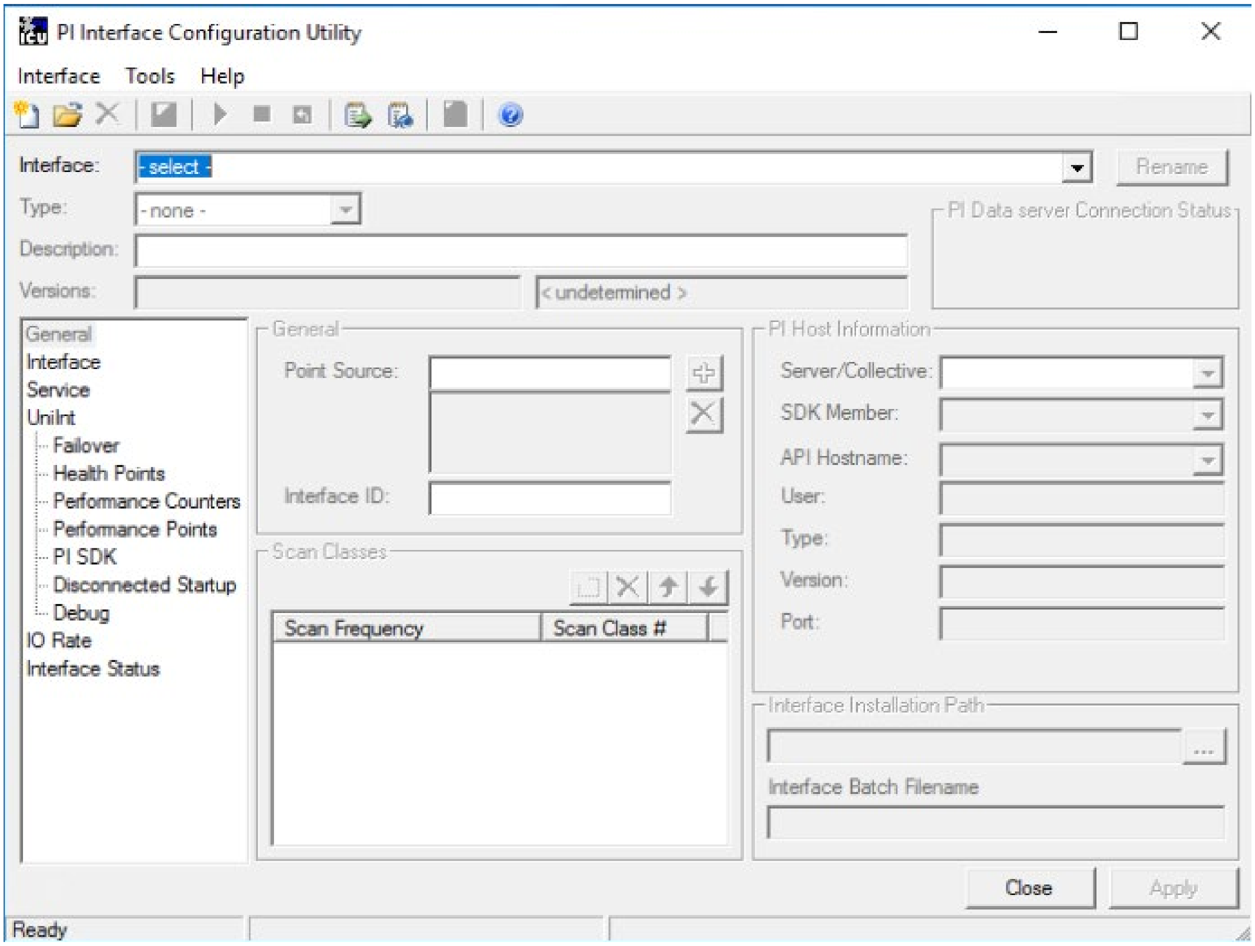
The following sections describe how to configure select PI components to enable the capabilities described in this guide. Configurations for the other PI components are not included for brevity.

2.6.3.1 PI to PI Interface (PCS)

The PCS uses the Rockwell FactoryTalk Historian to collect, store, and analyze historical process data. The PI to PI Interface is used to duplicate the process data to the DMZ Historian server. The following steps describe how to configure the PI-to-PI Interface to collect data from the Rockwell FactoryTalk Historian.

1. On the DMZ Historian server, launch the **PI Interface Configuration Utility** as shown in Figure 2-30 from the Start menu and sign in with the local administrator account.

Figure 2-30 Screenshot of the PI Interface Configuration Utility before the Interface is configured.



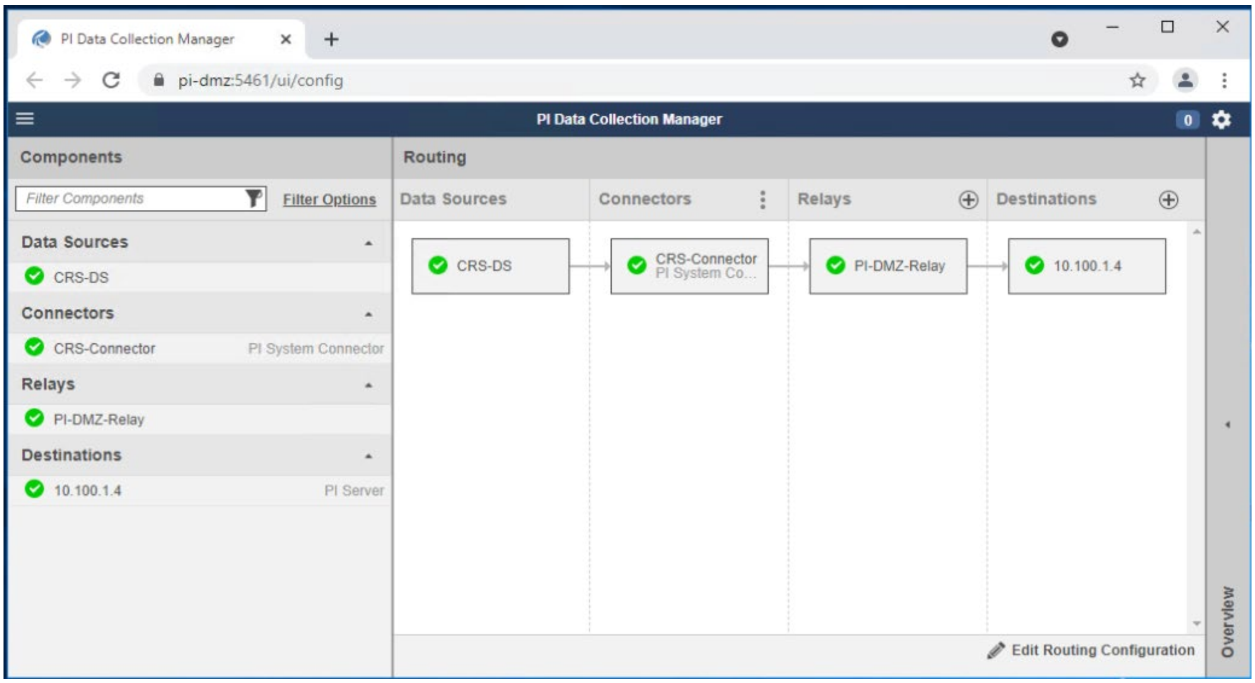
2. On the top menu, click **Interface > New Windows Interface Instance from BAT File…**
3. Navigate to **E:\Program Files (x86)\PIPC\Interfaces\PItoPI** and select the file **PItoPI.bat_new**.
4. In the “Select Host PI Data server/collective” dialog box, select **PI-DMZ** from the drop-down menu and click **OK**.
5. In the left navigation panel select **PItoPI**. In the Source host textbox, enter "172.16.2.4".
6. In the left navigation panel, select **Service**. In the “Create / Remove” section click the **Create** button. Click **Yes** in the dialog box.
7. Enter the commands `net start PItoPI` and `net stop PItoPI` in the files **pisrvsitestart.bat** and **pisrvsitestop.bat files**, respectively. Save and close the files.
8. At the bottom of the **PI Interface Configuration Utility** click the **Apply** button. On top menu bar click the green play button  to start the service.

9. Close the **PI Interface Configuration Utility**. The interface is now configured to pull tags from the Rockwell Historian.

2.6.3.2 PI System Connector (CRS)

The PI System Connector is used to duplicate process data on the DMZ Historian from the CRS Local Historian server. The following steps describe how to configure the PI-to-PI Interface to collect data from the OSIsoft PI Server.

Figure 2-31 Screenshot of the PI Data Collection Manager Displaying Green Checkmarks After the PI System Connector is Properly Configured



1. On the DMZ Historian server, launch the **PI Data Collection Manager** as shown in Figure 2-31 from the Start menu and sign in with the local administrator account.
   a. Click **+** on the Relays column to add a new connector relay. Use the following settings:
   b. Name: `PI-DMZ-Relay`
   c. Address: `10.100.1.4`
   d. Port: `5460`
2. Username: `.\piconnrelay_svc`
3. Click **Save Settings** to add the connector relay.
4. Click **+ Add Destination** to add the target PI Data Archive and PI AF Server. Use the following settings:
   a. Name: `10.100.1.4`

   b. PI Data Archive Address: `10.100.1.4`

   c. AF Server: `10.100.1.4`

5. Click **Save Settings** to add the destination.

6. On the CRS Local Historian server, open the **PI System Connector Administration** from the Start menu and sign in with the local administrator account.

7. Click **Set up Connector** to create a new connector.

8. Use the following information to request registration:

   a. Registration Server Address: https:`//PI-DMZ:5460`

   b. Registration Server Username: `piconnrelay_svc`

   c. Registration Server Password:

   d. Description: `Registration to PI-DMZ`

9. Click **Request Registration** to send the request to the DMZ Historian server.

10. On the DMZ Historian server, open the **PI Data Collection Manager** from the Start menu and sign in with the local administrator account.

11. Click **Untitled Connector 1** and click **Approve This Registration and Configure** to approve the PI System Connector registration.

12. In the **Untitled Connector 1** details panel, click **Edit**.

13. Use the following information to create the CRS-Connector connector:

    a. Name: `CRS-Connector`

    b. Description: `Registration to PI-DMZ`

14. Click **Save Settings** to create the CRS-Connector.

15. Click **CRS-Connector** in the **Connectors** column. On the **Overview** panel click **CRS-Connector**: **No Data Sources** option to create the data source.

16. On the **CRS-Connector** Connector Details in the **Overview** panel, click **+ Add Data Source**.

17. In the **Data Source Settings** window, use the following settings:

    a. Name: `CRS-DS`

    b. Source AF Server: `PI-Robotics`

    c. Source AD Database: `TestbedDatabase`

    d. Select **Collect All Data from this Entire Database.**

18. Click **Save** to save the data source.

19. Click 10.100.1.4 in the **Destination** column of the **Routing** panel and then click **Data** in the **10.100.1.4 Destination Details** panel to configure the destination database for the CRS-Connector.
20. In the **10.100.1.4 Destination Details** panel, change from **Change Default Settings for new connectors** to "CRS-Connector" and then click **Edit Destination Data Settings.**
21. In the **10.100.1.4 Destination Details** of the **Overview** panel, use the following settings:
    a. Change the connector to **CRS-Connector.**
    b. Database: `CRS-backup`
    c. Click on **Elements** and it will change **<select a path using the tree below>** to **$Elements**
    d. Use default settings in **Root AF Elements** and **Point Names.**
    e. **Create root Element CRS-Connector** checkbox: Checked
    f. **Prefix Point CRS-Connector** checkbox: Checked
22. Click **Save Destination Data Settings** to save the configuration.
23. Click the white space in the **Routing** panel.
24. Click **CRS-Connector: No Relays** in the **Overview** panel.
25. Select the **PI-DMZ-Relay** checkbox in the **Routing** panel.
26. Click the white space in the **Routing** panel again, then **Click PI-DMZ-Relay: No Destination** to add the routing between relays and destinations.
27. Select the **10.100.1.4** checkbox to add the routing between the relay and the destination.
28. Click **Save Configuration**.
29. In the **Save Routing and Data Configuration** window, select **Save and Start All Components** to continue.
30. Each box should now contain a green checkmark (i.e., Data Sources, Connectors, Relays, and Destinations). The elements in the AF database “testbeddatabase” on CRS Local Historian server is now replicated to AF database “CRS-backup” on the DMZ Historian server.
31. Finally, create a Windows firewall rule to open the inbound ports 5460, 5461, 5471, and 5472.

2.6.3.3 PI Asset Template Analysis Functions and Event Frames

Analysis functions and event frame templates were created to generate alerts in the PLC asset template when their respective anomalous events are detected. When an analysis function result is TRUE, an event frame is generated from the event frame template and ends when the analysis function result is FALSE or per a user-defined function. The following steps describe how the “Station Mode Error” analysis function and event frame template were created and used in Scenario 10.

1. On the CRS Local Historian server, open the **PI System Explorer** by navigating to **Start Menu > PI System > PI System Explorer**.
2. On the left navigation panel, select **Library**.
3. In the navigation tree in the **Library** panel, select **Templates > Event Frame Templates.**
4. Right click in the whitespace of the **Element Templates** window and select **New Template**.
    a. Enter the following:
    b. Name: `Station Mode Error`
    c. Description: `CRS Workcell machining station mode error`
5. Naming Pattern: `ALARM-%ELEMENT%.%TEMPLATE%.%STARTTIME:yyyy-MM-dd HH:mm:ss.fff%`
6. In the navigation tree in the **Library** panel, select **Templates > Element Templates > Machining_Station**.
7. In the **Machining_Station** panel select the **Analysis Templates** tab and click **Create a new analysis template.**
8. Enter the name “Station Mode Error” in the **Name** textbox, enter a description of the analysis in the Description textbox, and select the option “Event Frame Generation” for the **Analysis Type**.
9. Select “Station Mode Error” in the **Event Frame** template drop-down menu.
10. In the **Expression** field for “StartTrigger1”, enter the expression:

    ```
    'RawMode' < 0 OR 'RawMode' > 1;
    ```

11. Click the **Add…** drop-down menu and select **End Trigger**, and enter the expression:

    ```
    ('RawMode' > 0 AND 'RawMode' < 1)
    ```

12. Select the “Event-Triggered” option for the **Scheduling** type.
13. Click the **Check In** button on the top menu to save all changes to the database.

2.6.3.4 PI Web API

The PI Web API is used by Dragos to collect event frames from the DMZ Historian server. After completing installation of the PI Web API, the “Change PI Web API Installation Configuration” dialog displays. The following steps describe how to configure the Web API on the DMZ Historian server.

1. In the **Telemetry** section, verify the checkbox option and click **Next.**
2. In the **Configuration Store** section, select "PI-ROBOTICS" in the Asset Server drop-down menu and click **Connect**. Leave the default instance name.
3. In the **Listen Port** section, verify port 443 is entered in the **Communication Port Number** textbox and check the **Yes, please create a firewall Exception for PI Web API** checkbox.

4. In the **Certificate** section, click **Next** to continue and use the self-signed certificate or select **Change** to modify the certificate.

5. In the **API Service** section, leave the default service `NT Service\piwebapi` and click **Next**.

6. In the **Crawler Service** section, leave the default `service NT Service\picrawler` and click **Next**.

7. In the **Submit URL** section, enter the URL of the DMZ Historian server Web API service: `https://pi-dmz/piwebapi/`. Click **Next**.

8. In the **Review Changes** section, verify all the configuration settings, check the checkbox Accept all the configurations, and click **Next**.

9. Click **Finish** to complete the configuration.

2.6.3.5 Firmware Integrity Checking

Software was developed to demonstrate the ability of PI to obtain device and firmware data from a Beckhoff PLC for integrity checking purposes. A new PLC task was programmed to periodically query its operating system for hardware and software telemetry and make it available via Modbus TCP. PI will query these Modbus registers and use analysis functions to generate event frames if any tags do not match their expected values.

It is important to note that this capability was developed to demonstrate a method of maintaining visibility of PLC hardware and firmware version numbers for integrity purposes and is not secure or infallible. If a malicious actor takes control of the PLC, the hardware and firmware versions provided by the PLC can be spoofed.

The following steps describe how to sequentially configure this capability across multiple systems and software. Only one system or software is described in each section.

Beckhoff PLC Modbus TCP Server

The base Modbus TCP server configuration file only allows one PLC task to write to the registers. The following steps describe how to modify the configuration to allow two PLC tasks to write to the Modbus TCP server input registers.

1. Log in to the Windows CE Desktop of the Beckhoff PLC and open the XML file: `\TwinCAT\Functions\TF6250-Modbus-TCP\Server\TcModbusSrv.xml`

2. Modify the `<InputRegisters>` … `</InputRegisters>` section to the following:

```
<InputRegisters>
 <MappingInfo>
   <AdsPort>851</AdsPort>
   <StartAddress>32768</StartAddress>
   <EndAddress>32895</EndAddress>
   <VarName>GVL.mb_Input_Registers</VarName>
  </MappingInfo>
  <MappingInfo>
   <AdsPort>852</AdsPort>
   <StartAddress>32896</StartAddress>
   <EndAddress>33023</EndAddress>
   <VarName>GVL.mb_Input_Registers</VarName>
  </MappingInfo>
</InputRegisters>
```

3. Save and close the file.

4. Restart the PLC.

The Modbus TCP server will now have two register address ranges: 128 addresses for the PLC task at port 851, and 128 addresses for the PLC task at port 852.

Beckhoff PLC Project

A new PLC task must be created to perform the integrity checking and write the data to the Modbus TCP registers. The following steps describe how to create and configure the new task.

1. On the engineering workstation, open the **TwinCAT XAE Shell** by navigating to **Start Menu > Beckhoff > TwinCAT XAE Shell** and open the current PLC project.

2. In the **Solution Explorer**, right click **PLC** and select **Add New Item…**

3. In the **Add New Item** dialog box, select **Standard PLC Project**, enter the name `FirmwareIntegrityCheck` in the **Name** textbox, and click **Add**.

4. In the **Solution Explorer**, double click **SYSTEM > Tasks > PLCTask1**. Verify the **Auto Start** checkbox is checked and change the **Cycle Ticks** textbox to `100` ms.

5. In the **Solution Explorer**, right click **PLC > FirmwareIntegrityCheck > References** and click **Add library**… In the dialog box, select the library **System > Tc2_System** and click **OK**.

6. In the **Solution Explor**er, right click **PLC > GVLs** and click **Add > Global Variable List**. In the dialog box enter the name `GVL` in the **Name** textbox and click **Open**.

7. In the **Editor Window**, enter the following code:

```
VAR_GLOBAL
   mb_Input_Registers : ARRAY [0..127] OF WORD;
END_VAR
```

8. In the **Solution Explorer**, right click **PLC > FirmwareIntegrityCheck > POU** (Program Organizational Unit) and select **Add > POU**. In the **Add POU** dialog box, enter the name `GetSystemInfo`, select the type **Function Block**, select the **Implementation Language** `Structured Text (ST)` and click **Open**.

9. In the **Editor Window**, enter the following code in the **Variables** section:

```
// Gathers PLC information for system integrity checking
// (e.g., PLC serial number, TwinCAT version).
FUNCTION_BLOCK GetSystemInfo
VAR_INPUT
	NetId : T_AmsNetId; // AMS network ID of the PLC
END_VAR
VAR_OUTPUT
	HardwareSerialNo : WORD; // Serial number of PLC
	TwinCATVersion : WORD; // Version number of TwinCAT
	TwinCATRevision : WORD; // Revision number of
TwinCAT
	TwinCATBuild : WORD; // Build number of TwinCAT
END_VAR
VAR
	DeviceData : FB_GetDeviceIdentification; //PLC data
struct
	Timer : TON; // Timer to trigger the scan
	Period : TIME := T#5M; // Amount of time between
each scan
	State : INT := 0; // Function block state
END_VAR
```

10. In the **Editor Window**, enter the following code in the **Code** section:

```
CASE state OF
      0:
            // Start a new request for device
identification
            DeviceData(bExecute:=TRUE, tTimeout:=T#100MS,
sNetId:=NetId);
            // Switch to the next state once the request
completes
            IF DeviceData.bBusy = FALSE THEN
                  state := 10;
            END_IF
      10:
            // Store the interesting data into our internal
variables
            HardwareSerialNo :=
STRING_TO_WORD(DeviceData.stDevIdent.strHardwareSerialNo);
            TwinCATVersion   :=
STRING_TO_WORD(DeviceData.stDevIdent.strTwinCATVersion);
            TwinCATRevision  :=
STRING_TO_WORD(DeviceData.stDevIdent.strTwinCATRevision);
            TwinCATBuild     :=
STRING_TO_WORD(DeviceData.stDevIdent.strTwinCATBuild);
            // Reset the timer and move to the next state
            Timer(IN:= FALSE);
            state := 20;
      20:
            // Make sure the timer is running and change to
the
          // next state once the period has been reached
            Timer(IN:=TRUE,PT:=Period);
            IF Timer.Q = TRUE THEN
                  state := 0;
            END_IF
END_CASE
```

11. Save and close the POU.
12. In the **Solution Explorer**, double click **PLC > FirmwareIntegrityCheck > POUs > MAIN (PRG).**
13. In the **Editor Window**, enter the following into the **Variables** section (your AMS net ID may differ from what is shown below):

```
PROGRAM MAIN
VAR
      PLCInfo : GetSystemInfo; // Periodically collects
PLC data
      SelfNetId : T_AmsNetId := '5.23.219.8.1.1'; // Local
address
END_VAR
```

14. In the **Editor Window**, enter the following into the **Code** section:

```
      // Captures hardware serial numbers and TwinCAT version
      // numbers from the PLC and shares them with other
      // devices via Modbus TCP.
      PLCInfo( NetId:=SelfNetId,
               HardwareSerialNo => GVL.mb_Input_Registers[0],
               TwinCATVersion   => GVL.mb_Input_Registers[1],
               TwinCATRevision  => GVL.mb_Input_Registers[2],
               TwinCATBuild     => GVL.mb_Input_Registers[3]
              );
```

15. Save and close the POU.

16. In the top menu, select **Build > Build Project**. Once the build process completes select **PLC > Login**. In the **TwinCAT PLC Control** dialog box, select **Login with download**, verify the **Update boot project** checkbox is checked, and click **OK**. If the PLC code is not running after the download completes, select **PLC > Start** in the top menu.

17. The firmware integrity checking code is now running on the Beckhoff PLC. In the top menu select **PLC > Logout** and close the TwinCAT XAE Shell.

The PLC will now write the hardware serial number and firmware version numbers to the Modbus TCP server registers.

OSIsoft PI Points

The following steps describe how to create the PI points and tags in the CRS Local Historian server and duplicate the tags to the DMZ Historian server.

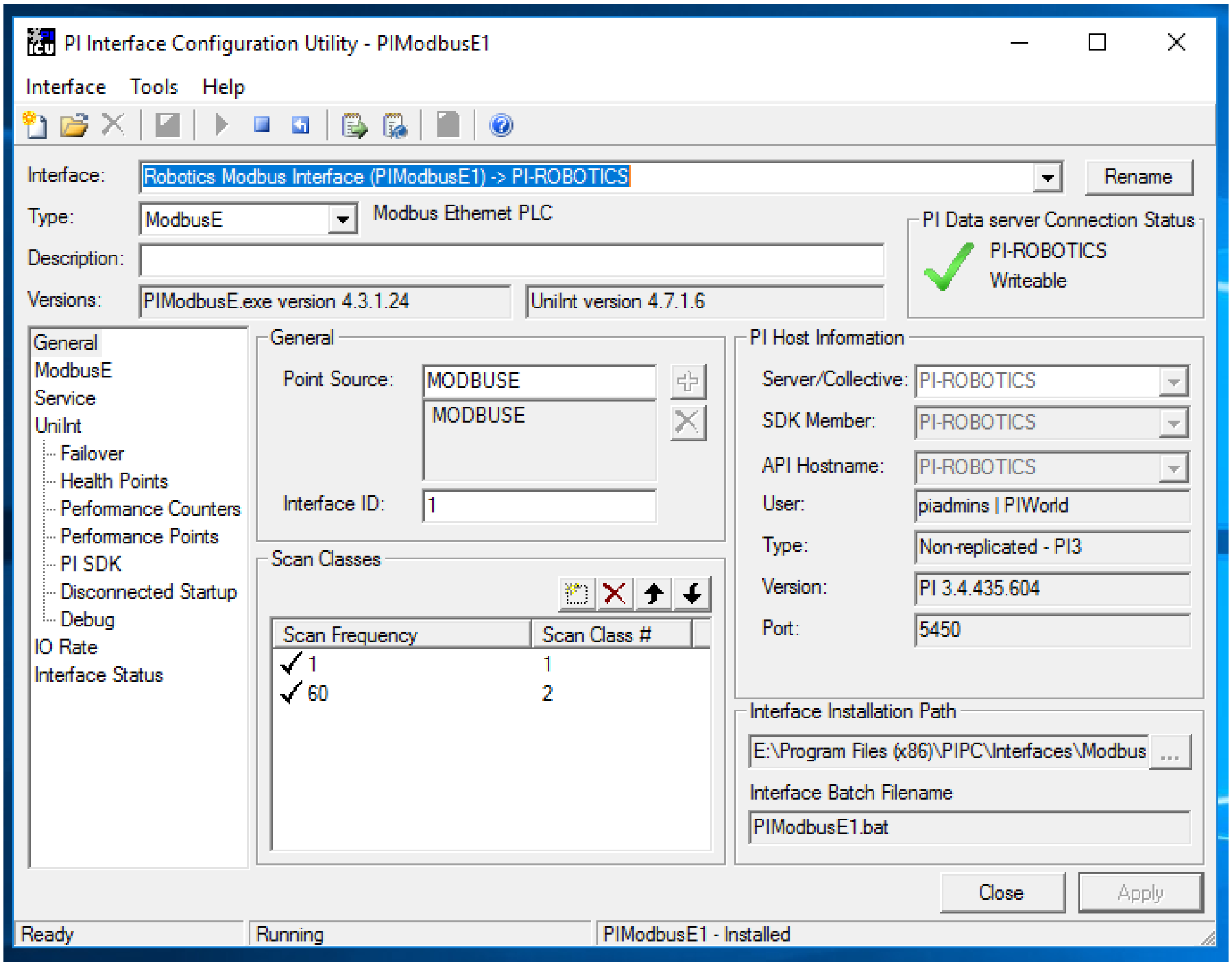
1. On the CRS Local Historian server, open the PI Interface Configuration Utility by navigating to **Start > All Programs > PI System > PI Interface Configuration Utility**.

2. In the **Interface** drop-down menu, select **Modbus Interface (PIModbusE1).**

3. Select the **General** menu option. In the **Scan Classes** section, click **New Scan Class**.

4. Set the **Scan Frequency** to "60" and the **Scan Class #** to the next sequential class number as shown in Figure 2-32 below.

Figure 2-32 Screenshot of the PI Interface Configuration Utility Showing the Added Scan Class # 2 for Polling the PLC Every 60 Seconds



5. Click **Apply** and close the program.

6. On the CRS Local Historian server, open the **PI System Management Tools** by navigating to **Start Menu > PI System > PI System Management Tools**.

7. In the System Management Tool panel, select **Points > Point Builder**.

8. Create a new tag for the PLC hardware serial number with the following configuration:

    a. Name: `PLC-HardwareSerialNumber`

    b. Server: `PI-ROBOTICS`

    c. Descriptor: `Hardware serial number of the CRS Beckhoff PLC`

    d. Point Source: `MODBUSE`

    e. Point Type: `Int16`

   f. Location 1: 1
   g. Location 2: 0
   h. Location 3: 104
   i. Location 4: 2
   j. Location 5: 32897
   k. Instrument Tag: 192.168.0.30

9. Create a new tag for the PLC TwinCAT build number with the following configuration:
   a. Name: PLC-TwinCATBuildNumber
   b. Server: PI-ROBOTICS
   c. Descriptor: Build number of the CRS PLC TwinCAT firmware.
   d. Point Source: MODBUSE
   e. Point Type: Int16
   f. Location 1: 1
   g. Location 2: 0
   h. Location 3: 104
   i. Location 4: 2
   j. Location 5: 32900
   k. Instrument Tag: 192.168.0.30

10. Create a new tag for the PLC TwinCAT revision number with the following configuration:
    a. Name: PLC-TwinCATRevisionNumber
    b. Server: PI-ROBOTICS
    c. Descriptor: Revision number of the CRS PLC TwinCAT firmware.
    d. Point Source: MODBUSE
    e. Point Type: Int16
    f. Location 1: 1
    g. Location 2: 0
    h. Location 3: 104
    i. Location 4: 2

   j. Location 5: `32899`

   k. Instrument Tag: `192.168.0.30`

11. Create a new tag for the PLC TwinCAT version number with the following configuration as shown in Figure 2-33:

   a. Name: `PLC-TwinCATVersionNumber`

   b. Server: `PI-ROBOTICS`

   c. Descriptor: `Version number of the CRS PLC TwinCAT firmware.`

   d. Point Source: `MODBUSE`

   e. Point Type: `Int16`

   f. Location 1: `1`

   g. Location 2: `0`

   h. Location 3: `104`

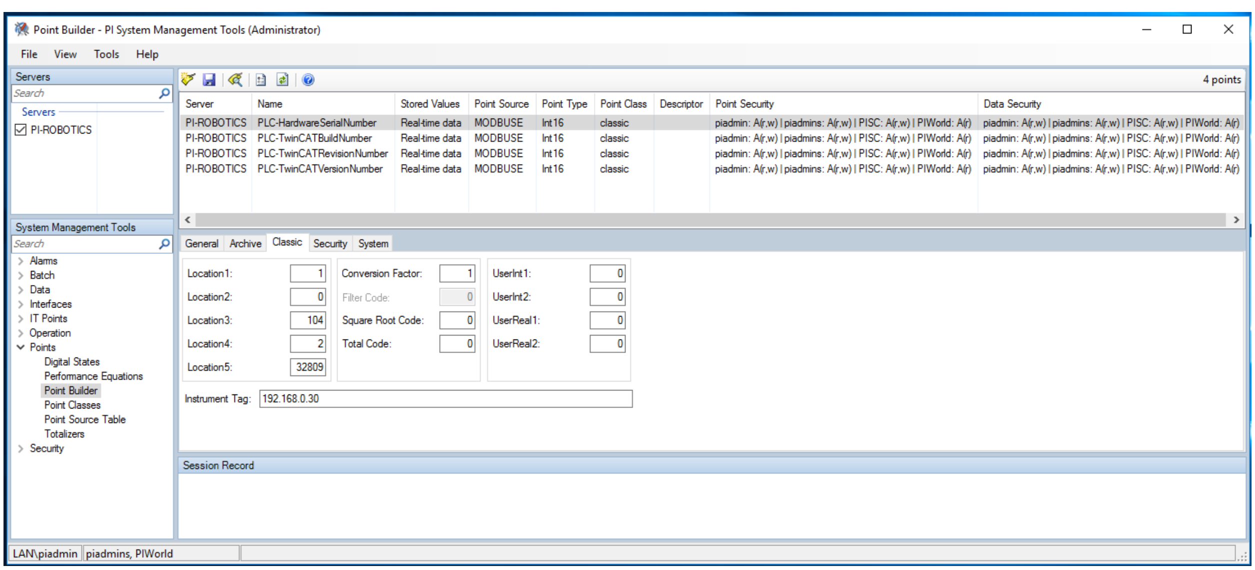
   i. Location 4: `2`

   j. Location 5: `32898`

   k. Instrument Tag: `192.168.0.30`

12. Close the **PI System Management Tools** program. The PI points are now available to the DMZ Historian server via the PI System Connector.

Figure 2-33 Screenshot of the PI System Management Tools Component After Configuring the PI Points for PLC Hardware and Firmware Version Number Integrity Checking



13. On the DMZ Historian server, open **PI System Explorer** by navigating to **Start Menu > PI System > PI System Explorer**.
14. On the left navigation panel, select **Library**.
15. In the navigation tree in the **Library** panel, select **Templates > Element Templates > PLCTemplate**.
16. Open the **Attribute Templates** tab in the **PLCTemplate** panel.
17. On the top menu bar, click **New Attribute Template** and create a new attribute for the PLC hardware serial number by entering the following configuration:
    a. Name: `HardwareSerialNumber`
    b. Description: `Hardware serial number of the CRS Beckhoff PLC.`
    c. Value Type: `Int16`
    d. Data Reference: `PI Point`
    e. Tag: `\\PI-ROBOTICS\PLC-HardwareSerialNumber`
18. On the top menu bar click **New Attribute Template** and create a new attribute for the expected hardware serial number by entering the following configuration:
    a. Name: `HardwareSerialNumber-Expected`
    b. Description: `Expected hardware serial number of the CRS Beckhoff PLC.`
    c. Value Type: V
    d. Data Reference: `None`
19. On the top menu bar, click **New Attribute Template** and create a new attribute for the PLC TwinCAT build number by entering the following configuration:
    a. Name: `TwinCATBuildNumber`
    b. Description: `Build number of the CRS PLC TwinCAT firmware.`
    c. Value Type: `Int16`
    d. Data Reference: `PI Point`
    e. Tag: `\\PI-ROBOTICS\PLC-TwinCATBuild`
20. On the top menu bar, click **New Attribute Template** and create a new attribute for the PLC TwinCAT revision number by entering the following configuration:
    a. Name: `TwinCATRevisionNumber`
    b. Description: `Revision number of the CRS PLC TwinCAT firmware.`

c. Value Type: `Int16`

d. Data Reference: V

e. Tag: `\\PI-ROBOTICS\PLC-TwinCATRevision`

21. On the top menu bar, click **New Attribute Template** and create a new attribute for the PLC TwinCAT version number by entering the following configuration:

    a. Name: `TwinCATVersionNumber`

    b. Description: `Version number of the CRS PLC TwinCAT firmware.`

    c. Value Type: `Int16`

    d. Data Reference: `PI Point`

    e. Tag: `\\PI-ROBOTICS\PLC-TwinCATVersion`

22. On the top menu bar, click **New Attribute Template** and create a new attribute for the string representation of the version, revision, and build numbers by entering the following configuration:

    a. Name: `TwinCATVersion`

    b. Description: `Version number of the CRS PLC TwinCAT firmware.`

    c. Value Type: `String`

    d. Data Reference: `String Builder`

    e. String: `'TwinCATVersionNumber';.;'TwinCATRevisionNumber';.;'TwinCAT BuildNumber';`

23. On the top menu bar click, **New Attribute Template** and create a new attribute for the PLC expected TwinCAT version number by entering the following configuration as shown in Figure 2-34:

    a. Name: `TwinCATVersion-Expected`

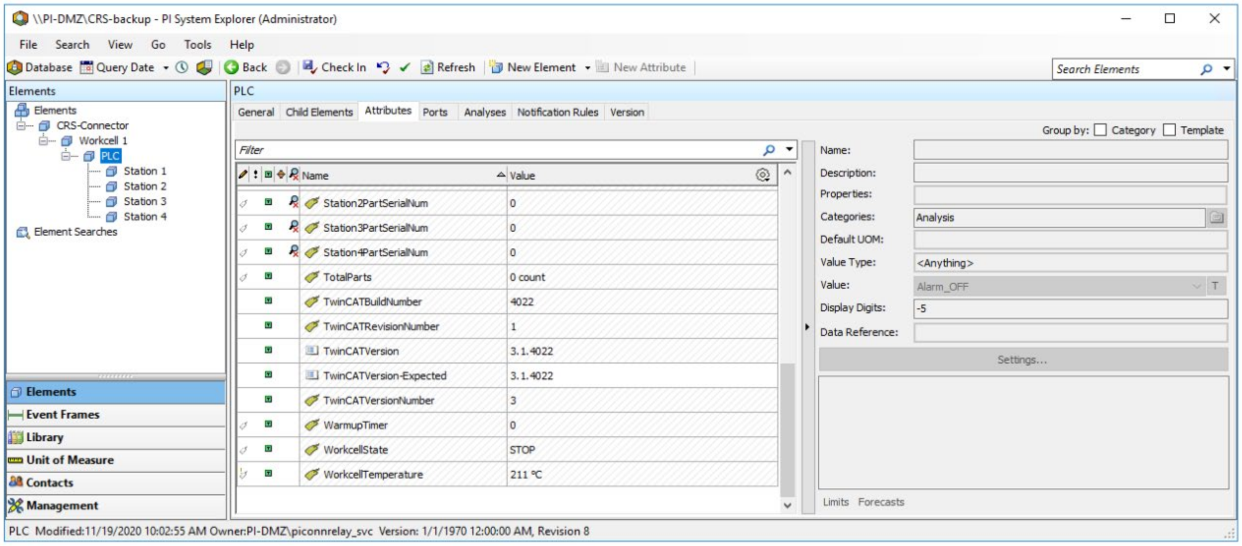
    b. Description: `Expected version number of the CRS PLC TwinCAT firmware.`

    c. Value Type: `String`

    d. Data Reference: `None`

The PI points are now available as PLC attributes in the Asset Framework on the DMZ Historian server.

Figure 2-34 Screenshot of PI System Explorer Displaying some Attributes of the PLC Element. Attributes for the TwinCAT version number are visible in the list.



OSIsoft PI Analyses and Event Frames

The following steps describe how to create the PI analyses and event frame templates to generate event frames when the hardware or firmware version numbers do not match the expected values.

1. In the navigation tree in the **Library** panel, select **Templates > Event Frame Templates**.

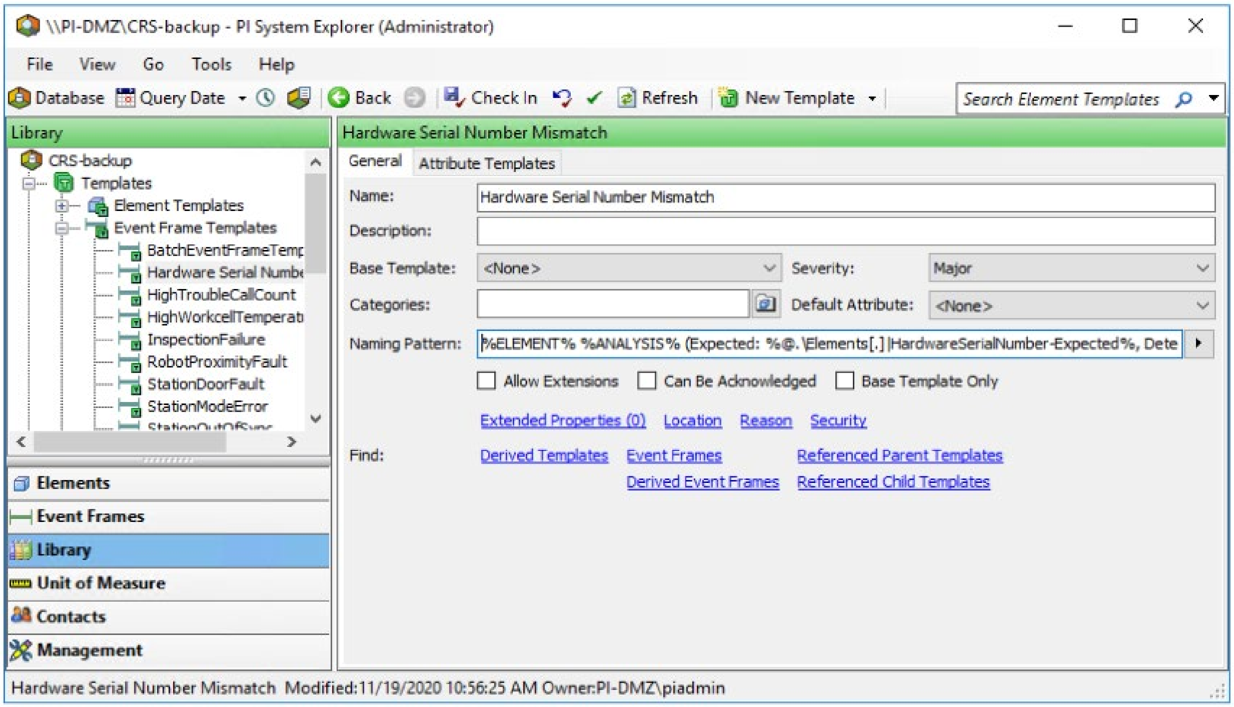
2. On the top menu bar click **New Template** and enter the following configuration as shown in Figure 2-35:

    a. Name: `Hardware Serial Number Mismatch`

    b. Naming pattern: %ELEMENT% %ANALYSIS% (Expected: %@.\Elements`[.]|HardwareSerialNumber-Expected%, Detected: %@.\Elements[.]|HardwareSerialNumber%) %STARTTIME:yyyy-MM-dd HH:mm:ss.fff%`

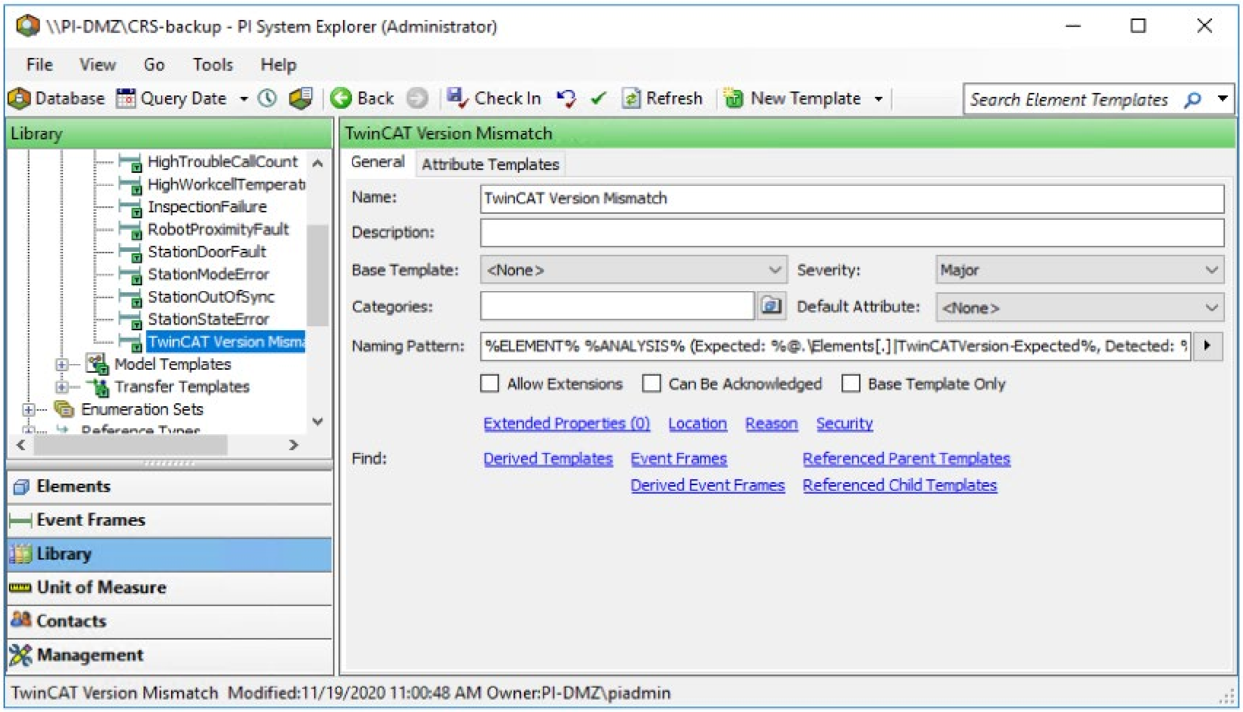
This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Figure 2-35 Screenshot of PI System Explorer Displaying the Hardware Serial Number Mismatch Event Frame Template.



3. On the top menu bar, click **New Template** and enter the following configuration as shown in Figure 2-36:
   a. Name: `TwinCAT Version Mismatch`
   b. Naming pattern: `%ELEMENT% %ANALYSIS% (Expected: %@.\Elements[.]|TwinCATVersion-Expected%, Detected: %@.\Elements[.]|TwinCATVersion%) %STARTTIME:yyyy-MM-dd HH:mm:ss.fff%`

Figure 2-36 Screenshot of PI System Explorer Displaying the TwinCAT Version Mismatch Event Frame Template



4. Click **Check In** on the top menu to save all changes to the database.
5. In the navigation tree in the **Library** panel, select **Templates > Element Templates > PLCTemplate**.
6. Open the **Analysis Templates** tab in the **PLCTemplate** panel and click **Create a new analysis template**.
7. Enter the following configuration as shown in Figure 2-37:
    a. Name: `Hardware Serial Number Mismatch`
    b. Description: `The PLC hardware serial number does not match the expected serial number.`
    c. Analysis Type: `Event Frame Generation`
    d. Enable analyses when created from template: Checked
    e. Generation Mode: `Explicit Trigger`
    f. Event Frame Template: `Hardware Serial Number Mismatch`
8. In the **Expression** field for “StartTrigger1”, enter the expression:

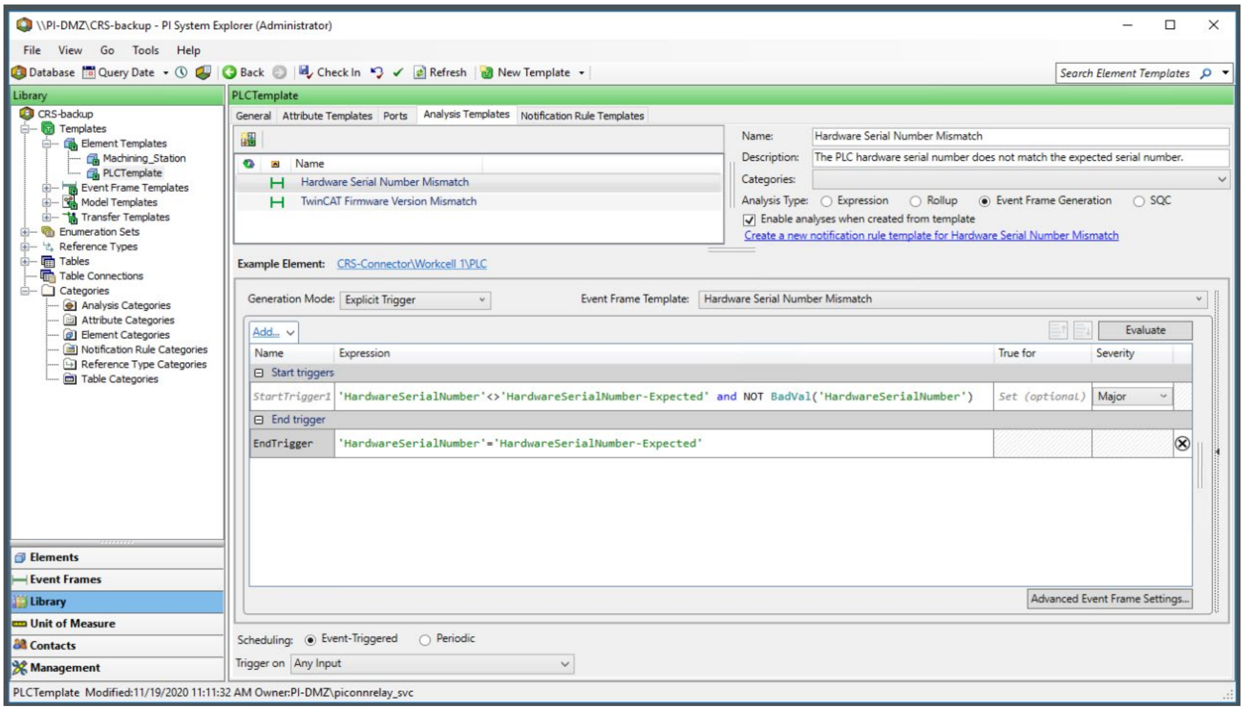
```
'HardwareSerialNumber'<>'HardwareSerialNumber-Expected' and NOT
BadVal('HardwareSerialNumber');
```

9. Click **Add**… drop-down menu and select **End Trigger**, and enter the expression:

```
'HardwareSerialNumber'='HardwareSerialNumber-Expected';
```

10. Select the “Event-Triggered” option for the **Scheduling** type and “Any Input” for the **Trigger On** drop-down menu.

Figure 2-37 Screenshot of PI System Explorer Displaying the Hardware Serial Number Mismatch Analysis Template in the PLC Element Template



11. To create a new analysis template for TwinCAT firmware version mismatch, click **Create a new analysis template**.

12. Enter the following configuration as shown in Figure 2-38:

    a. Name: `TwinCAT Firmware Version Mismatch`

    b. Description: `The TwinCAT version installed in the PLC does not match the expected version.`

    c. Analysis Type: `Event Frame Generation`

    d. Enable analyses when created from template: Checked

    e. Generation Mode: `Explicit Trigger`

   f. Event Frame Template: `Hardware Serial Number Mismatch`

13. In the **Expression** field for "StartTrigger1", enter the expression:

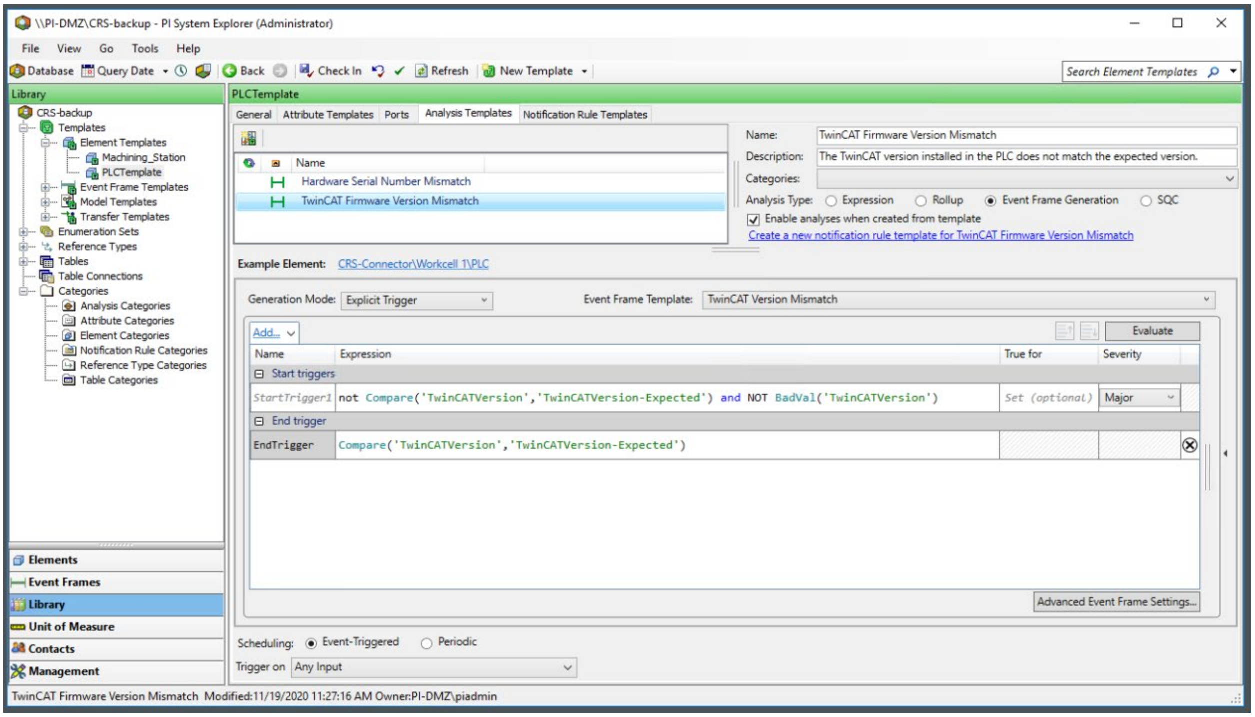
```
not Compare('TwinCATVersion','TwinCATVersion-Expected') and NOT
BadVal('TwinCATVersion');
```

14. Click the **Add…** drop-down menu and select **End Trigger**, and enter the expression:

```
Compare('TwinCATVersion','TwinCATVersion-Expected');
```

15. Select the "Event-Triggered" option for the **Scheduling** type and "Any Input" from the **Trigger On** drop-down menu.

Figure 2-38 Screenshot of PI System Explorer Displaying the TwinCAT Firmware Version Mismatch Analysis Template in the PLC Element Template



16. On the top menu bar click **Check In** , verify the changes in the dialog box and click **Check In**.

17. On the left navigation panel, select **Elements**.

18. In the navigation tree in the **Elements** panel, select **CRS-Connector > Workcell 1 > PLC.**

19. Open the **Attributes** tab in the PLC panel.

20. Select the attribute **HardwareSerialNumber-Expected** and enter the expected hardware serial number (e.g., 5870) in the **Value** textbox.

21. Select the attribute **TwinCATVersion-Expected** and enter the expected hardware serial number (e.g., 3.1.4022) in the **Value** textbox.

22. On the top menu bar and click **Check In**, verify the changes in the dialog box, and click **Check In**.

Event frames will now be generated in the DMZ Historian if the PLC reports a hardware serial number that does not match the expected value or if the TwinCAT firmware version number does not match the expected value.

2.7 Security Onion

Security Onion is a Linux-based, open source security playbook. It includes numerous security tools for intrusion detection, log management, incident response, and file integrity monitoring. For this project, the tool Wazuh was used in Builds 2 and 4 for file integrity checking. Wazuh works at the host-level to detect unusual and unauthorized activity and changes to file and software configurations. Security Onion and Wazuh use Elastic Stack components, Elasticsearch, Filebeat, and Kibana to store, search, and display alert data.

Note: Wazuh is a fork of the open source project OSSEC, a host-based intrusion detection system. In some places in Wazuh and this document, the term OSSEC will be used in place of Wazuh.

2.7.1 Host and Network Configuration

Wazuh is an agent-based software. For this project, an existing Security Onion server was used, and the Wazuh agent was installed on multiple endpoints in both the PCS and CRS environments. The tables below list the network configuration for the Security Onion server (Table 2-13) and the hosts (Table 2-14 and Table 2-15) with the installed agent.

Table 2-13 Security Onion Domain Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
Security Onion Server	Hyper-V VM	Ubuntu 16.04 LTS	4	16GB	450GB	Testbed LAN 10.100.0.26
Nessus VM	Hyper-V VM	Windows 2012R2	2	6GB	65GB	Testbed LAN 10.100.0.25
Dispel VDI	Hyper-V VM	Windows 2016	2	8GB	126GB	DMZ LAN 10.100.1.61
DMZ Historian	Hyper-V VM	Windows 2016	4	8GB	80GB/171GB	DMZ LAN 10.100.1.4

Table 2-14 Security Onion PCS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
PCS Engineering Workstation	HP Z230 Tower PC	Windows 7	4	16GB	465GB	PCS LAN 3 172.16.3.10
PCS HMI Host	Supermicro Z97X-Ud5H	Windows 7	4	8GB	600GB	PCS LAN 1 172.16.1.4

Table 2-15 Security Onion CRS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
CRS Engineering Workstation	Dell Precision T5610	Windows 10	8	16GB	465GB	CRS Supervisory 192.168.0.20

2.7.2 Installation

Security Onion Server version 3.9 and Wazuh Agent version 3.9 were used.

Installation of Wazuh involves setting up the central server and installing agents on hosts that needed to be monitored.

Security Onion server contains the Wazuh manager and API components as well as the Elastic Stack. The Wazuh manager is responsible for collecting and analyzing data from deployed agents. The Elastic Stack is used for reading, parsing, indexing, and storing alert data generated by the Wazuh manager.

The Wazuh agent, which runs on the monitored host, is responsible for collecting system log and configuration data and detecting intrusions and anomalies. The collected data is then forwarded to the Wazuh manager for further analysis.

The Security Onion server was already a part of the lab infrastructure prior to this effort. For the server component installation process, please follow the guidance from the Security Onion Installation Guide for version 3.9 available at https://documentation.wazuh.com/3.9/installation-guide/index.html.

For information on adding agents to the server, please follow the guidance from the Security Onion Installation Guide for version 3.9 available at https://documentation.wazuh.com/3.9/user-manual/registering/index.html.

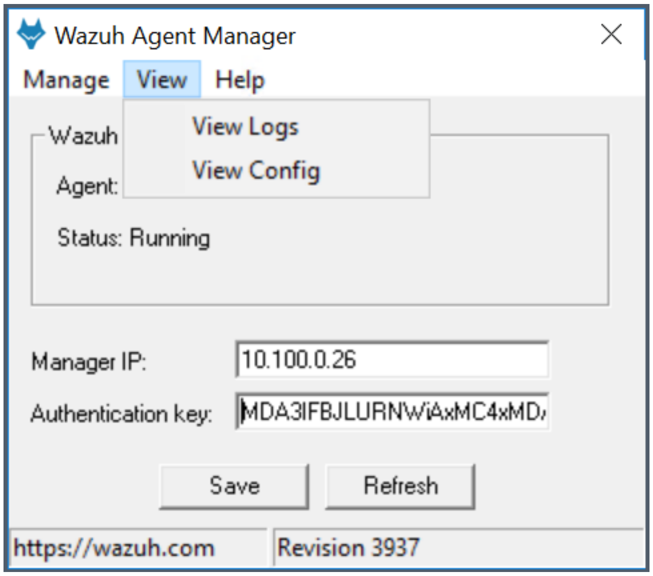
2.7.3 Configuration

1. Configure Additional Directories or Files for Wazuh Agent File Integrity Monitoring:

    a. Files and directories to be monitored are specified in the ossec.conf file on each host.

i. To view or edit this file, click the **View** tab in the Wazuh Configuration Manager on the host machine and select View Config as shown in Figure 2-39.

Figure 2-39 Wazuh Agent Manager



b. Selecting **View>View Config** opens the ossec.conf file in Notepad. Alternatively, the file can be opened in Notepad from its location in the "C:\Program Files (x86)\ossec-agent" directory on the host machine, as shown in Figure 2-40.

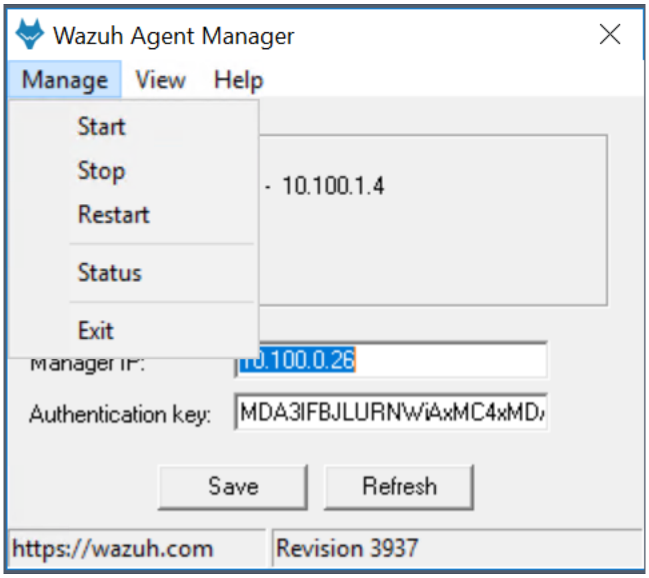
Figure 2-40 ossec.conf File

```
<!-- Directories added for NCCOE Project -->
<directories check_all="yes" whodata="yes">C:\testscenarios</directories>
<directories check_all="yes" whodata="yes">C:\EngWorkstation_Share</directories>
<directories check_all="yes" whodata="yes">C:\Program Files (x86)\ControlFLASH</directories>
<directories check_all="yes" whodata="yes">C:\Users\Administrator\Documents</directories>
<directories check_all="yes" whodata="yes">C:\Users\Administrator\Downloads</directories>

<ignore>%PROGRAMDATA%\Microsoft\Windows\Start Menu\Programs\Startup\desktop.ini</ignore>

<ignore type="sregex">.log$|.htm$|.jpg$|.png$|.chm$|.pnf$|.evtx$</ignore>
```

c. To add files or directories to the default configuration, copy and modify an existing line in the ossec.conf file to ensure the proper XML syntax is used.

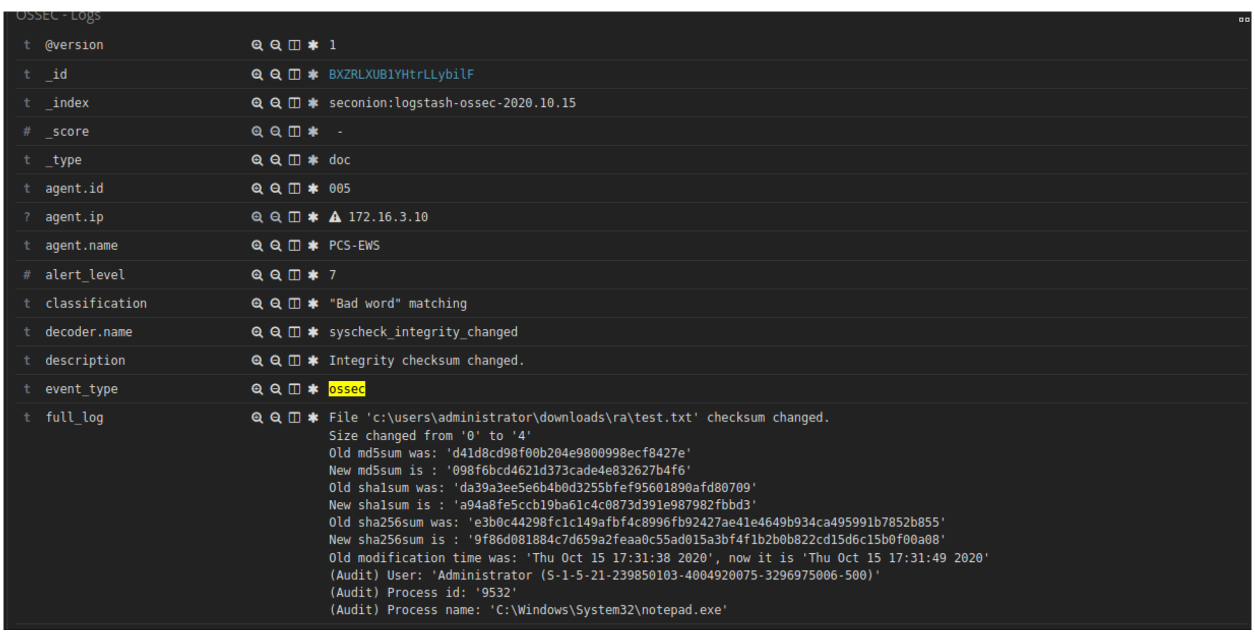
d. Once the changes are made, save the ossec.conf file and restart the Wazuh Agent by opening the Configuration Manager, selecting the **Manage** tab, and **Restart** as shown in Figure 2-41.

Figure 2-41 Wazuh Agent Manager User Interface



e. Changes to the files or directories specified in the ossec.conf file will be detected and sent to the Wazuh Manager. Figure 2-42 shows the log received after a file change was detected.

Figure 2-42 Log Received After a File Change Was Detected



2.8 TDi ConsoleWorks

The TDi ConsoleWorks implementation in Builds 1 and 3 consists of a single VM hosted on VMWare ESXi to meet the user authentication and authorization capabilities. ConsoleWorks provides a secure web interface through which authenticated and authorized users receive access to graphical and shell interfaces on configured ICS components.

2.8.1 Host and Network Configuration

ConsoleWorks resides on a VM that was reconfigured for supporting Builds 1 and 3 as described in Table 2-16 and Table 2-17 respectively.

Table 2-16 ConsoleWorks Build 1 Deployment

Name	System	OS	CPU	Memory	Storage	Network
ConsoleWorks	VMWare VM	CentOS 7	8x vCPU	8GB	500 GB 750 GB	Testbed LAN 10.100.0.53

Table 2-17 ConsoleWorks Build 3 Deployment

Name	System	OS	CPU	Memory	Storage	Network
ConsoleWorks	VMWare VM	CentOS 7	8x vCPU	8GB	500 GB 750 GB	CRS 192.168.0.65

2.8.2 Installation

ConsoleWorks version 5.3-1u3 is installed on a CentOS 7 operating system using the following procedures. Product installation guides and documentation are available at https://support.tditechnologies.com/product-documentation. Follow these steps for installation:

1. Harden and configure the operating system:

   a. Log in to the system with privileged access and set the Static IP Address information by editing */etc/sysconfig/network-scripts/ifcfg-eth0* using the following settings:

      i. For Build 1 use the following network configuration:

         1) IP Address: **10.100.0.53**

         2) Subnet Mask: **255.255.255.0**

         3) Gateway: **10.100.0.1**

         4) DNS: **10.100.0.17**

      ii. For Build 3 use the following network configuration:

         1) IP Address: **192.168.0.65**

2) Subnet Mask: **255.255.255.0**

3) Gateway: **192.168.0.2**

4) DNS: **10.100.0.17**

iii. Restart the network service as follows:

```
# systemctl restart network
```

b. Set the NTP Configuration as follows:

i. In */etc/ntp.conf,* add as the first server entry:

```
server 10.100.0.15
```

c. Apply the following Department of Defense (DOD) Security Technology Implementation Guide (STIG) settings:

i. Ensure ypserv is not installed using the following command:

```
# yum remove ypserv
```

ii. Ensure Trivial File Transfer Protocol (TFTP) is not installed using the following command:

```
# yum remove tftp-server
```

iii. Ensure RSH-SERVER is not installed using the following command:

```
# yum remove rsh-server
```

iv. Ensure File Transfer Protocol (FTP) is not installed using the following command:

```
# yum remove vsftpd
```

v. Ensure TELNET-SERVER is not installed using the following command:

```
# yum remove telnet-server
```

vi. Configure SSH to use SSHv2 only.

1) To disable SSHv1, ensure only Protocol 2 is allowed in the /etc/ssh/sshd_config.

```
Protocol 2
PermitRootLogin no
Ciphers aes128-ctr, aes192-ctr, aes256-ctr, aes128-
cbc
MACs hmac-sha2
```

vii. Disallow authentication using an empty password as follows:

1) Add **PermitEmptyPasswords no** to /etc/ssh/sshd_config file.

2) Remove any instances of the **nullok** option in /etc/pam.d/system-auth and /etc/pam.d/password-auth files.

viii. Enable FIPS Mode as follows:

1) FIPS mode can be enabled by running the command:

```
# yum install dracut
# dracut -f
```

2) When step 1) is complete, add **fips=1** to the /etc/default/grub file and run the command:

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

3) When step 2) completes, reboot the server with this command:

```
# reboot
```

ix. Enable server auditing

1) Ensure events on the server are being recorded for investigation in the event of an outage or attack. This can be enabled by running the command:

```
# systemctl start auditd.service.
```

x. Configure system to only install approved digitally signed packages:

1) Configure yum to verify the Certificate Authority is from an approved organization. To enable this, ensure that **gpgcheck=1** is in the /etc/yum.conf file.

xi. Enable the firewall:

1) To enable the firewall, run the following commands:

```
# yum install firewalld and
# systemctl start firewalld.
```

2) Check Firewall Zone and confirm only SSH and HTTPS is allowed. Note: the default zone is Public and SSH is already permitted. For the implementation, we checked the configuration using the following command:

```
# firewall-cmd --list-all
```

3) Add the HTTPS configuration to the firewall using the following command:

```
# firewall-cmd --zone=public --permanent --add-
service=https
```

xii. Enable SELinux and set to "targeted":

1) Add SELINUX=enforcing and SELINUXTYPE=targeted in the /etc/selinux/config file and then reboot the server with this command:

```
# reboot
```

xiii. Enable Antivirus as follows:

1) ClamAV is used for the lab implementation using the following commands adapted from information found on https://www.clamav.net/documents/clam-antivirus-user-manual:

```
# yum install -y epel-release
```

```
# yum -y install clamav-server clamav-data
clamav-update clamav-filesystem clamav clamav-
scanner-systemd clamav-devel clamav-lib clamav-
server-systemd
```

2) Update SELinux policy to allow ClamAV to function

 # setsebool -P antivirus_can_scan_system 1

3) Make a backup copy of the scan.conf file and update to remove the Example string from the file using these commands:

 # cp /etc/clamd.d/scan.conf /etc/clamd.d/scan.conf.bk

 # sed -i '/^Example/d' /etc/clamd.d/scan.conf

4) Uncomment the following line from /etc/clamd.d/scan.conf:

LocalSocket /var/run/clamd.scan/clamd.sock

5) Configure freshclam to automatically download updated virus definitions using these commands:

 # cp /etc/freshclam.conf /etc/freshclam.conf.bak

 # sed -i -e "s/^Example/#Example/" /etc/freshclam.conf

6) Manually run freshclam to confirm the settings as follows:

 # freshclam

7) Start and enable the clamd service with these commands:

 # systemctl start clamd@scan

 # systemctl enable clamd@scan

8) Ensure log directory is available with this command:

 # mkdir /var/log/clamav

9) Create the daily scan script to scan directories of interest. Note: for the lab implementation only the /home volume was selected for scanning.

 # vi /etc/cron.daily/clamav_scan.sh

```
File Contents

#!/bin/bash
SCAN_DIR="/home"
LOG_FILE="/var/log/clamav/dailyscan.log"
/usr/bin/clamscan -ri $SCAN_DIR >> $LOG_FILE
```

10) Set the file to have execute privilege with this command:

```
# chmod +x /etc/cron.daily/clamav_scan.sh
```

2. Download and Install the ConsoleWorks packages
   a. Login to TDi Technology Support Portal (https://support.tditechnologies.com/get_con-soleworks) to download the ConsoleWorks for Linux 5.3-1u3 installation package. Cre-dentials will be provided by TDi.
   b. After downloading the ConsoleWorks installation package, copy it to the ConsoleWorks VM using a Secure Copy (scp) utility.
   c. Follow the procedures from TDi ConsolWorks New Installation and Upgrade Guide for Linux Chapter 3: Automated New Installation of ConsoleWorks
      i. During installation, create a New Invocation named "NCCOE".
      ii. Create a new certificate.
      iii. Set the system to automatically start the ConsoleWorks Invocation.
   d. Login to the platform and initiate the offline registration process (Figure 2-43).
   e. Once the license file is obtained, complete the registration process (Figure 2-44).

Figure 2-43 ConsoleWorks Registration Screen

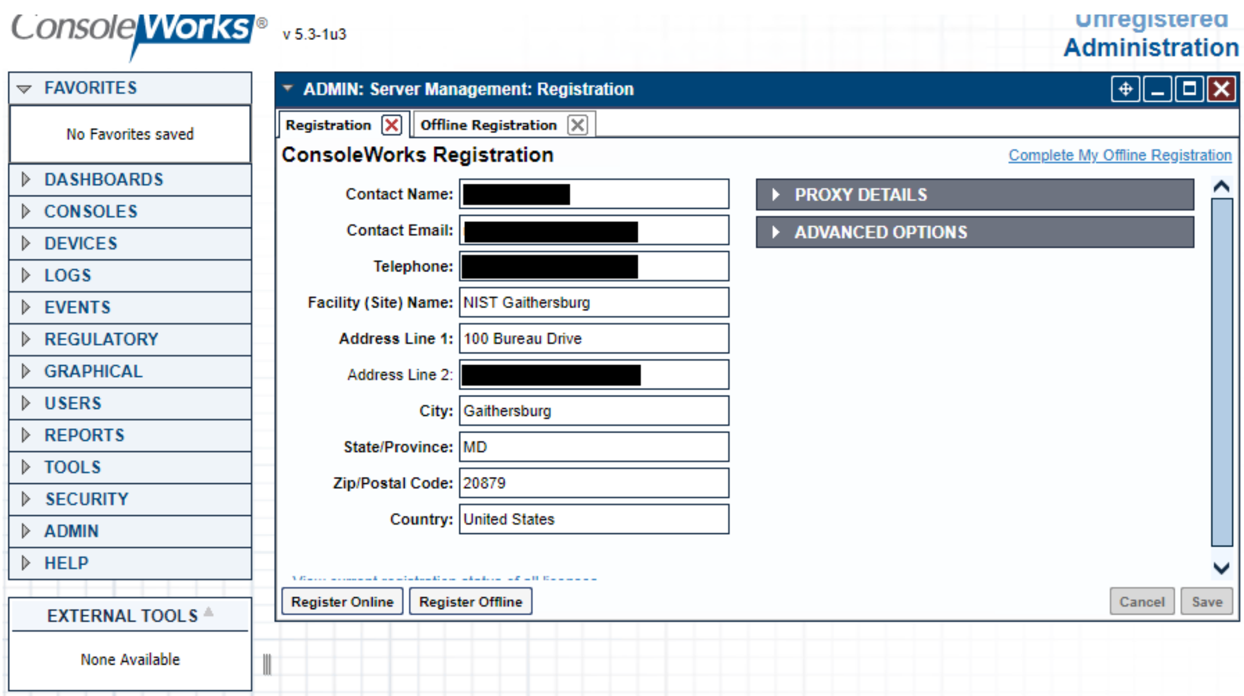
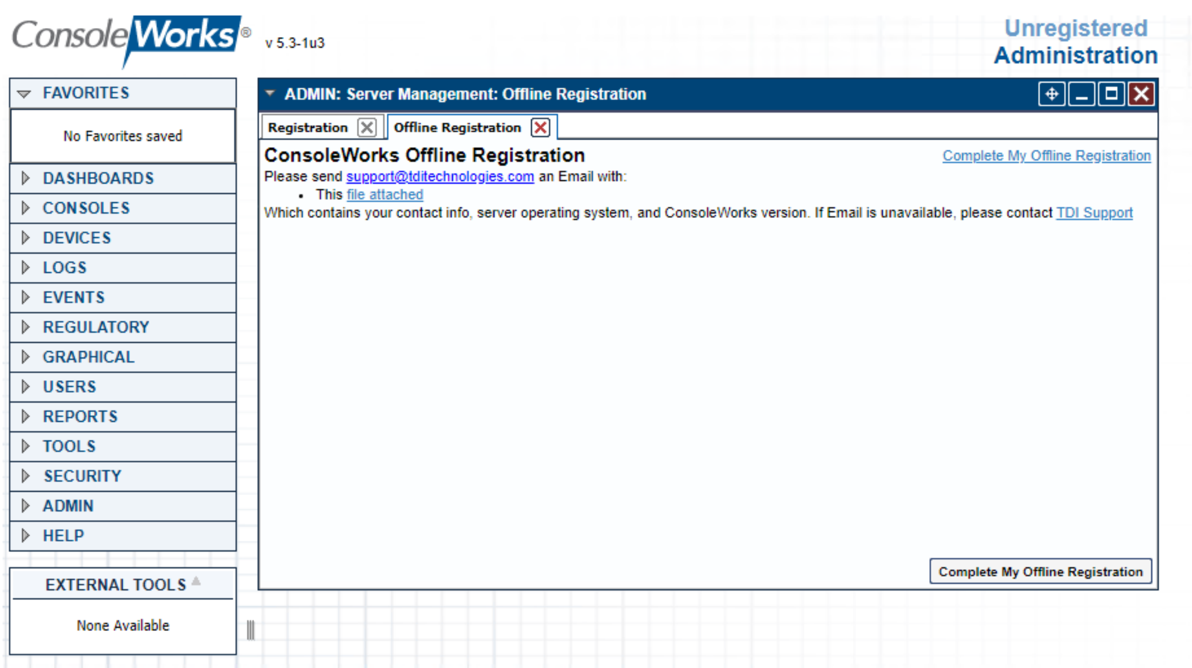


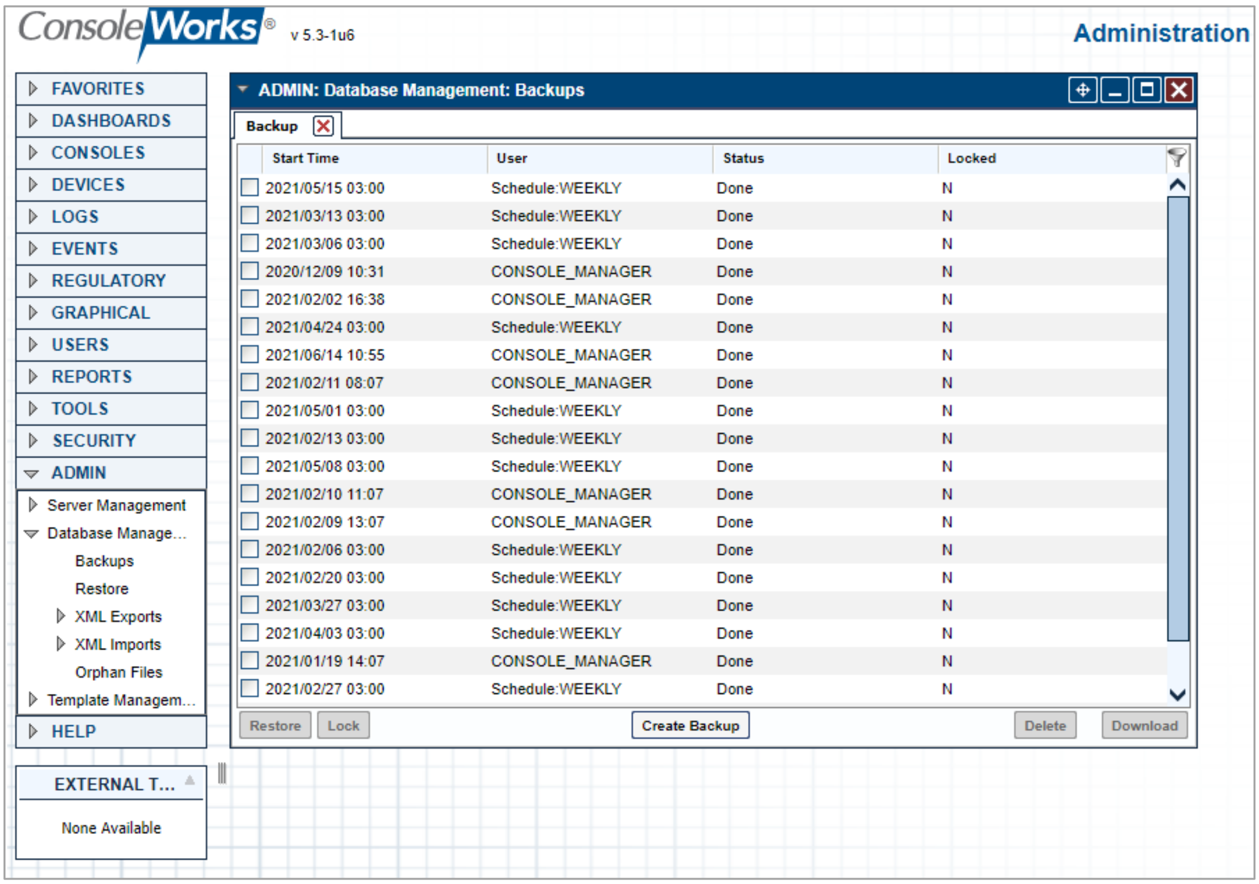
Figure 2-44 ConsoleWorks Offline Registration Process



f. This completes the default installation and establishes a basic ConsoleWorks server configuration. For the lab implementation, ConsoleWorks support provided two additional add-on packages (XML) files to setup the environment: ONBOARDING_1-DASHBOARDS_NCCoE.zip providing preconfigured dashboards for accelerating configurations; and NCCOE_ACRs_20210122_083645.zip providing the access control rules, tags, and automation scripts used for the dashboards. These packages are scheduled for inclusion in future releases or can be requested from ConsoleWorks.

i. Prior to installing these packages, a backup of the configuration should be made (Figure 2-45) by accessing **Admin > Database Management > Backups** and clicking **Create Backup**.

Figure 2-45 ConsoleWorks System Backups

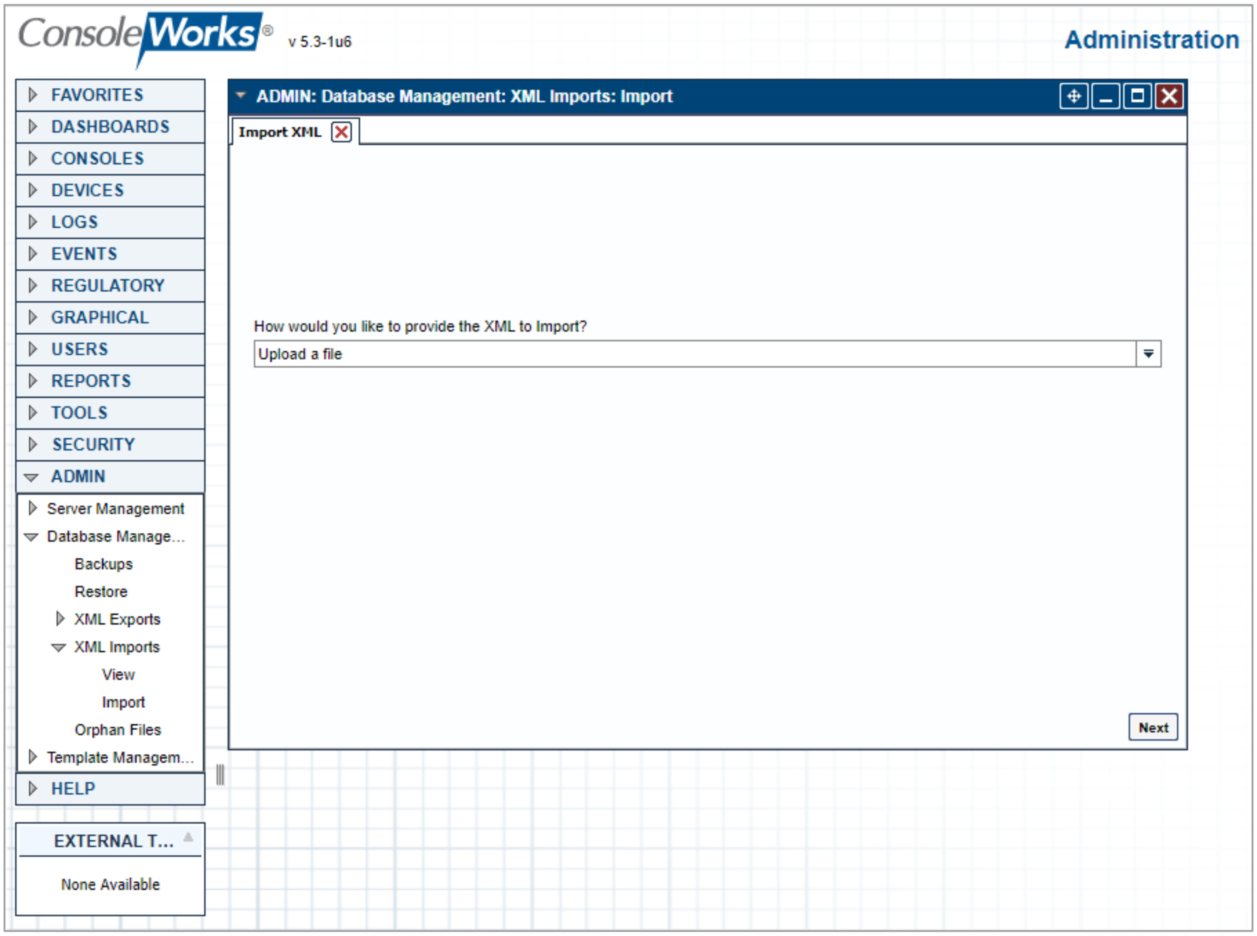


ii. Perform the XML Imports (Figure 2-46) by accessing **Admin > Database Management > XML Imports** following these steps:

1) Import the *Dashboard Add-On XML* file.

2) Import the *Supporting Configuration Add-On* XML file.

Figure 2-46 ConsoleWorks Importing System Configurations and Components



2.8.3 Configuration

The ConsoleWorks implementation required the following changes to the lab Cisco VPN appliance to allow remote users to access the ConsoleWorks system:

1. Login to the Cisco Firepower Appliance.
2. Create the Following Destination Network Objects:
    a. For Build 1:
        i. Name: ConsoleWorks
        ii. IP Address: 10.100.0.52
    b. For Build 3:
        i. Name: CRS-NAT-IP
        ii. IP Address: 10.100.0.20
3. Create the Following VPN-Rule:

a. For Build 1:

i. Action: Allow

ii. Source Networks: VPN-Pool

iii. Destination Networks: ConsoleWorks

iv. Destination Ports: TCP (6): 5176; HTTPS

b. For Build 3:

i. Action: Allow

ii. Source Networks: VPN-Pool

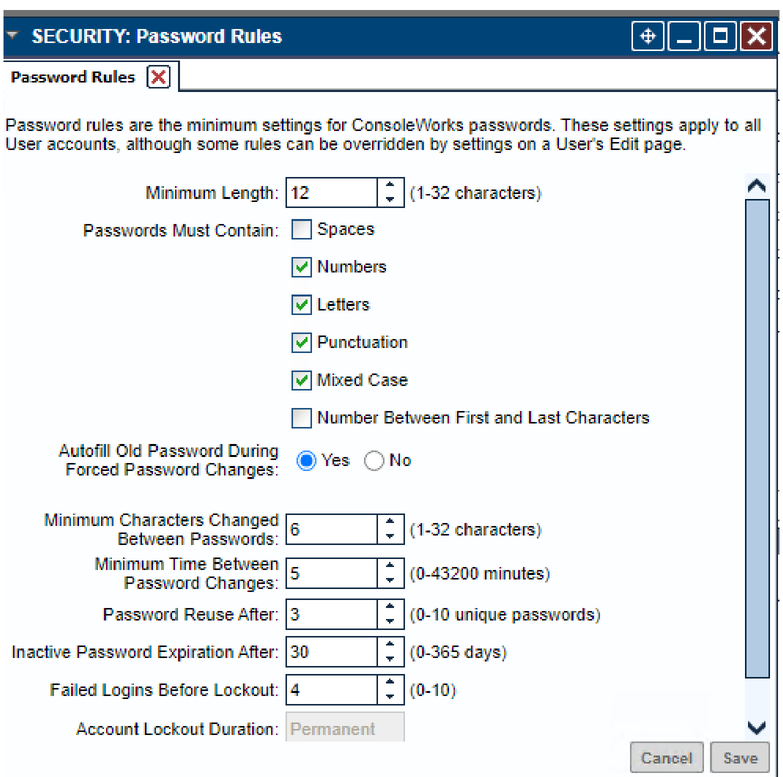
iii. Destination Networks: CRS-NAT-IP

iv. Destination Ports: TCP (6): 5176; HTTPS

ConsoleWorks is then configured as follows. For configuration procedures, please see the ConsoleWorks documentation available at https://support.tditechnologies.com/product-documentation.

1. Configure ConsoleWorks **Password Rules** (Figure 2-47):

Figure 2-47 ConsoleWorks Password Settings



2. Add user accounts:

a. **NCCOE_ADMIN**

b. **NCCOE_USER**

3. Configure the Graphical Gateway to allow users to use RDP within ConsoleWorks following these steps (Figure 2-48):

    a. Name: **LOCAL_GG**

    b. Description: **Local GUI Gateway**

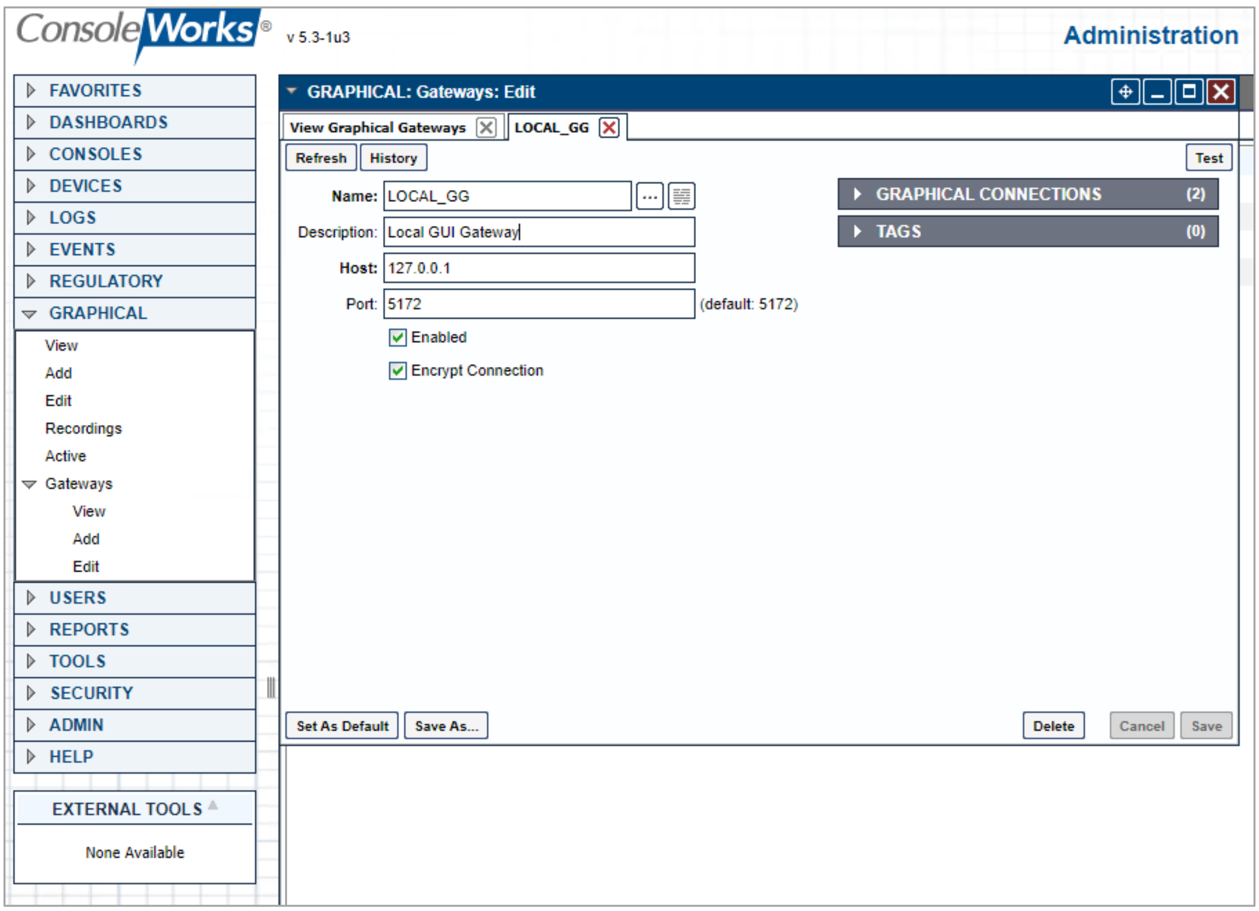
    c. Host: **127.0.0.1**

    d. Port: **5172**

    e. Enabled: **Selected**

    f. Encrypt Connection: **Selected**

Figure 2-48 ConsoleWorks Add the Local Graphical Gateway for RDP Access



4. Configure Device Types to organize the registered devices within the system as follows:

    a. Enter the information for the supported device types as shown in the example device type (Figure 2-49) for each type listed in Table 2-18 (and shown in Figure 2-50).

Table 2-18 ConsoleWorks Device Type List

Name	Description	Parent Device Type	Order
NETWORKING	Devices supporting networked communications		1
IT_FWROUTER	Network Router/Firewall for supporting IT Communications	NETWORKING	1
IT_SWITCH	Network switch supporting IT communications	NETWORKING	1
OT_FWROUTER	ICS Firewall/Router for ICS Network Separation	NETWORKING	1
OT_SWITCH	ICS Switch for supporting OT Subnets	NETWORKING	1
SERVERS	Devices for providing one or more IT/OT Services		1
IT_SERVERS	Servers providing IT Services	SERVERS	1
OT_SERVERS	Servers providing OT Services	SERVERS	1
WORKSTATIONS	Computers used to support IT/OT Operations		1
HMI	Specialized workstation supporting human-machine interfaces	WORKSTATIONS	1
IT_WORKSTATIONS	Computers used by users to support IT Operations	WORKSTATIONS	1
OT_WORKSTATIONS	Computers used by users to support OT Operations	WORKSTATIONS	1

Figure 2-49 ConsoleWorks Example Device Type Definition

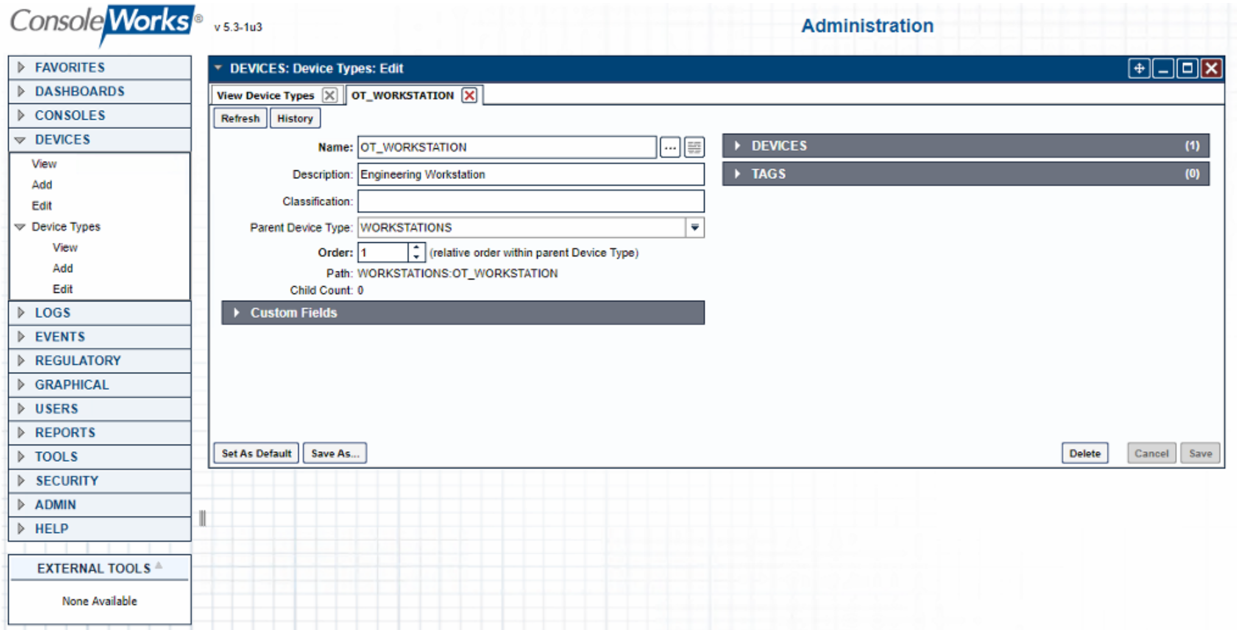
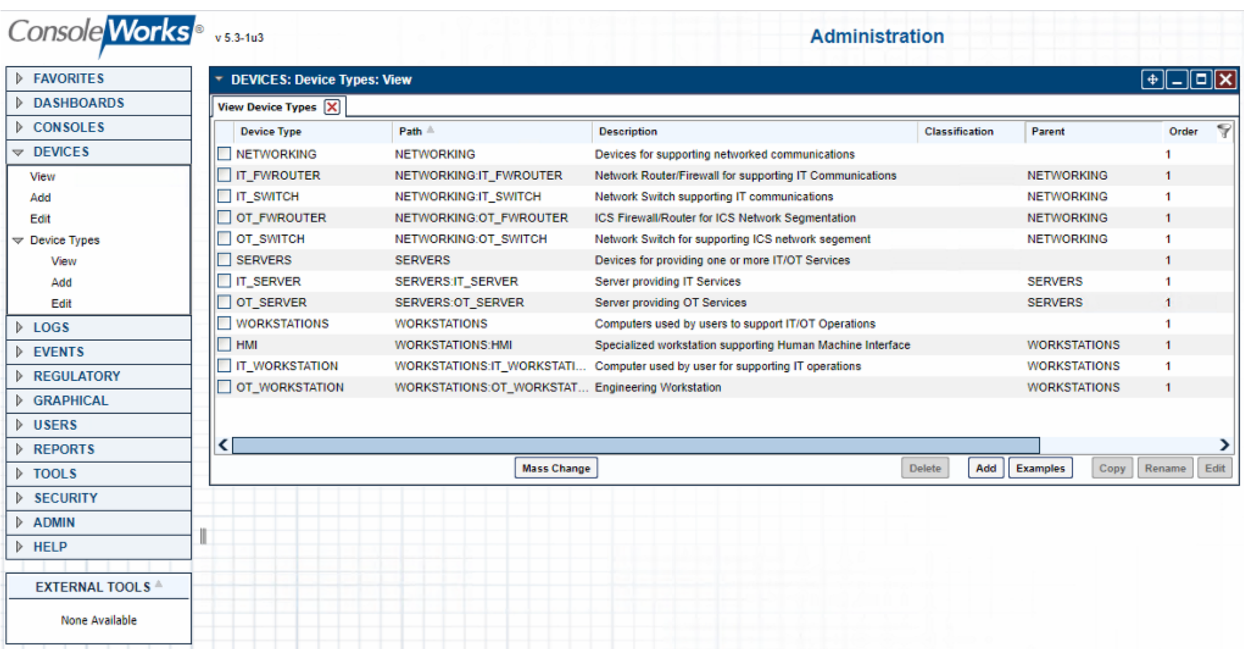
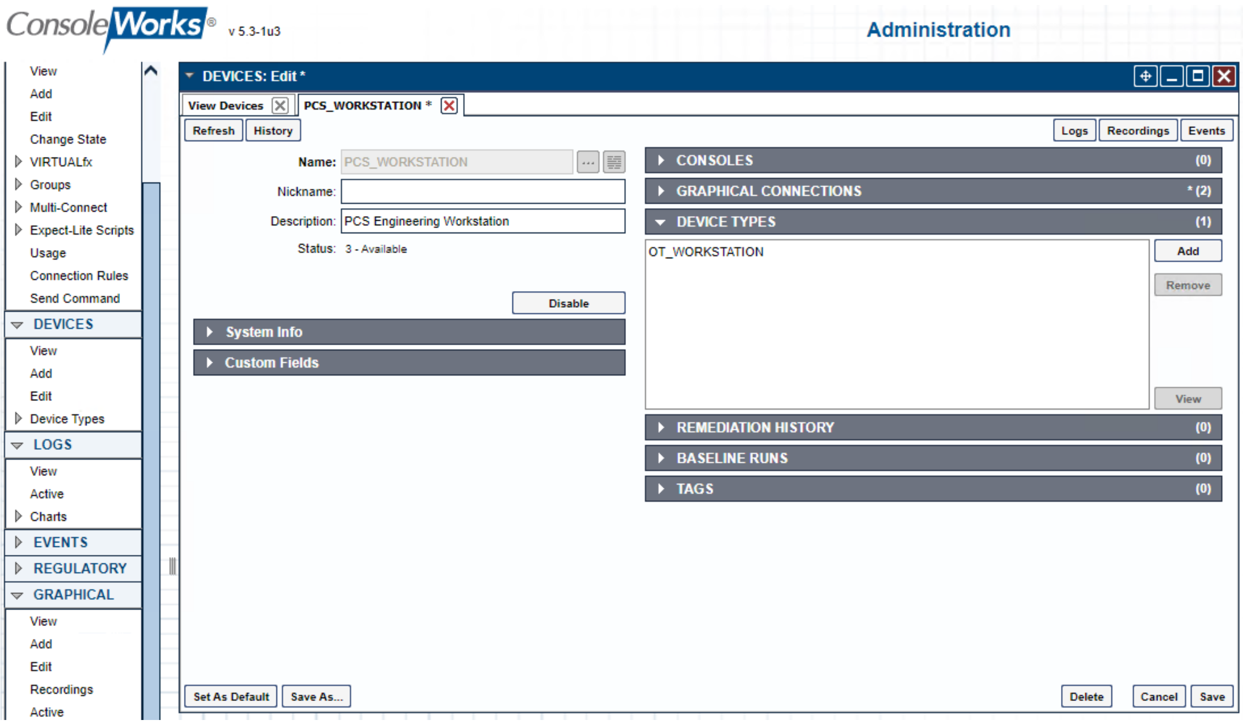


Figure 2-50 ConsoleWorks List of Device Types



5. Configure Devices for each system within the testbed that is accessible from ConsoleWorks.

Figure 2-51 ConsoleWorks Example Device Definition

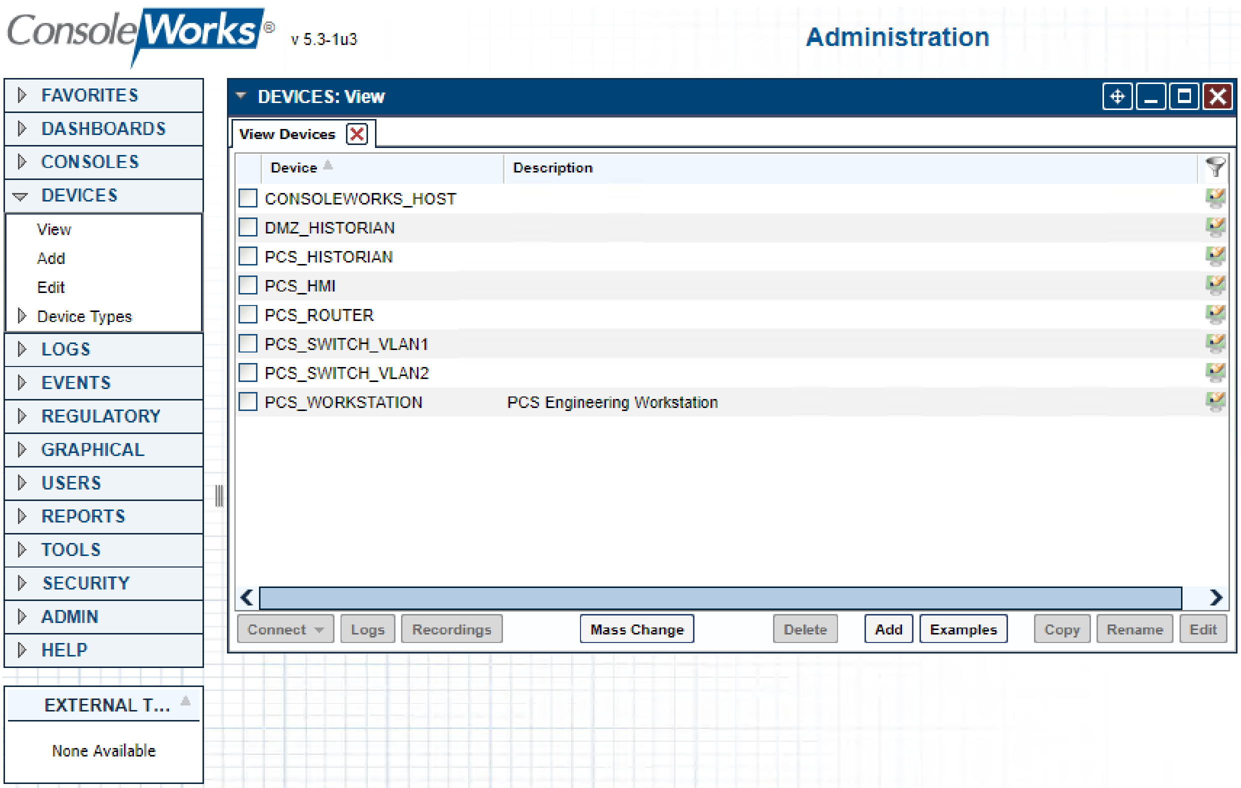


a. For Build 1 (PCS), enter the information for the devices as shown in the example device (Figure 2-51) for each device listed in Table 2-19 (Figure 2-52).

Table 2-19 ConsoleWorks PCS (Build 1) Devices

Name	Description	Device Type
DMZ_HISTORIAN	Historian in DMZ Subnet	IT_SERVER
PCS_HISTORIAN	Local Historian in PCS Subnet	OT_SERVER
PCS_HMI	PCS HMI Workstation	HMI
PCS_ROUTER	PCS Boundary Firewall/Router	OT_FWROUTER
PCS_SWITCH_VLAN1	PCS VLAN 1 OT Switch	OT_SWITCH
PCS_SWITCH_VLAN2	PCS VLAN 2 OT Switch	OT_SWITCH
PCS_WORKSTATION	PCS Engineering Workstation	OT_WORKSTATIONS

Figure 2-52 ConsoleWorks List of PCS (Build 1) Devices

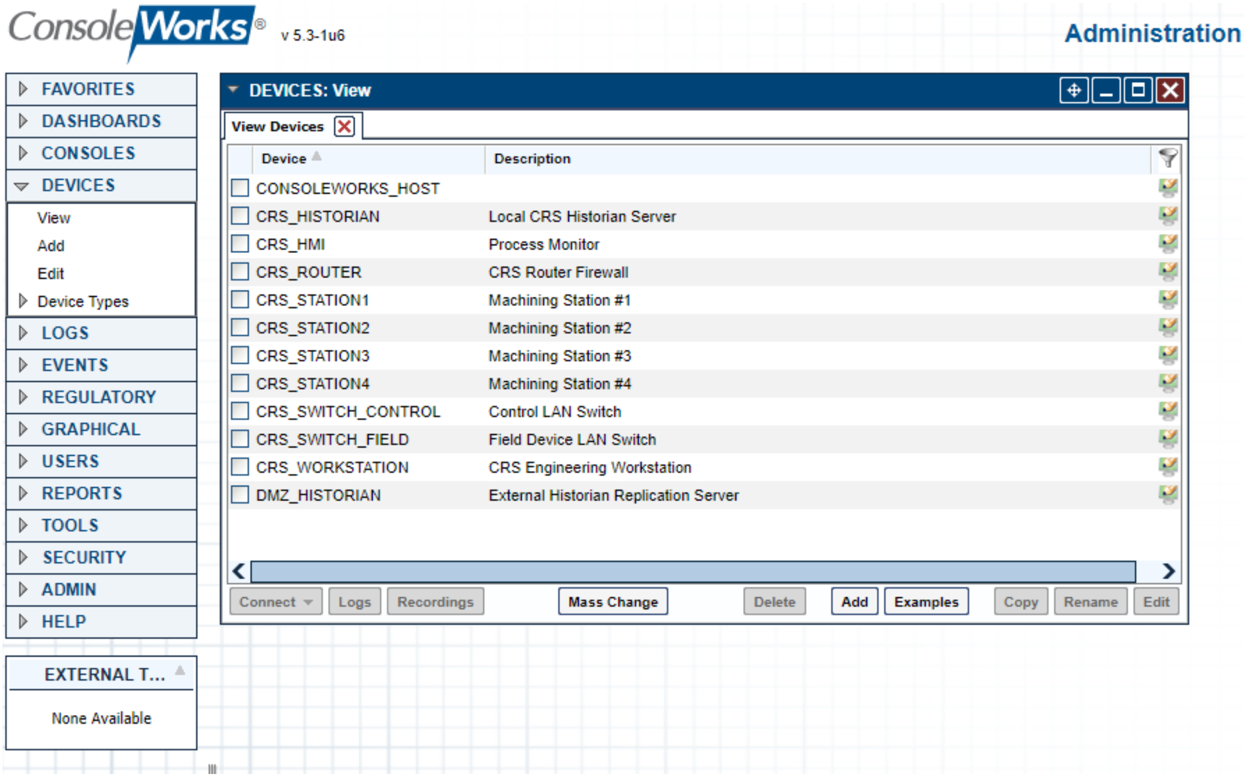


b. For Build 3 (CRS) , enter the information for the devices as shown in the example device (Figure 2-51) for each device listed in Table 2-20 (also shown in Figure 2-53).

Table 2-20 ConsoleWorks CRS (Build 3) Devices

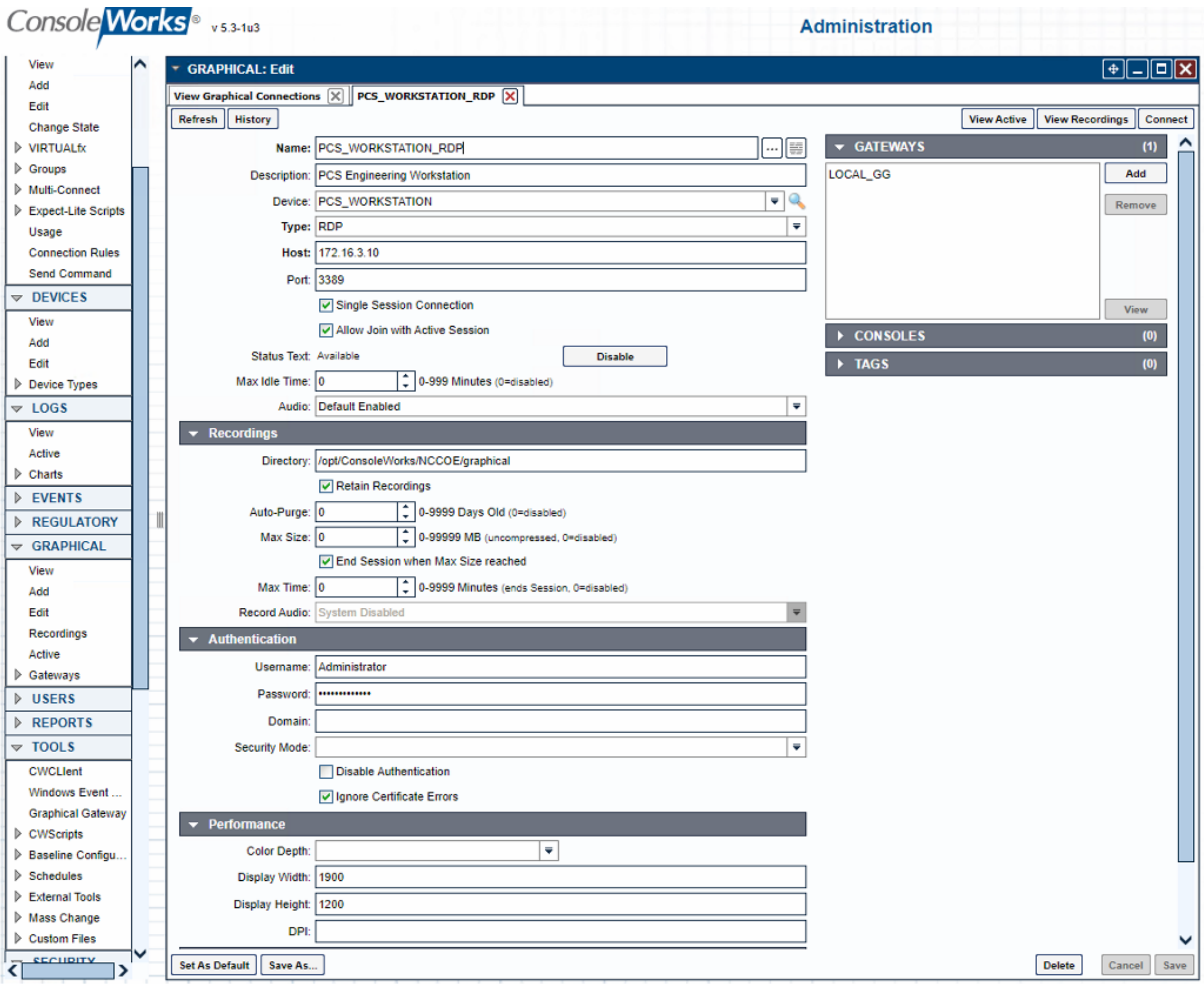
Name	Description	Device Type
DMZ_HISTORIAN	Historian in DMZ Subnet	IT_SERVER
CRS_HISTORIAN	Local Historian in CRS Subnet	OT_SERVER
CRS_HMI	CRS HMI Workstation	HMI
CRS_ROUTER	CRS Boundary Firewall/Router	OT_FWROUTER
CRS_SWITCH_CONTROL	OT Switch for Control Network	OT_SWITCH
CRS_SWITCH_FIELD	OT Switch for Field Network	OT_SWITCH
CRS_WORKSTATION	CRS Engineering Workstation	OT_WORKSTATIONS
CRS_STATION1	Machining Station #1	OT_WORKSTATIONS
CRS_STATION2	Machining Station #2	OT_WORKSTATIONS
CRS_STATION3	Machining Station #3	OT_WORKSTATIONS
CRS_STATION4	Machining Station #4	OT_WORKSTATIONS

Figure 2-53 ConsoleWorks List of CRS (Build 3) Devices



6. Configure Graphical Connections for the PC (RDP) based devices.

Figure 2-54 ConsoleWorks Example RDP Configuration



a. For Build 1 (PCS), enter the information for the Graphical Connections as shown in the example (Figure 2-54) for each graphical connection listed in Table 2-21 (also shown in Figure 2-55). For each entry, the following are common settings for all graphical connections:

   i. Under Gateway, click Add and select LOCAL_GG.

   ii. Single Session Connection: Checked

   iii. Allow Join with Active Session: Checked

   iv. Under Recordings:

      1) Directory: **/opt/ConsoleWorks/NCCOE/graphical**

      2) Retain Records: **Checked**

      3) Auto-Purge: **0**

4) Max Size: **0**

5) End Session when Max Size Reached: **Checked**

6) Max Time: **0**

v. Authentication

1) Specify local or domain credentials, which are securely stored by ConsoleWorks, to allow complex passwords/credentials without having to share between users.

2) Ignore Certificate Errors: Checked only if self-signed certificates are in use.

vi. Performance

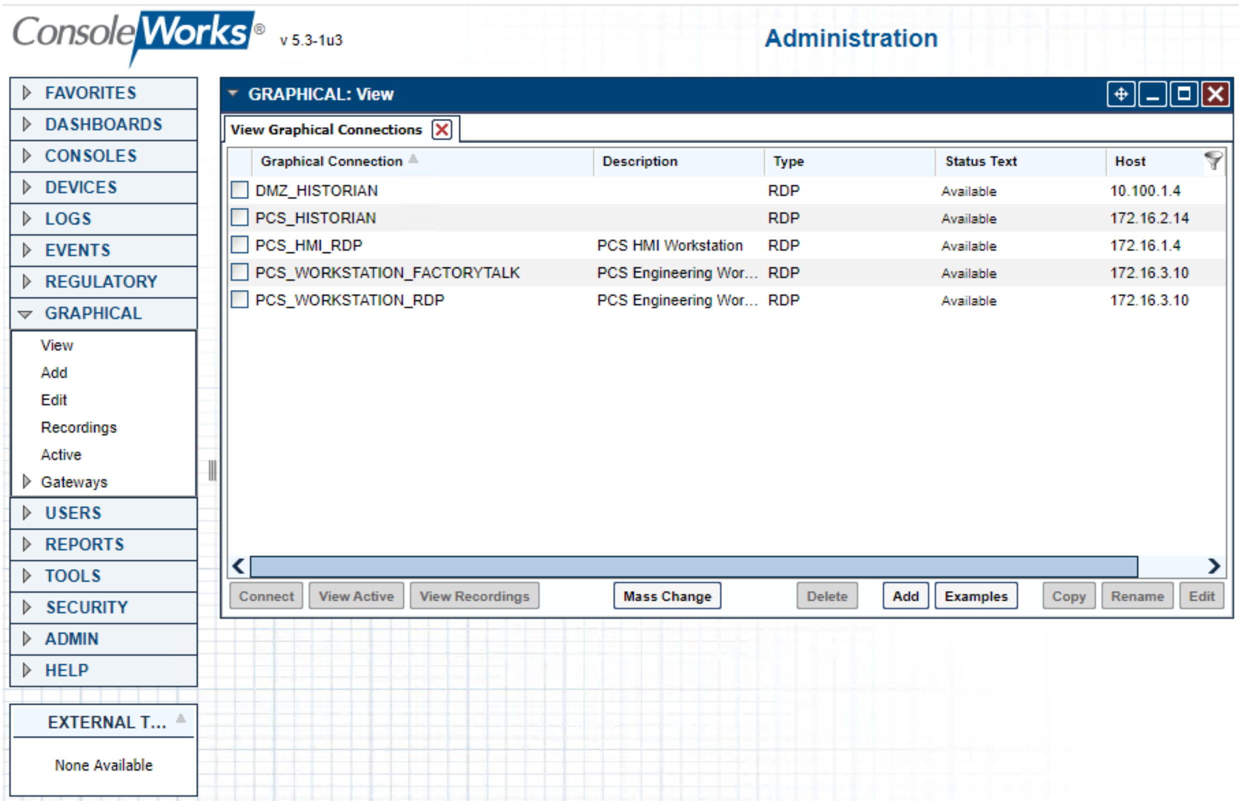
1) Display Width: **1900**

2) Display Height: **1200**

Table 2-21 ConsoleWorks PCS (Build 1) Graphical Connections

Name	Device	Type	Host	Port
DMZ_HISTORIAN	DMZ_HISTORIAN	RDP	10.100.1.4	3389
PCS_HISTORIAN	PCS_HISTORIAN	RDP	172.16.2.14	3389
PCS_HMI_RDP	PCS_HMI	RDP	172.16.2.4	3389
PCS_WORKSTATION_RDP	PCS_WORKSTATION	RDP	172.16.3.10	3389

Figure 2-55 ConsoleWorks List of PCS (Build 1) RDP Connections



b. For Build 3 (CRS), enter the information for the graphical connections as shown in the example (Figure 2-54) for each graphical connection listed in Table 2-22 (also shown in Figure 2-56). For each entry, the following are common settings for all graphical connections.

   i. Under Gateway, click **Add** and select **LOCAL_GG**.

   ii. Under Recordings, use these settings:

      1) Directory **/opt/ConsoleWorks/NCCOE/graphical**

      2) Retain Records **Checked**

      3) Auto-Purge: **0**

      4) Max Size: **0**

      5) End Session when Max Size Reached: **Checked**

      6) Max Time: **0**

   iii. Authentication:

      1) Specify local or domain credentials, which are securely stored by ConsoleWorks, to allow complex passwords/credentials without having to share between users.

iv. Performance

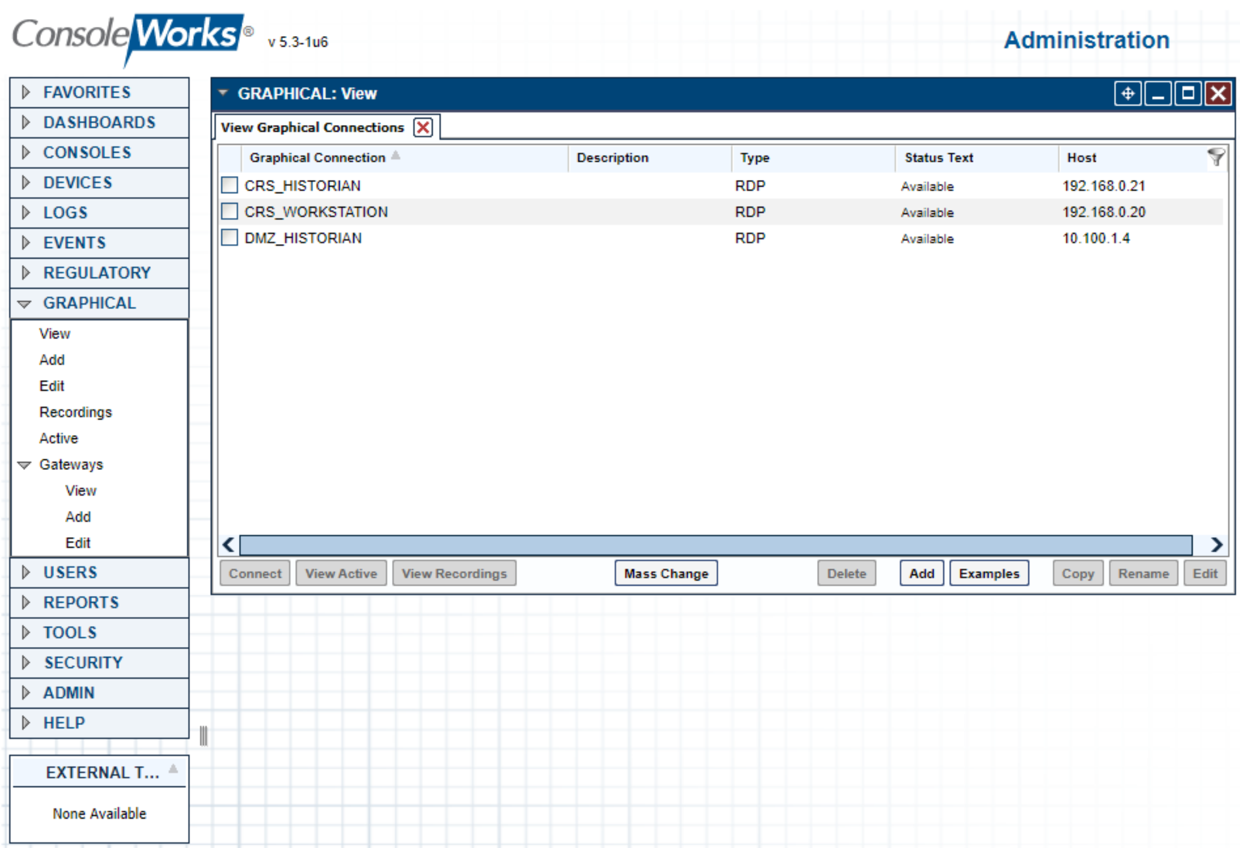
1) Display Width: **1900**

2) Display Height: **1200**

Table 2-22 ConsoleWorks CRS (Build 3) Graphical Connections

Name	Device	Type	Host	Port
DMZ_HISTORIAN	DMZ_HISTORIAN	RDP	10.100.1.4	3389
CRS_HISTORIAN	CRS_HISTORIAN	RDP	192.168.0.21	3389
CRS_WORKSTATION	CRS_WORKSTATION	RDP	192.168.0.20	3389

Figure 2-56 ConsoleWorks List of CRS (Build 3) RDP Connections



7. Configure console connections for non-graphical (e.g., SSH) interfaces to devices (Figure 2-57).

Figure 2-57 ConsoleWorks Example Console (SSH) Connection

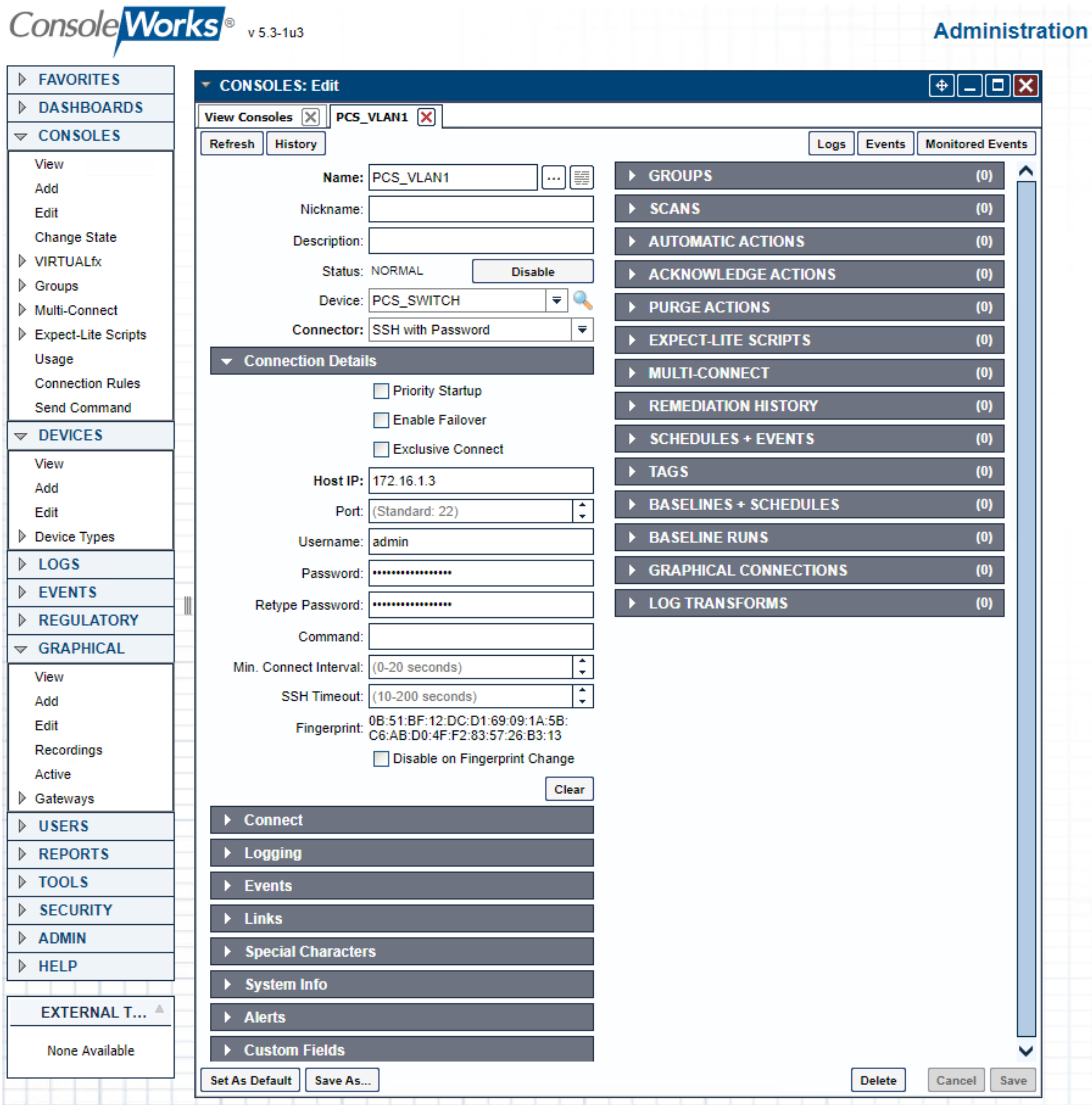
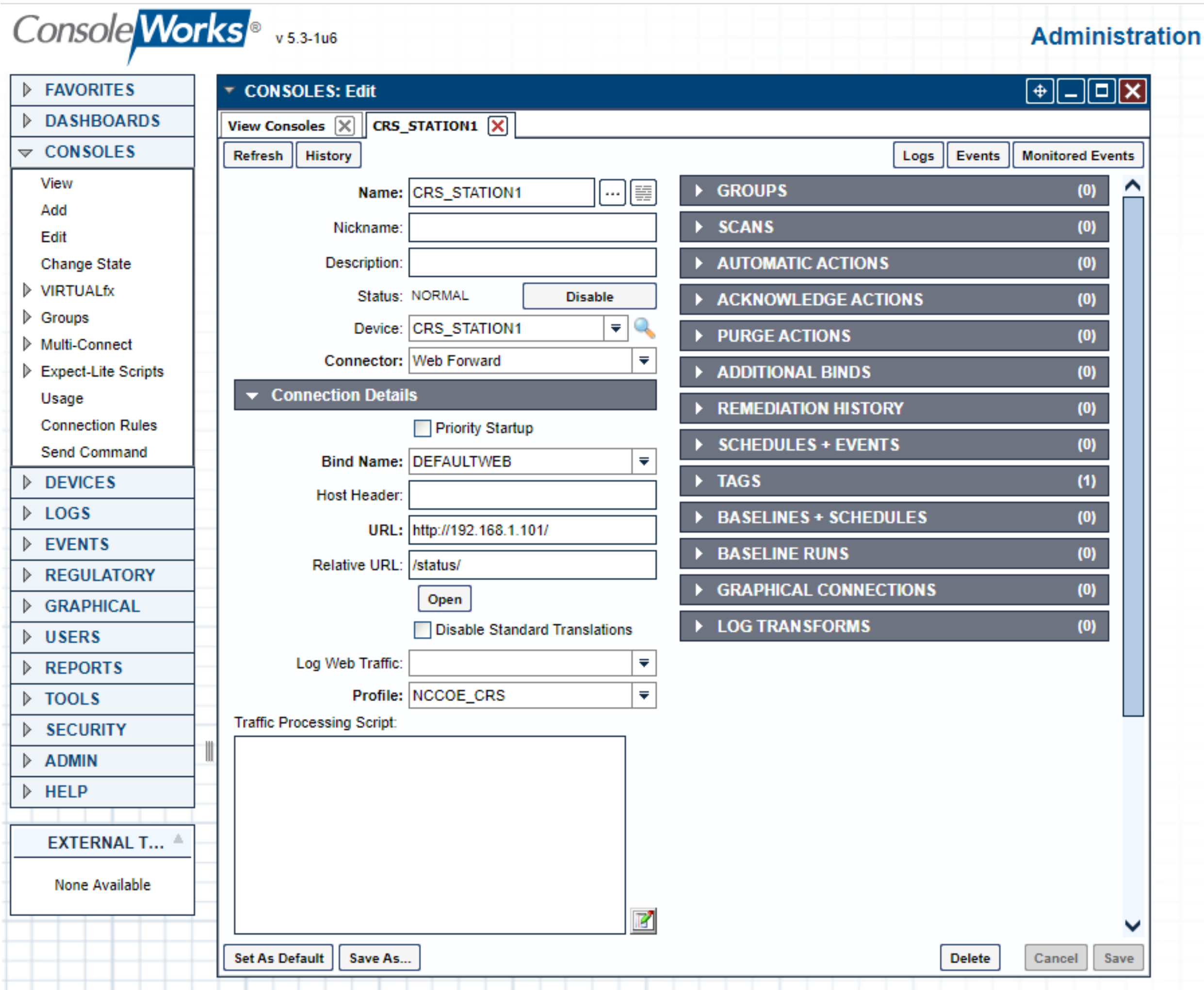


Figure 2-58 ConsoleWorks Example Console (Web Forward) Connection



a. For Build 1 (PCS), enter the information for the Console Connections as shown in the examples (Figure 2-57 and Figure 2-58) for each console connection listed in Table 2-23 (also shown in Figure 2-59). For each entry, the following are common settings for all console connections.

   i. Under **Connection Details**:

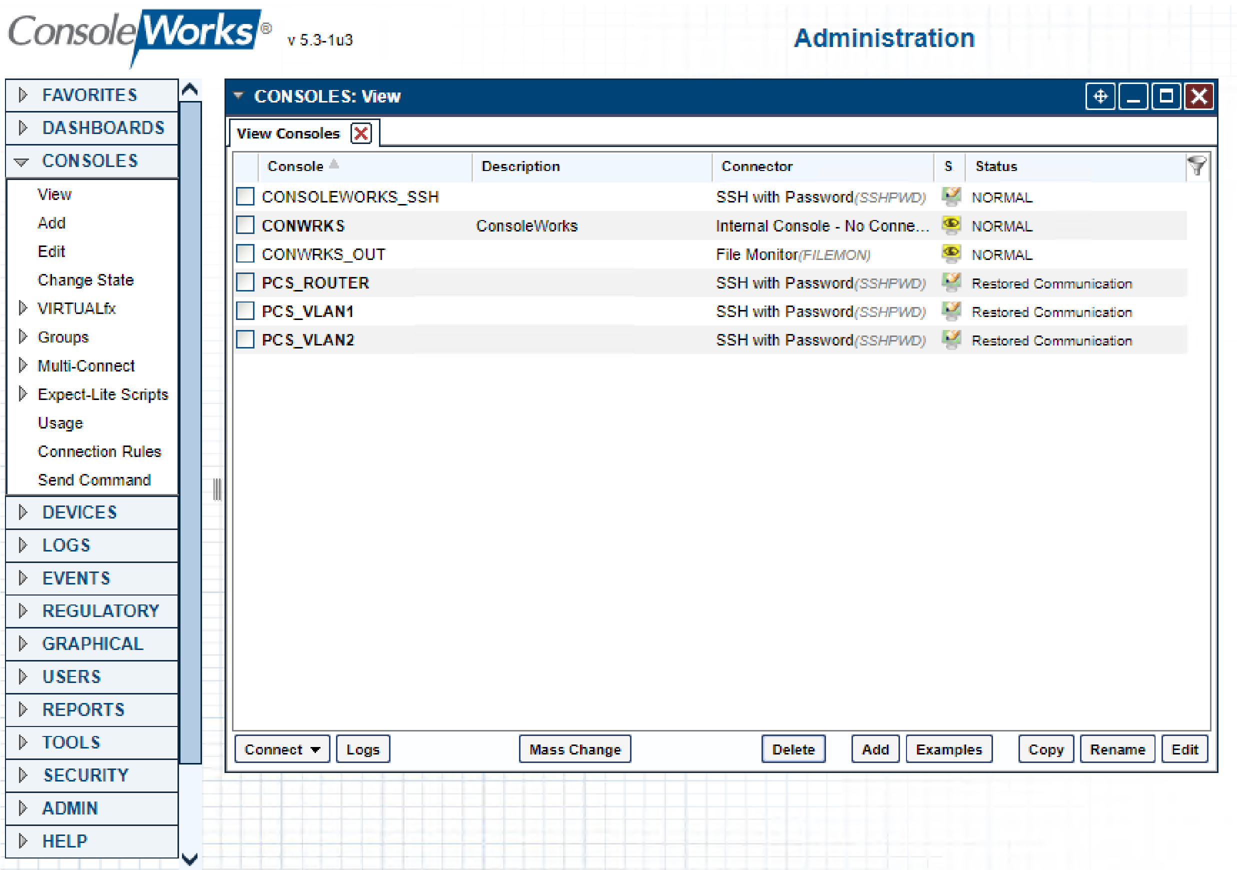
      1) Specify the username and password, which are securely stored by ConsoleWorks, to allow complex passwords/credentials without having to share between users.

Table 2-23 ConsoleWorks PCS (Build 1) Console Connections

Name	Device	Connector	Host	Port
PCS_ROUTER	PCS_ROUTER	SSH with Password	10.100.2.8	22
PCS_VLAN1	PCS_SWITCH_VLAN1	SSH with Password	172.16.1.3	22

Name	Device	Connector	Host	Port
PCS_VLAN2	PCS_SWITCH_VLAN2	SSH with Password	172.16.2.2	22

Figure 2-59 ConsoleWorks List of PCS (Build 1) Console Connections



b. For Build 3 (CRS), enter the information for the console connections as shown in the example (Figure 2-57 and Figure 2-58) for each console connection listed in Table 2-24 (Figure 2-60). For each entry, the following are common settings for all console connections.

   i. Under **Connection Details**

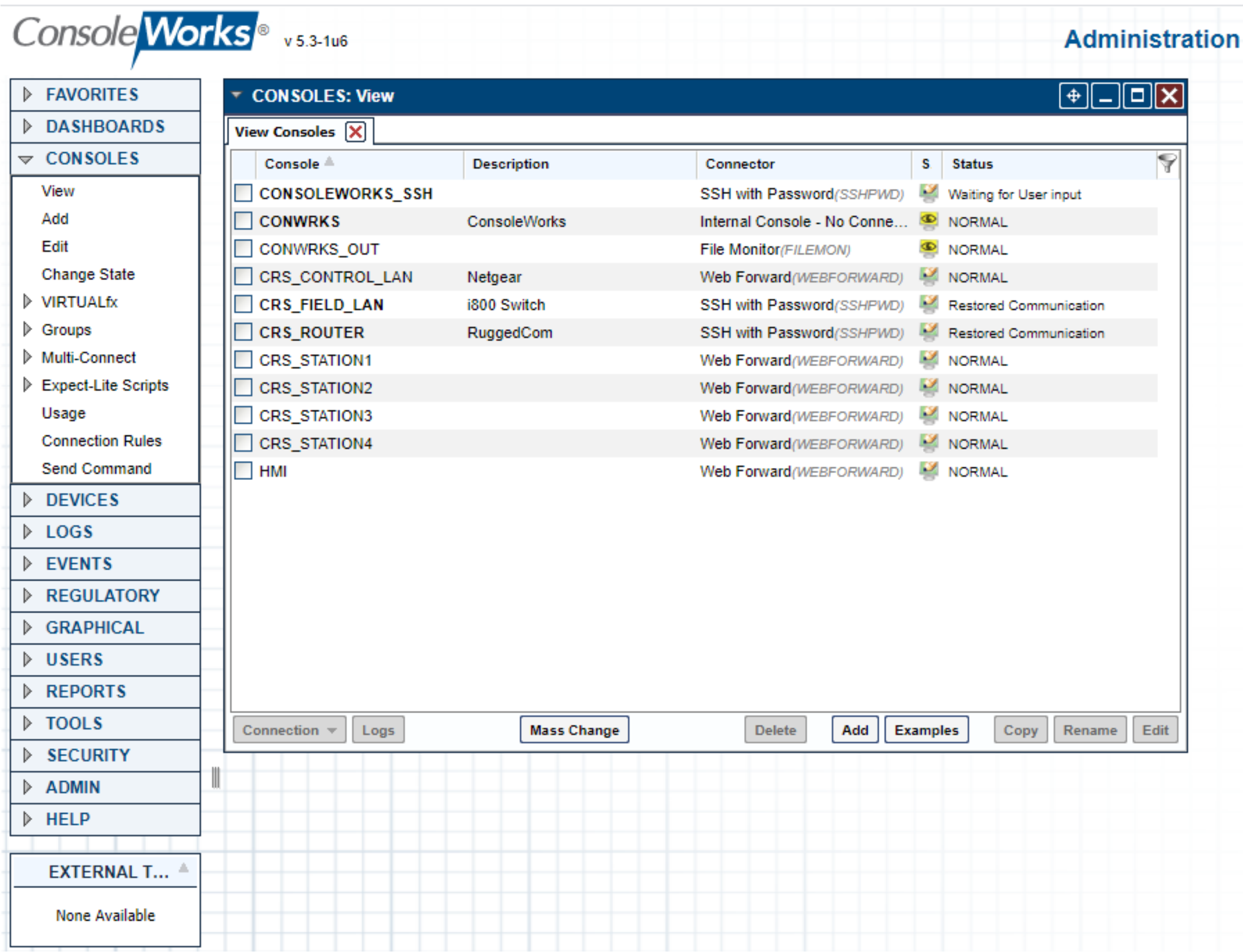
      1) Specify the username and password, which are securely stored by ConsoleWorks, to allow complex passwords/credentials without having to share between users.

Table 2-24 ConsoleWorks CRS (Build 3) Console Connections

Name	Device	Connector	Host	Port
CRS_CONTROL_LAN	CRS_SWITCH_CONTROL	Web Forward	192.168.0.239	80
CRS_FIELD_LAN	CRS_SWITCH_FIELD	SSH with Password	192.168.1.10	22

Name	Device	Connector	Host	Port
CRS_ROUTER	CRS_ROUTER	SSH with Password	192.168.0.2	22
CRS_STATION1	CRS_STATION1	Web Forward	192.168.1.101	80
CRS_STATION2	CRS_STATION2	Web Forward	192.168.1.102	80
CRS_STATION3	CRS_STATION3	Web Forward	192.168.1.103	80
CRS_STATION4	CRS_STATION4	Web Forward	192.168.1.104	80
HMI	CRS_HMI	Web Forward	192.168.0.98	80

Figure 2-60 ConsoleWorks List of CRS (Build 3) Console Connections



8. Configure tags to support profiles and access controls.

Figure 2-61 ConsoleWorks List of Tags for PCS (Build 1)

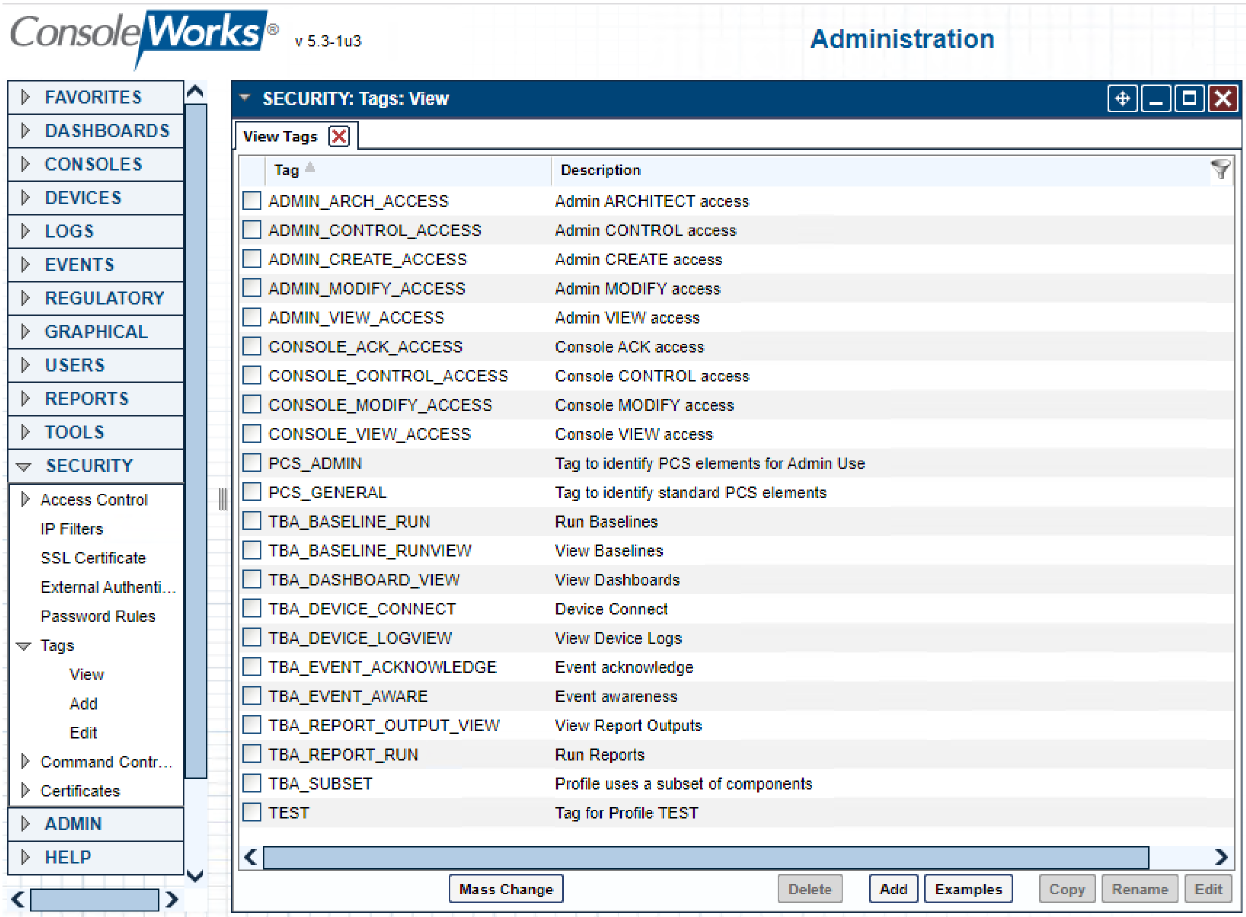
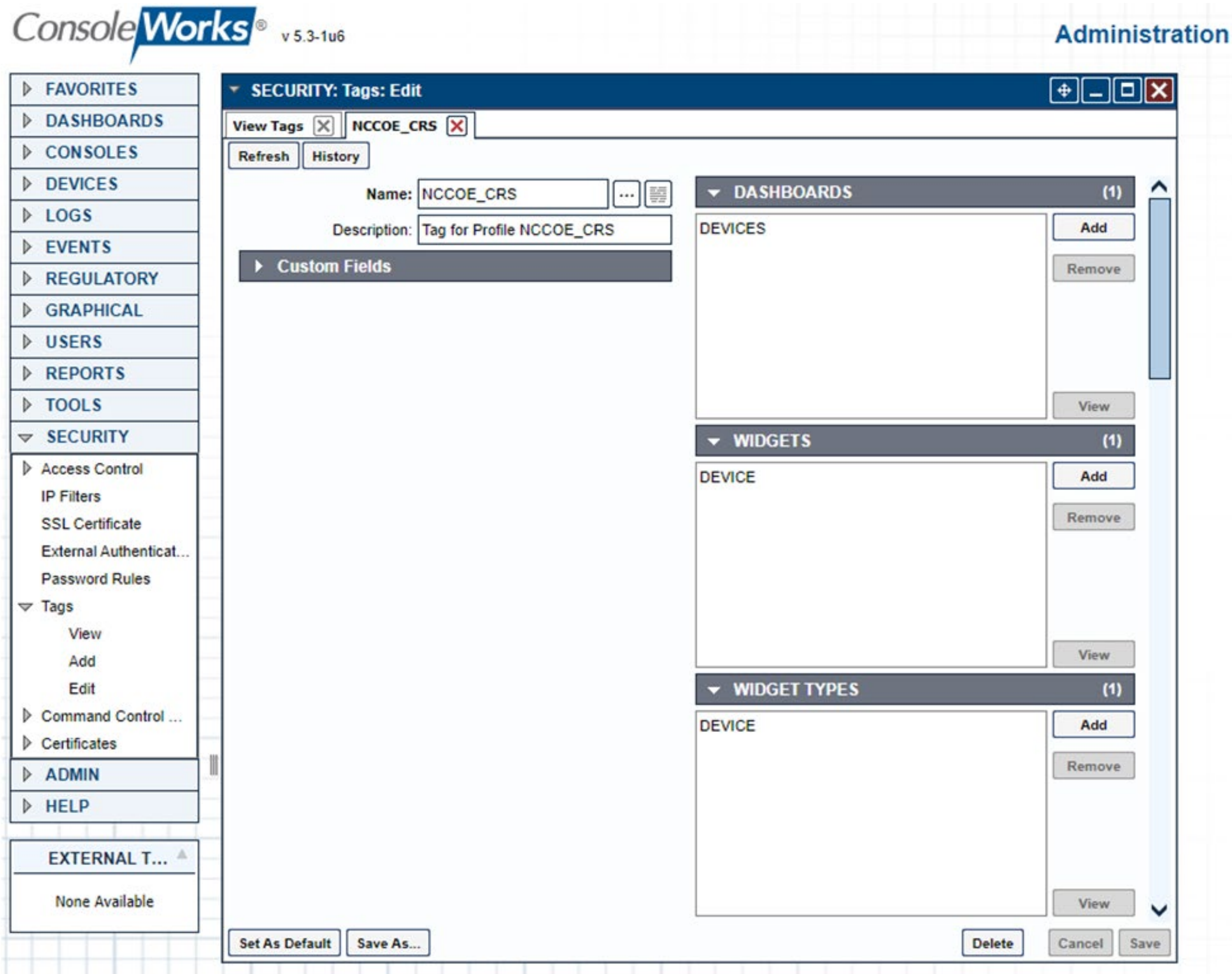


Figure 2-62 ConsoleWorks Example Tag Definition Screen



a. For Build 1 (PCS) the following tags were created as shown in Figure 2-61. Figure 2-62 shows an example of a single tag.

i. Name: **PCS_GENERAL**

1) Under **Dashboards**, click **Add** and select **Devices**.

2) Under **Custom UI Classes** click **Add** and select:

a) DEVICE_LISTGRID

b) LISTGRID

3) Under **Devices**, click **Add** and select:

a) DMZ_HISTORIAN

b) PCS_HISTORIAN

c) PCS_HMI

i. PCS_WORKSTATION

4) Under **Graphical Connections**, click **Add** and select:

a) DMZ_HISTORIAN

b) PCS_HISTORIAN

c) PCS_HMI_RDP

d) PCS_WORKSTATION_RDP

ii. Name: **PCS_ADMIN:**

1) Under **Dashboards** click **Add** and select **Devices**

2) Under **Custom UI Classes** click **Add** and select:

a) DEVICE_LISTGRID

b) LISTGRID

3) Under **Consoles**, click **Add** and select:

a) PCS_ROUTER

b) PCS_SWITCH_VLAN1

c) PCS_SWITCH_VLAN2

4) Under Devices, click Add and select:

a) PCS_ROUTER

b) PCS_SWITCH_VLAN1

c) PCS_SWITCH_VLAN2

b. For Build 3 (CRS) Create the following:

i. Name: **NCCOE_CRS**

1) Under **Dashboards**, click **Add** and select **Devices**.

2) Under **Custom UI Classes**, click **Add** and select:

a) DEVICE_LISTGRID

b) LISTGRID

3) Under Consoles, click Add and select:

a) CRS_STATION1

b) CRS_STATION2

c) CRS_STATION3

d) CRS_STATION4

e) HMI

4) Under **Devices**, click **Add** and select:

a) CRS_HMI

b) CRS_STATION1

c) CRS_STATION2

d) CRS_STATION3

e) CRS_STATION4

f) CRS_WORKSTATION

5) Under **Graphical Connections**, click **Add** and select:

a) CRS_WORKSTATION

ii. Name: **NCCOE_ADMIN**

1) Under Dashboards click Add and select Devices

2) Under Custom UI Classes click Add and select:

a) DEVICE_LISTGRID

b) LISTGRID

3) Under **Consoles** click **Add** and select:

a) CRS_CONTROL_LAN

b) CRS_FIELD_LAN

c) CRS_ROUTER

4) Under **Devices** click **Add** and select:

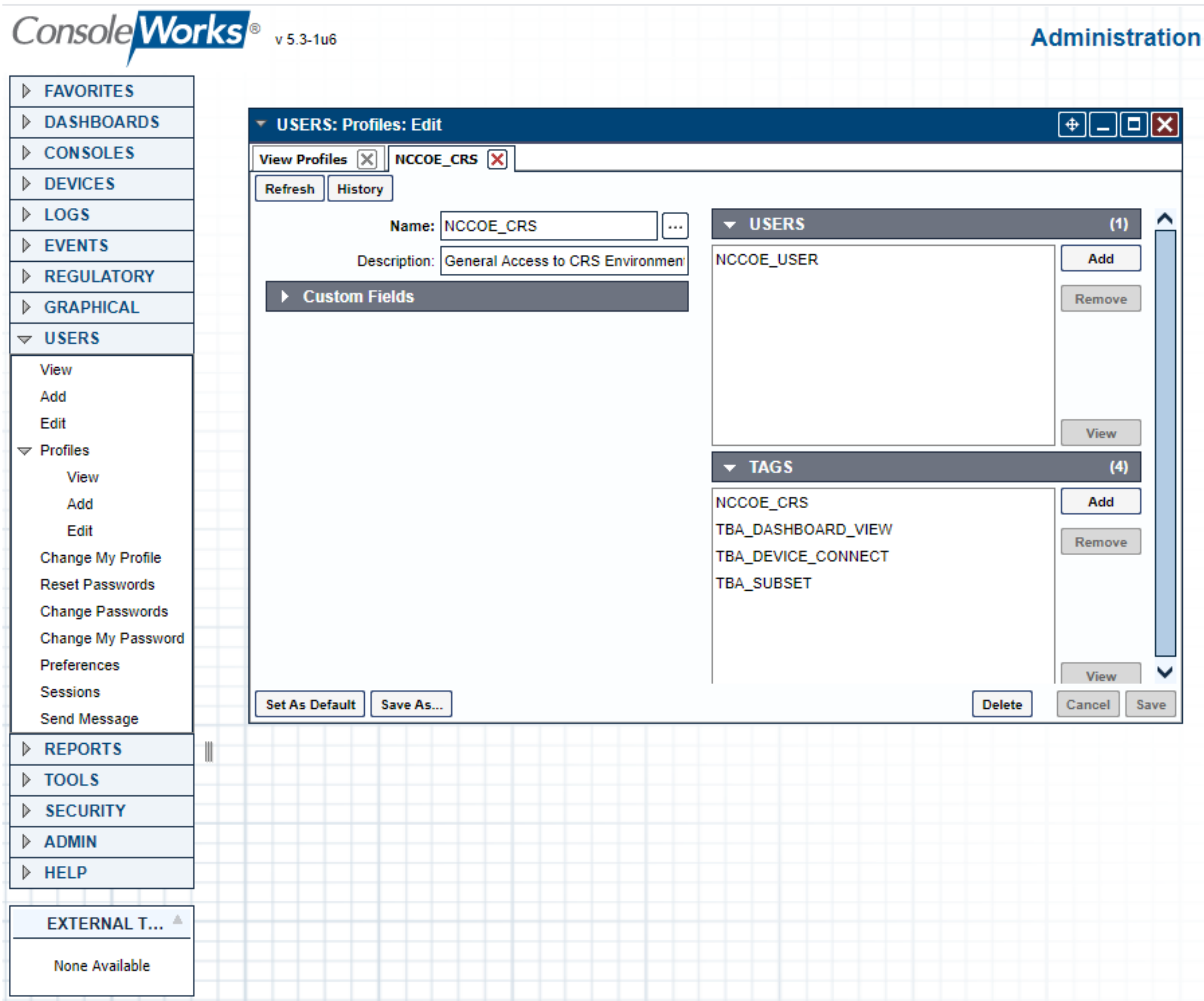
a) CRS_SWITCH_CONTROL

b) CRS_SWITCH_FIELD

c) CRS_ROUTER

9. Configure profiles to provide user accounts with granular access controls to available resources (Figure 2-63).

Figure 2-63 ConsoleWorks Example Profile



a. For Build 1 (PCS) the following profiles were created:

i. **PCS_GENERAL**

1) Under Users click Add and select

a) NCCOE_USER

2) Under Tags click Add and select

a) PCS_GENERAL

b) TBA_DASHBOARD_VIEW

c) TBA_DEVICE_CONNECT

d) TBA_SUBSET

ii. **PCS_ADMIN**

1) Under **Users**, click **Add** and select:

a) NCCOE_ADMIN

2) Under **Tags**, click **Add** and select:

a) PCS_ADMIN

b) TBA_DASHBOARD_VIEW

c) TBA_DEVICE_CONNECT

d) TBA_SUBSET

e) CONSOLE_CONTROL_ACCESS

f) CONSOLE_VIEW_ACCESS

b. For Build 3 (CRS) create the following:

i. **NCCOE_CRS** profile for the NCCOE_USER with access to Tags:

1) Under **Users**, click **Add** and select:

a) NCCOE_USER

2) Under **Tags** click **Add** and select the following:

a) NCCOE_CRS

b) TBA_DASHBOARD_VIEW

c) TBA_DEVICE_CONNECT

d) TBA_SUBSET

e) CONSOLE_CONTROL_ACCESS

f) CONSOLE_VIEW_ACCESS

ii. **NCCOE_ADMIN** profile for the NCCOE_USER with access to Tags:

1) Under Users, click Add and select:

a) NCCOE_ADMIN

2) Under Tags click Add and select the following:

a) NCCOE_ADMIN

b) TBA_DASHBOARD_VIEW

c) TBA_DEVICE_CONNECT

d) TBA_SUBSET

e) CONSOLE_CONTROL_ACCESS

f) CONSOLE_VIEW_ACCESS

2.9 Tenable.OT

The Tenable.OT implementation in Build 1 consists of a single appliance to meet BAD, hardware modification, firmware modification, and software modification capabilities. Tenable.OT utilizes a combination of passive and active sensors to monitor critical networks for anomalies and active querying to retrieve information about endpoints in the PCS environment.

2.9.1 Host and Network Configuration

Tenable.OT is installed and configured to support the PCS environment in Build 1. The overall build architecture is described in Figure B-1, and the Tenable.OT specific components are listed in Table 2-25.

Table 2-25 Tenable.OT Appliance Details.

Name	System	OS	CPU	Memory	Storage	Network
Tenable.OT	Model: NCA-4010C-IG1	CentOS 7	Intel Xeon D-1577	64 GB	64 Gb 2 TB 2 TB	Testbed LAN 10.100.0.66

2.9.2 Installation

The Tenable.OT (Version 3.8.17) appliance is installed in a rack with network connections for the Management/Query traffic on Port 1 and SPAN traffic on Port 2 of the appliance. Documentation for Tenable.OT is available at https://docs.tenable.com/Tenableot.htm.

2.9.3 Configuration

This section outlines the steps taken to configure Tenable.OT to fully integrate and support the PCS environment. These include setting NTP settings to synchronize the system time with the lab time source, configuring the scanning options for the PCS environment, and configuring network objects and policies to enhance alerting for DMZ specific remote connections.

1. Enable connection through PCS Firewall

    a. Add the following rules (Table 2-26) to the PCS Firewall to allow Tenable.OT to perform asset discovery and controller scanning.

Table 2-26 Firewall Rules for Tenable.OT

Rule Type	Source	Destination	Protocol:Port(s)	Purpose
Allow	10.100.0.66	172.16.0.0/22	ICMP	Asset Discovery
Allow	10.100.0.66	172.16.2.102	TCP:44818,2222	PLC Controller Scans

2. Set NTP Services as follows:

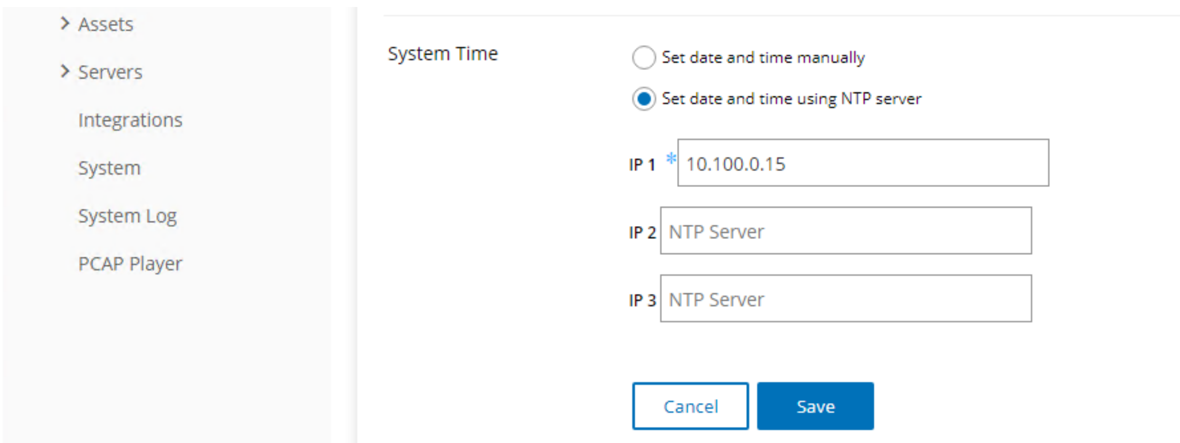
a. After logging into the appliance, navigate to **Local Settings > Device**.

b. To the right of **System Time**, click **Edit** to display the time service options (Figure 2-64).

c. Enter the NTP Server information: **10.100.0.15**

d. Click **Save**.

Figure 2-64 Tenable.OT Local Device Setting for NTP Service



3. Configure Scanning Options as follows:

   a. Set Asset Discovery Scans:

      i. Navigate to **Local Settings > Queries > Asset Discovery** (Figure 2-65)

      ii. Enable both scan options.

      iii. Select **Edit** next to **Asset Discovery**.

         1) Enter the following CIDR for the PCS, DMZ, and Testbed networks:

            a) **172.16.0.0/22**

            b) **10.100.0.0/24**

            c) **10.100.1.0/24**

         2) Set the scan properties as follows:

            a) Number of Assets to Poll Simultaneously: **10**

            b) Time Between Discovery Queries: **1 second**

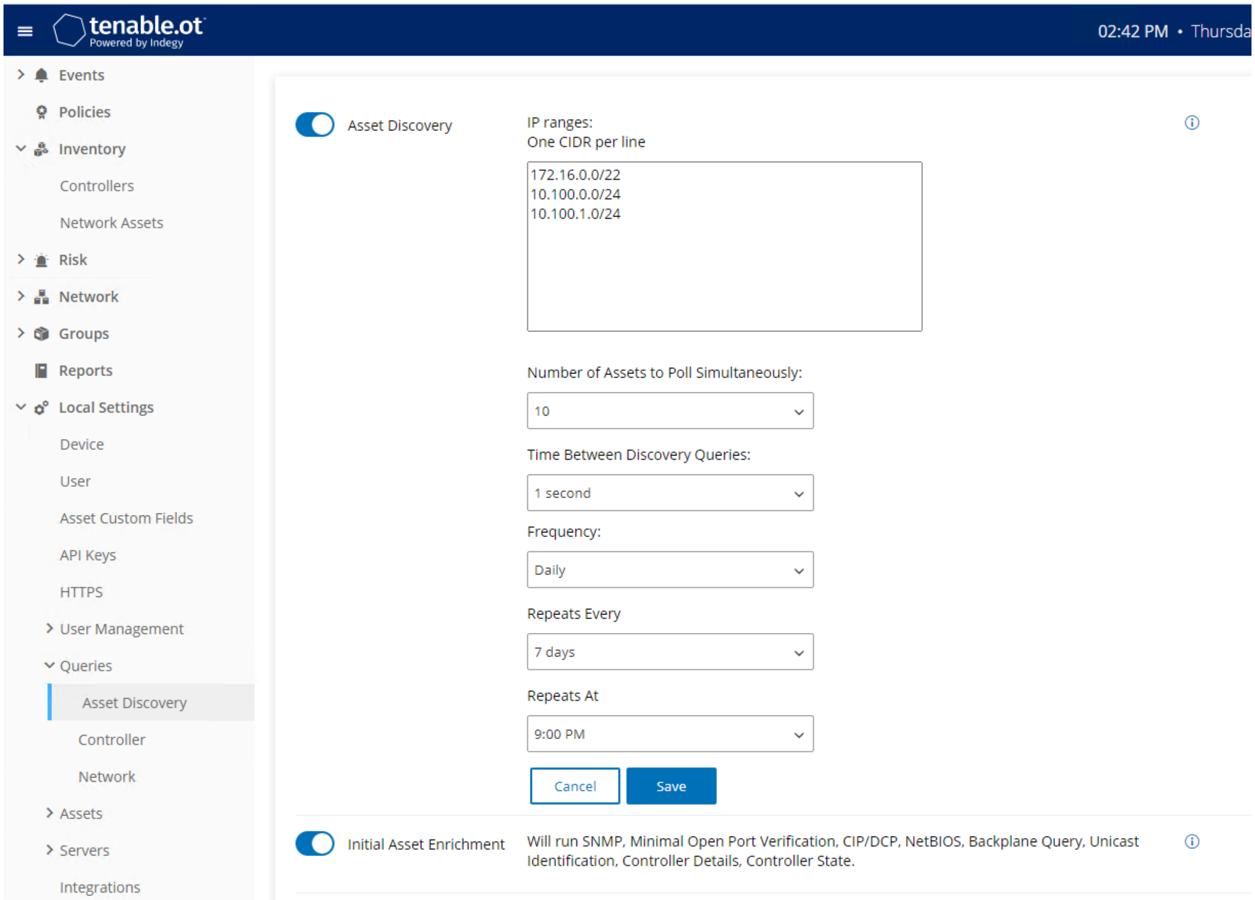
            c) Frequency: **Daily**

            d) Repeats Every: **7 Days**

            e) Repeats at: **9:00 PM**

         3) Click **Save**.

Figure 2-65 Tenable.OT Asset Discovery Settings



b. Set Controller Scans as follows:

   i. Navigate to **Local Settings > Queries > Controller** (Figure 2-66)

   ii. Enable the following options:

      1) All Controller Queries

      2) Periodic Snapshots

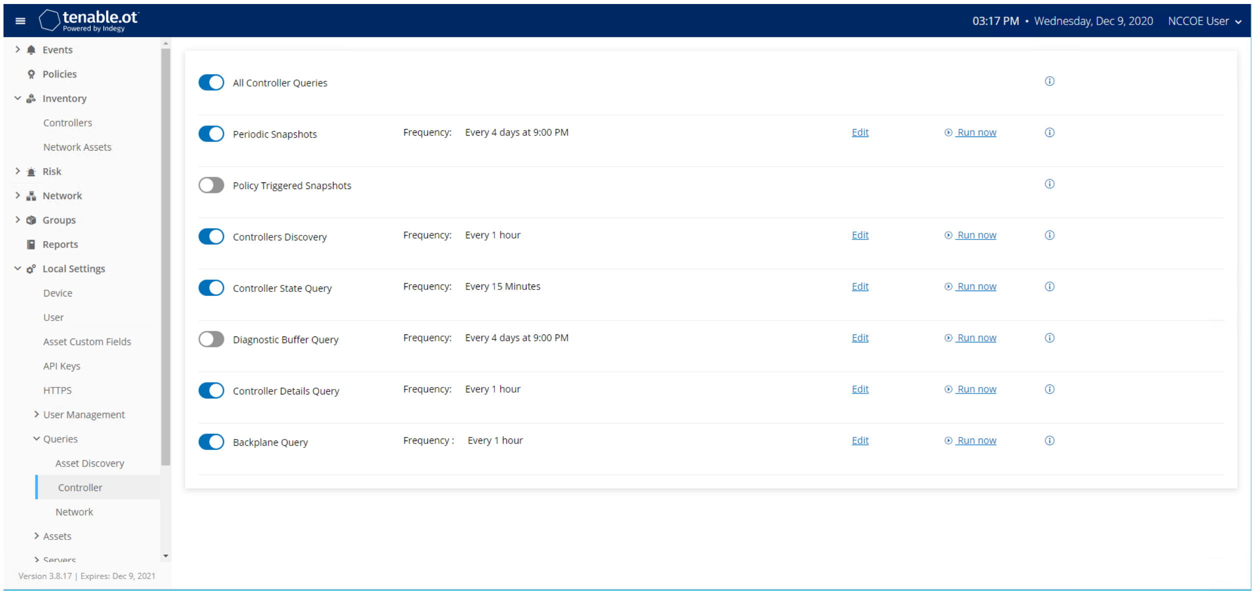
      3) Controller Discovery

      4) Controller Status Query

      5) Controller Details Query

      6) Backplane Query

Figure 2-66 Tenable.OT Controller Scans



c. Set Network Scans as follows:

i. Navigate to **Local Settings > Queries > Network** (Figure 2-67)

ii. Enable the following options:

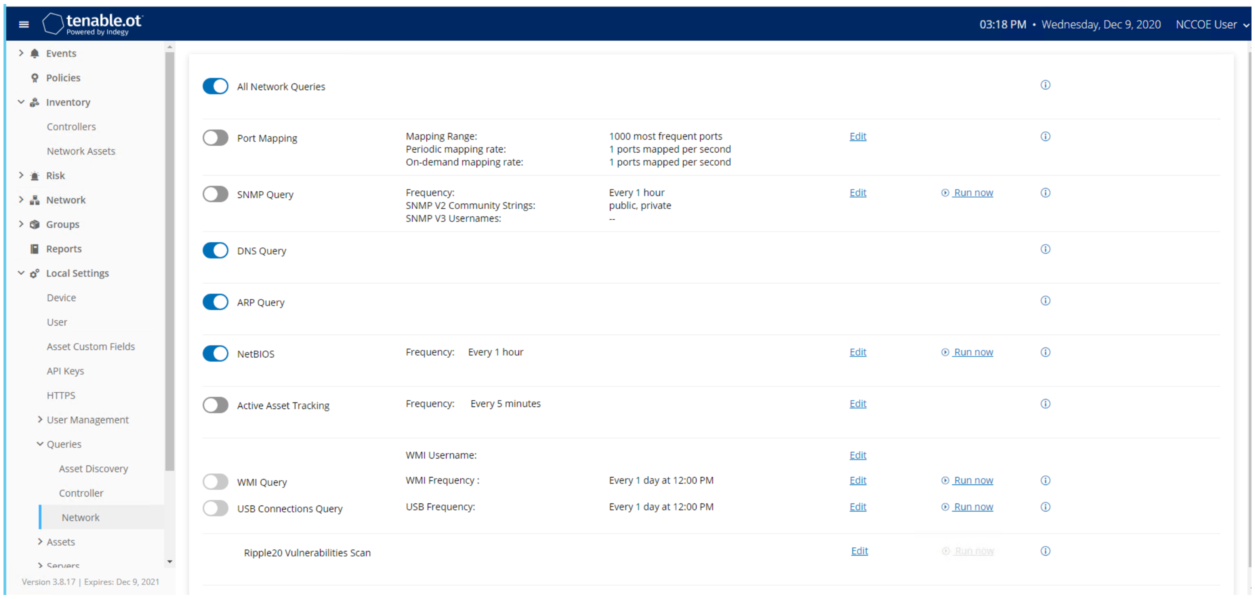
1) All Network Queries

2) DNS Query

3) ARP Query

4) NetBIOS Query

Figure 2-67 Tenable.OT Network Scan Settings



4. Create Group Object as follows:

    a. Set DMZ Group Object

        i. Navigate to **Groups > Asset Groups**

        ii. Click **Create Asset Group** to initiate the Wizard process.

            1) Select **IP Range** for the Asset Group Type (Figure 2-68) and Click **Next**.

            2) Enter the asset name in **Name**, the starting IP address in **Start IP**, and the ending IP Address in **End IP** (Figure 2-69) and Click **Create**.

Figure 2-68 Tenable.OT Create Asset Group Type

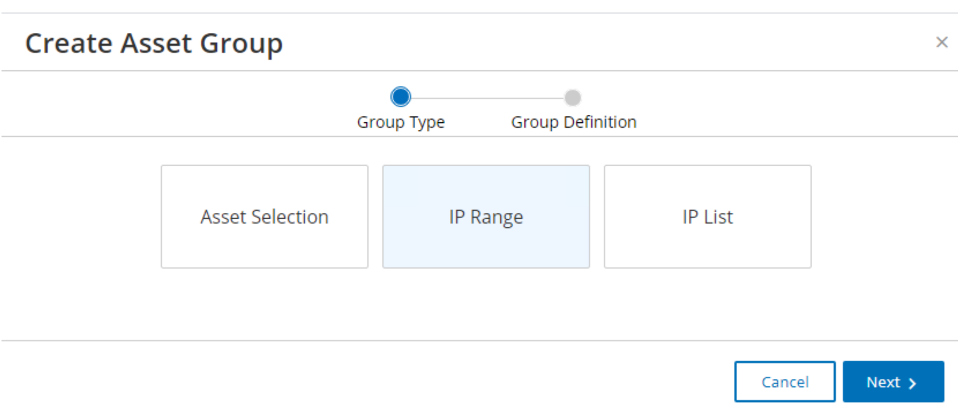
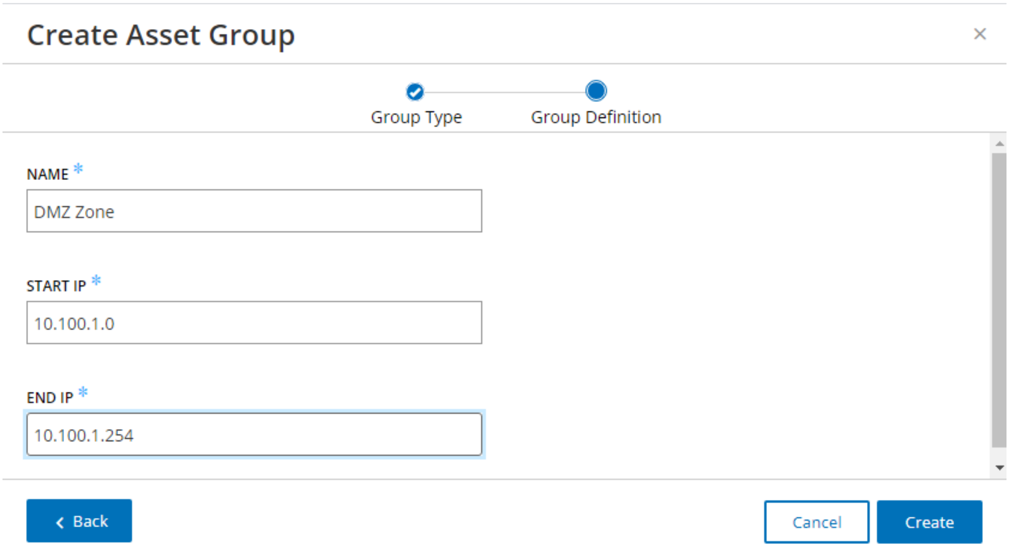


Figure 2-69 Tenable.OT Create Asset Group Definition



5. Create Policy to Detect External RDP Traffic:

   a. In the left side navigation, click **Policies**.

   b. Click **Create Policy** in the upper right corner of the page (Figure 2-70), then follow these steps:

      i. For the Event Type (Figure 2-71), select as **a Network Events > RDP Connection (Authenticated)** and click **Next**.

      ii. For the Policy Definition (Figure 2-72), specify the following parameters and click **Next**:

         1) **Policy Name**: Enter "External RDP Communications"

         2) **Source Group**: Select "In" from the first drop-down, and "DMZ" from the second drop-down.

         3) **Destination Group**: Select "In" from the first drop-down and select "In Any Asset" from the second drop-down.

         4) **Schedule Group**: Select "In" from the first drop-down, and "In Any Time" from the second drop-down.

      iii. For the Policy Action (Figure 2-73), select **Medium** Sensitivity and click **Create**.

Figure 2-70 Tenable.OT Policy Settings

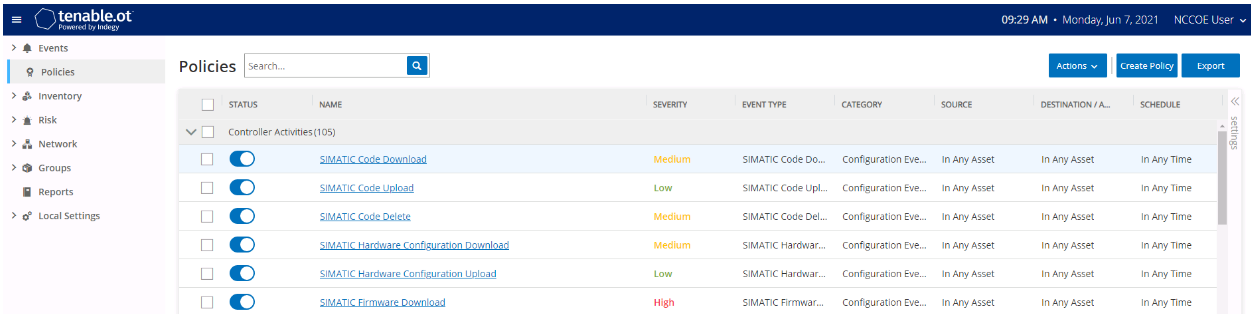


Figure 2-71 Tenable.OT Create Policy – Event Type Options

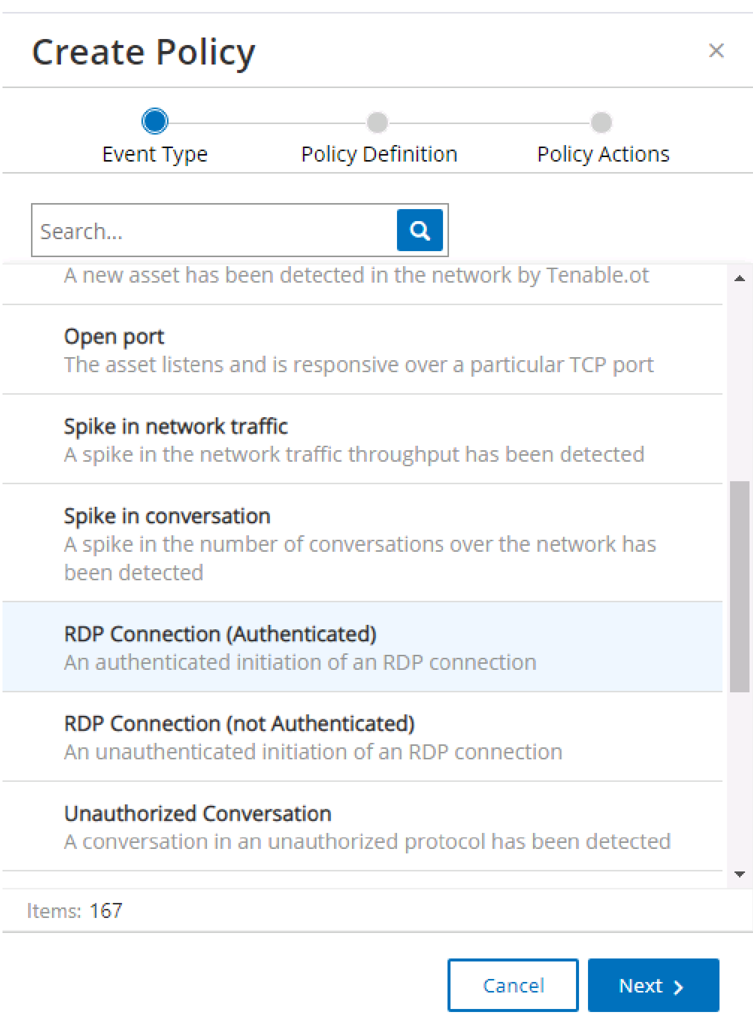


Figure 2-72 Tenable.OT Create Policy - Definition

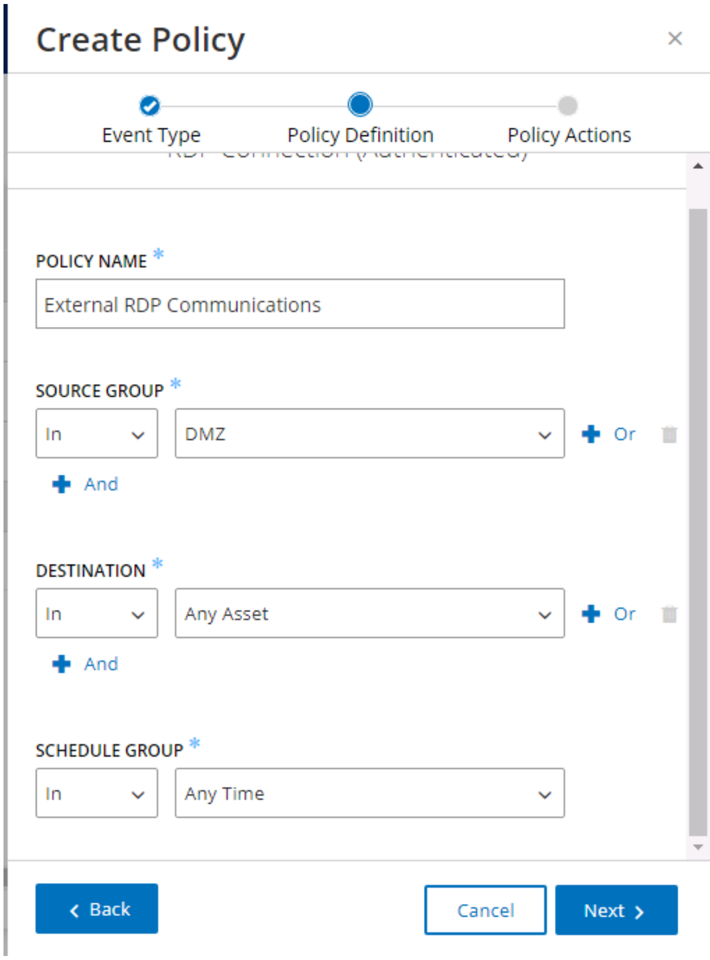
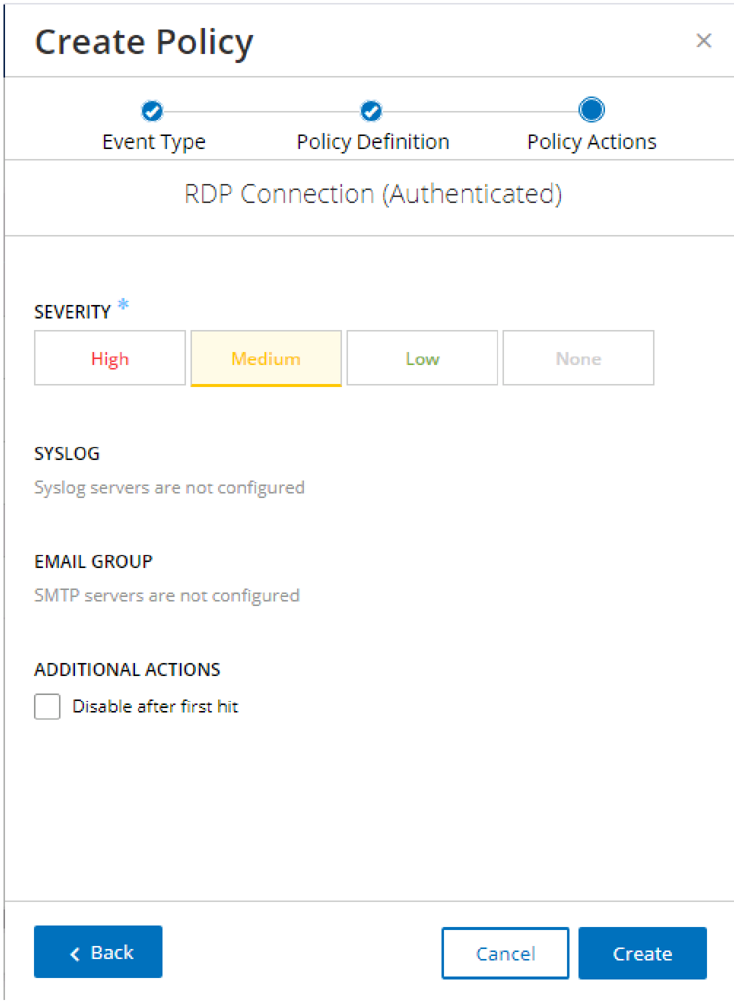


Figure 2-73 Tenable.OT Create Policy - Actions



2.10 VMware Carbon Black App Control

VMWare Carbon Black App Control is an endpoint protection tool that provides multiple file integrity and application features, including application allow/deny listing and file modification or deletion protection. Carbon Black was used for Builds 1 and 4 as the application allowlisting (AAL) and file integrity checking tool.

2.10.1 Host and Network Configuration

The following tables (Table 2-27, Table 2-28, and Table 2-29) detail the host and network configuration of the Carbon Black App Control server for PCS and CRS.

Table 2-27 Carbon Black App Control Domain Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
Carbon Black Server	VMware ESXi VM	Windows Server 2016 Datacenter	4	8GB	500GB	Testbed LAN 10.100.0.52
Windows Server	Hyper-V VM	Windows Server 2012 R2	2	6GB	65GB	Testbed LAN 10.100.0.25
OSIsoft Pi Server	Hyper-V VM	Windows Server 2016 Standard	4	8GB	80GB/171GB	DMZ 10.100.1.4
Dispel VDI	Hyper-V VM	Windows Server 2016 Datacenter	2	8GB	126GB	N/A

Table 2-28 Carbon Black App Control PCS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
PCS HMI Workstation	Supermicro Z97X-Ud5H	Windows 7	4	8GB	233GB	PCS 172.16.1.4
PCS Engineering Workstation	Supermicro Z97X-Ud5H	Windows 7	4	16GB	465GB	PCS 172.16.3.10

Table 2-29 Carbon Black App Control CRS Hosts Deployment

Name	System	OS	CPU	Memory	Storage	Network
CRS Engineering Workstation	Dell Precision T5610	Windows 10	8	16GB	465GB	CRS Supervisory 192.168.0.20
CRS OSIsoft Pi Server	Hyper-V VM	Windows Server 2016 Standard	4	16GB	80GB/171GB	CRS Supervisory 192.168.0.21

2.10.2 Installation

Prepare the Carbon Black App Control Server (fka CB_Protection) in accordance with the CB Protection Operating Environment Requirements v8.1.6 document that is provided for installation. This document, and all Carbon Black documentation, can be found on the website https://community.carbonblack.com.

1. Install Carbon Black App Control Server (fka CB_Protection) using these steps:

a. Created the nccoeCarbon domain user account on LAN AD to be used for installation and administration of CB App Control Server and add this user to the local administrators' group on the server.

b. Install SQL Server Express 2017 according to the CB Protection SQL Server Configuration v8.1.4 document.

c. Install the CB App Control Server according to the CB Protection Server Install Guide v8.1.6 document.

2.10.3 Configuration

Follow these steps to configure Windows Server 2016:

1. On the Carbon Black App Control Server, configure Windows Server 2016:

   a. Based on Carbon Black documentation (Figure 2-74), Windows Server 2016 will need to have the following features for the Internet Information Services (IIS) role enabled for Carbon Black to work (Figure 2-75).

Figure 2-74 Excerpt from Carbon Black Documentation on Support Server Requirements

Carbon Black.

CB Protection Web Server Platform: Support Server

Common Requirements ①

In the IIS Roles Manager, verify the following configuration:

- Common HTTP Features:
  - Static Content
  - Default Document
  - HTTP Errors
  - HTTP Redirection
- Application development:
  - ASP.NET (version 4.5)
  - .NET Extensibility (version 4.5)
  - CGI
  - ISAPI Extensions
  - ISAPI Filters
- Health & Diagnostics:
  - HTTP Logging
  - Logging Tools
  - Request Monitor
  - Tracing
- Security:
  - URL Authorization
  - Request Filtering
  - IP and Domain Restrictions
- Performance: None
- Management Tools:
  - IIS Management Console
  - IIS Management Scripts and Tools
  - Management Service
- FTP Publishing Service: None

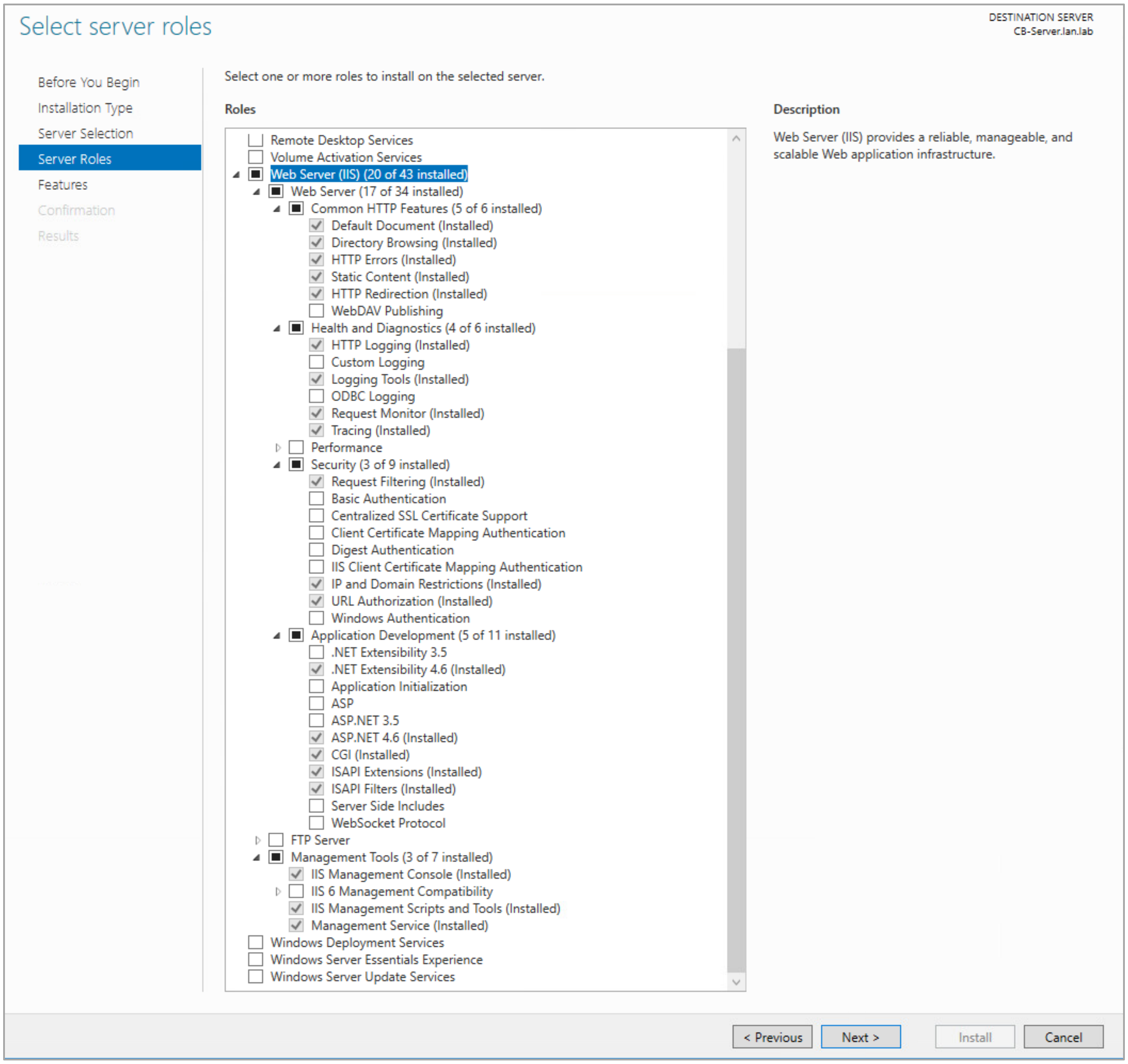
Restrictions ②

Beginning with v8.0.0, the console relies on the CB Protection API. An incorrectly configured IIS server can prevent console access.

- To confirm API functionality, go to **System Configuration > Advanced Options** in your current console and check the "API Access Enabled" box. If a green dot appears next to the checkbox, then you can assume that IIS is configured correctly. Otherwise, make sure you meet the following restrictions:
- Site Bindings:
  The CB Protection API will not connect to localhost if the console web application is bound to a specific IP address instead of "*". Make sure that "*" is added to the list of bindings.
- IP Address and Domain Restrictions:
  If you must limit console access to specific IP addresses, be sure that the IPv6 localhost address is added to the list.
- Application Pools:
  CB Protection must be run within the DefaultAppPool application pool. Using a different app pool results in the CB Protection server not having the appropriate credentials to access the SQL Server database.
- Authentication:
  You must disable Basic Authentication and Windows Authentication so that the CB Protection Server handles authentication. Otherwise, users will not be able to log into the CB Protection Server.

Version	Part Of OS	Current Version	Supported Architecture	Supported Level	Additional Notes/Requirements
IIS 8.5	Windows 2012 Server R2 only		x64		① ② Common Requirements and Restrictions are listed in the table above Additional requirements: Private memory for IIS should be increased to 800 MB
IIS 10	Windows 2016 Server		X64		① ② Common Requirements and Restrictions are listed in the table above Additional requirements: Private memory for IIS should be increased to 800 MB

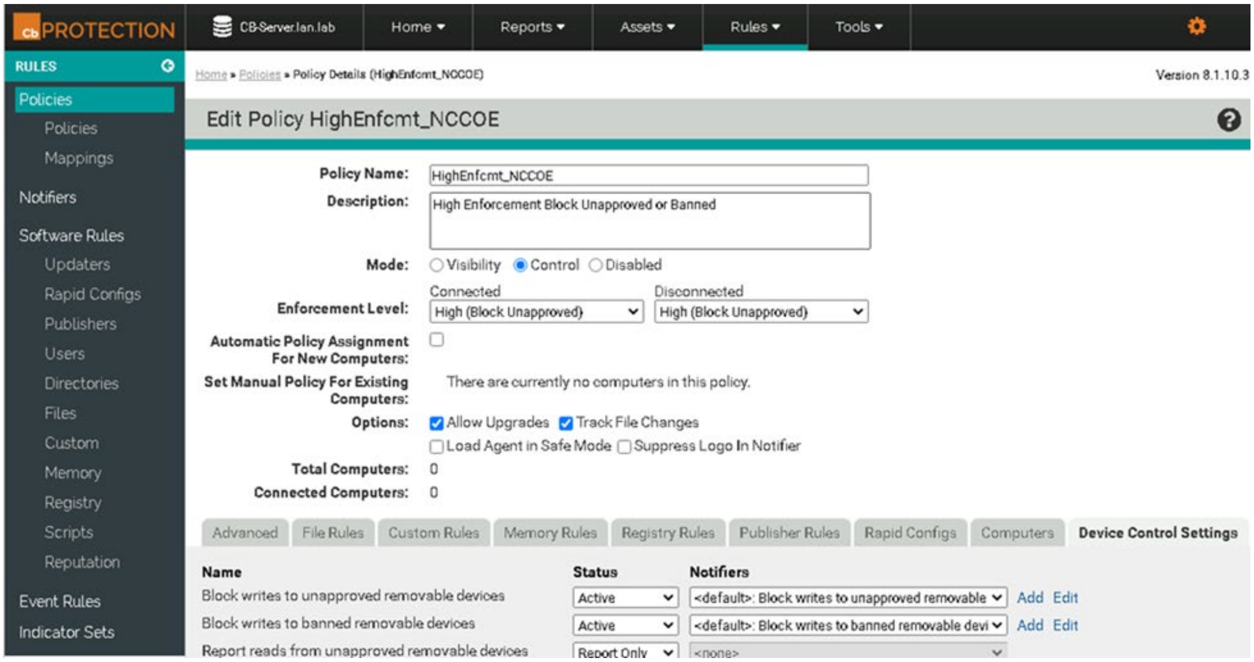
Figure 2-75 IIS Configuration for Carbon Black, Server Roles



2. Manually update the Windows Server firewall configuration to allow inbound port 41002 traffic from CB App Control clients/agents.
3. Configure Policy in the Carbon Black Console using these steps:
    a. In the CB App Control Console, go to **Rules > Policies.**
    b. Create a new policy with the desired enforcement level. In this case, a high enforcement level was chosen to actively block execution of unapproved or banned executables (Figure 2-76).

This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

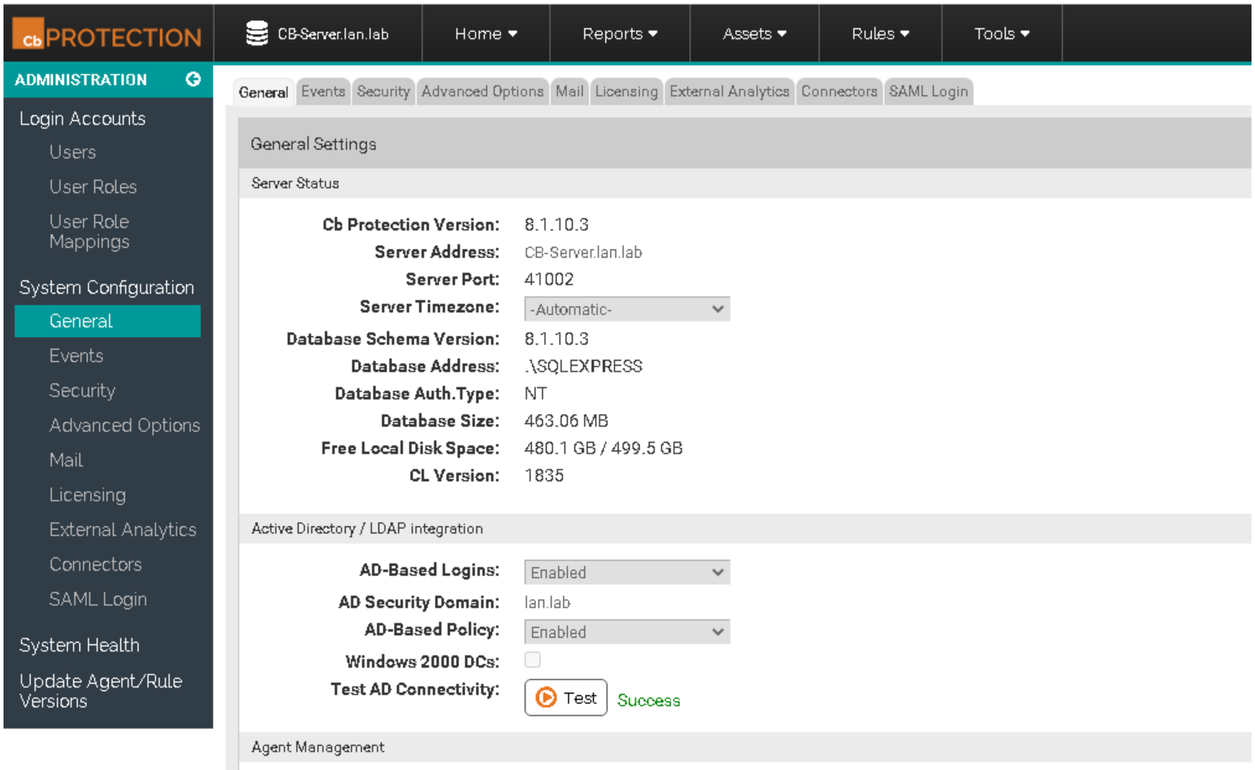
Figure 2-76 Carbon Black Policy Edit



4. Enable AD Integration Features as follows:

   a. Enable AD integration features on the CB App Control Console for domain user account login and AD-Based Policy mapping. AD-Based Policy mapping allows automatic policy assignment to be mapped to AD users, groups, computers, organizational units (OUs), etc., as configured by a CB App Control Console administrator (Figure 2-77).

Figure 2-77 Carbon Black App Control System Configuration

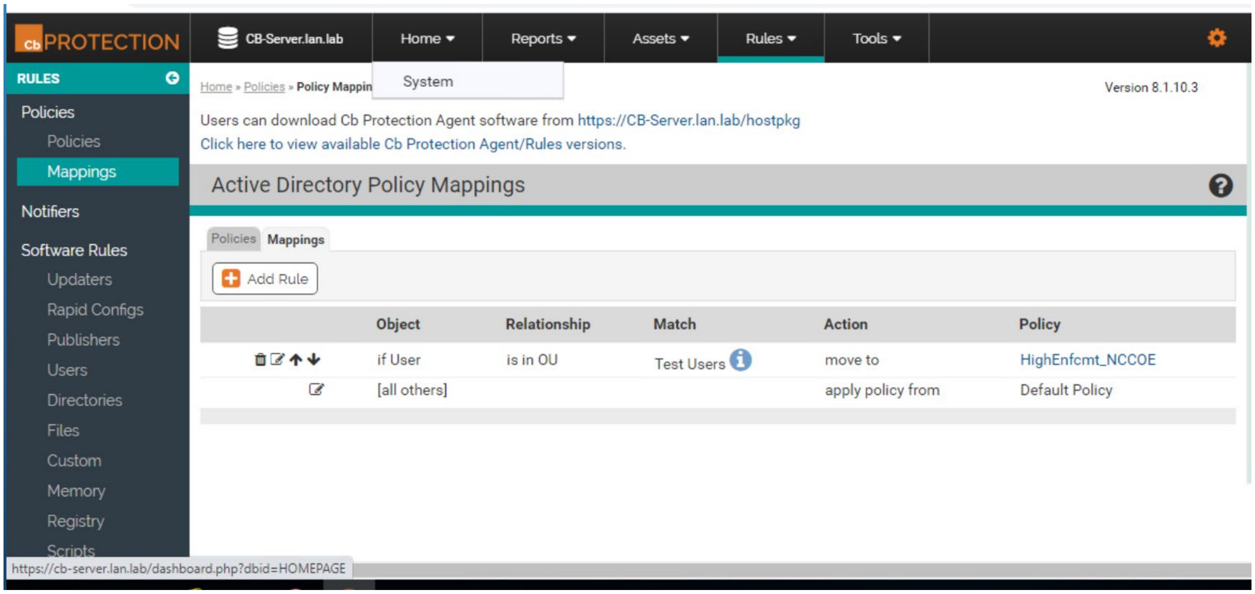


5. Add users from AD and assign policies:

   a. Add "Test Users" OU from the AD to policy mapping settings and assign the "High-Enfcmt_NCCOE" policy (Figure 2-78).

      This OU includes the "nccoeUser" and "nccoeAdmin" user accounts created for the test scenarios. This policy will be automatically applied to these users logged in on any computer that is running the CB Protection Agent. The "HighEnfcmt_NCCOE" policy is set to High Enforcement level, which will actively block all unapproved or banned files, applications, or devices.

Figure 2-78 Carbon Black App Control AD Policy Mappings



6. Download and install CB App Control Agent from CB App Control Server

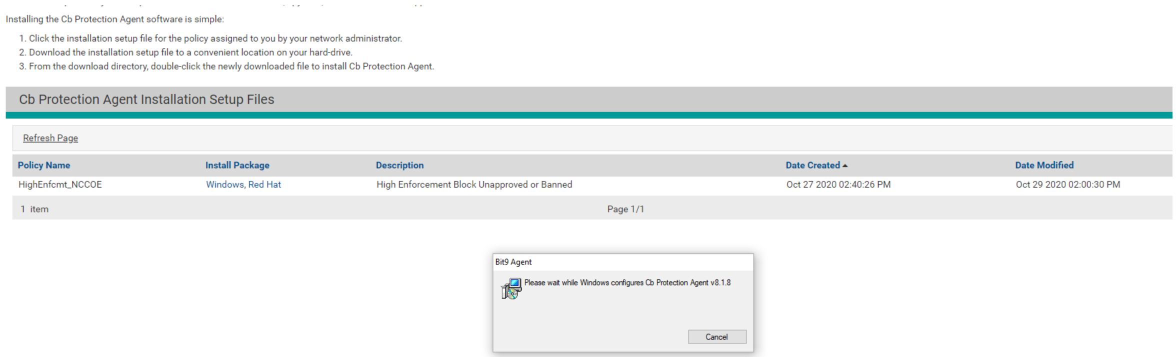
(The process outlined below uses the CRS Engineering Workstation as an example, but the process was the same for all the agent computers.). Follow these steps:

a. Open the browser on the CRS Engineering Workstation and enter the URL to download the agent installer: https://CB-Server.lan.lab/hostpkg. This URL is on the Carbon Black server itself and is accessed on the local network. CB-Server.lan.lab is the full host name we gave this server during installation.

   i. If the host cannot access CB-Server.lan.lab, update the environment DNS Server by mapping the IP address, 10.100.0.52, to CB-Server.lan.lab or add the mapping to the local host file.

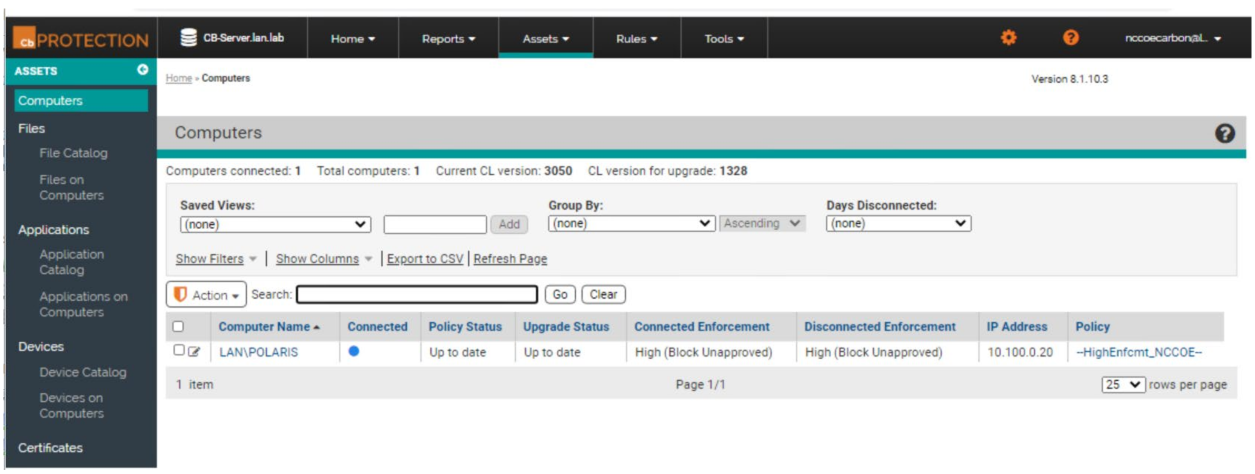
b. Download the Windows CB App Control Agent installer from the CB App Control Server and install on the CRS Engineering Workstation (Figure 2-79).

Figure 2-79 Carbon Black Agent Download



c. Check the CB App Control Console to verify communication and initialization of the new CRS Engineering Workstation agent computer on the CB App Control Server (Figure 2-80).

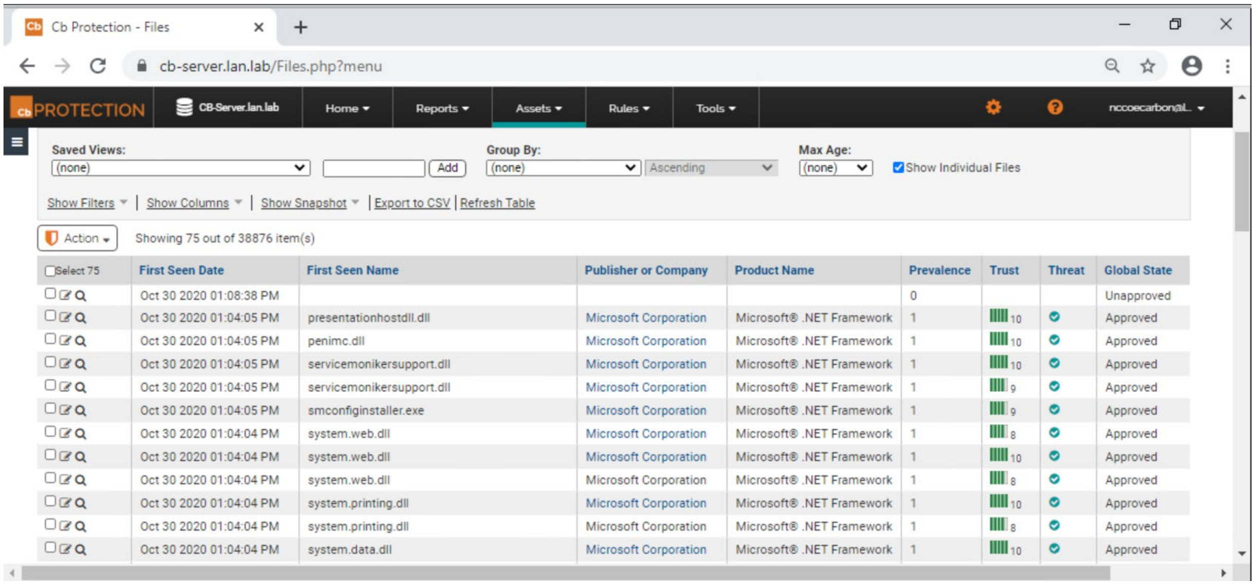
Figure 2-80 Carbon Black App Control Computers



d. Approve all new trusted files and publishers that were added from the CRS Engineering Workstation to the catalog on the CB App Control Server.

e. This image (Figure 2-81) shows the **CB Protection - Files** page of the CB App Control Console.

Figure 2-81 Carbon Black App Control File Catalog



2.11 Windows Software Restriction Policy (SRP)

Windows SRP is a feature that is a part of the Windows operating system. It identifies applications that are running on any domain-controlled computer, and it can block any programs that have not been allow-listed. Configuring Windows SRP is done through group policy object management. Windows SRP was used for AAL in Builds 2 and 3.

2.11.1 Host and Network Configuration

Windows SRP configuration is established by Group Policy Objects (GPOs) located on the two AD servers. The domain controllers were common across all builds as detailed in Table 2-30.

Table 2-30 Windows SRP Domain Servers

Name	System	OS	CPU	Memory	Storage	Network
AD (Primary) Server	Hyper-V VM	Windows 2012R2	2x vCPU	2 GB	45 GB	Testbed LAN 10.100.0.17
AD (Secondary) Server	Hyper-V VM	Windows 2012R2	1x vCPU	2 GB	21 GB	Testbed LAN 10.100.0.13

The following systems were configured to utilize Windows SRP for each build. Additional details for each build are available in Section 4.5 of Volume B.

Build 2 supports the testing within the PCS environment. The overall build architecture is provided in Figure B-2. The Windows SRP specific components are in Table 2-31.

Table 2-31 Windows SRP Build 2 Deployment

Name	System	OS	CPU	Memory	Storage	Network
Windows Server	Hyper-V VM	Windows 2012R2	2x vCPU	6 GB	65 GB	Testbed LAN 10.100.0.25
Dispel VDI	Hyper-V VM	Windows 2016	2x vCPU	8 GB	126 GB	DMZ LAN 10.100.1.61
DMZ Historian	Hyper-V VM	Windows 2016	4x vCPU	8 GB	80 GB, 171 GB	DMZ LAN 10.100.1.4
Engineering Workstation	HP Z230 Workstation	Windows 7	Intel i5-4570	16 GB	465 GB	172.16.3.10
HMI Host	Generic	Windows 7	Intel i5-4590	8 GB	233 GB	PCS VLAN 1 172.16.1.4

Build 3 supports the testing within the CRS environment. The overall build architecture is provided in Figure B-3. The Windows SRP specific components are in Table 2-32.

Table 2-32 Windows SRP Build 3 Deployment

Name	System	OS	CPU	Memory	Storage	Network
Windows Server	Hyper-V VM	Windows 2012R2	2x vCPU	6 GB	65 GB	Testbed LAN 10.100.0.25
DMZ Historian	Hyper-V VM	Windows 2016	4x vCPU	8 GB	80 GB, 171 GB	DMZ LAN 10.100.1.4
Engineering Workstation	Dell T5610	Windows 10	2x Intel E3-2609 v2	16 GB	465 GB	CRS Supervisory LAN 192.168.0.20
CRS Local Historian	Hyper-V VM	Windows 2016	4x vCPU	16 GB	80 GB, 171 GB	CRS Supervisory LAN 192.168.0.21

2.11.2 Installation

Windows SRP is a feature of the Windows operating system and therefore did not require any specific installation for use in the project.

2.11.3 Configuration

The Windows SRP configuration required setting GPOs on the AD servers to enable the policy on all hosts that were part of the Windows domain. Additionally, hosts that were not part of the Windows domain had GPO settings configured locally to the host. Follow these steps to configure AD with user accounts and set enforcement policies:

1. Set up AD with a "Test User" OU and add the NCCOE User (nccoeUser) and Admin (nccoeAdmin) accounts for this project to the OU.

2. To allow the NCCOE Admin account to be included as a local administrator within the environment, modify the Default Domain GPO to add administrators to the Restricted Group and include the NCCOE Admin account.

3. To support applying GPOs as local settings to non-domain computers, download LGPO.zip from Microsoft Security Compliance Toolkit 1.0 available at https://www.microsoft.com/en-us/download/details.aspx?id=55319.

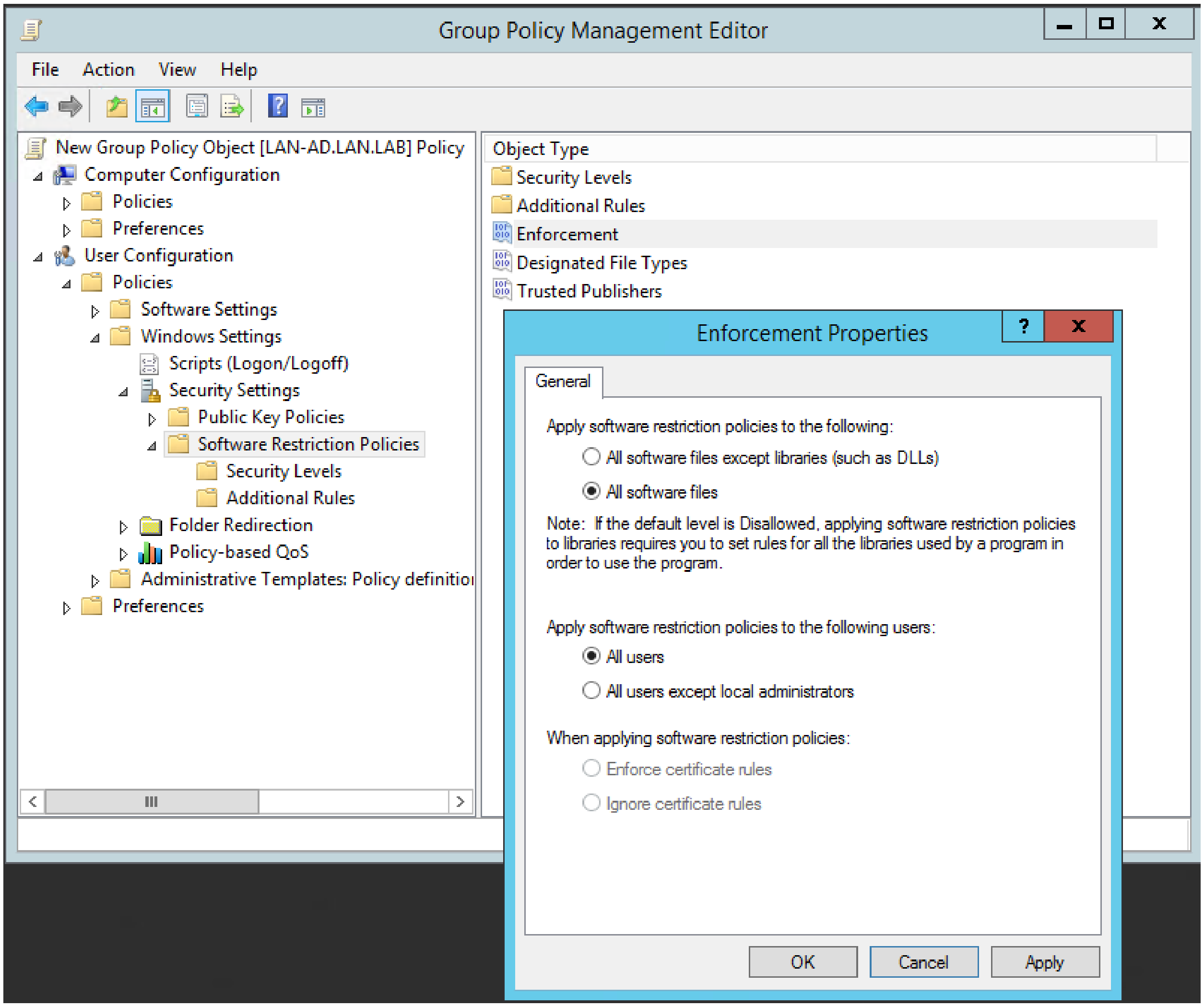
4. Review the National Security Agency (NSA) Guidance for Application Whitelisting using Software Restriction Policies and Guidelines for Application Whitelisting ICSs available at https://www.iad.gov/iad/library/reports/application-whitelisting-using-srp.cfm and https://www.iad.gov/iad/library/ia-guidance/security-configuration/industrial-control-systems/guidelines-for-application-whitelisting-industrial-control-systems.cfm respectively.
5. Create the Windows SRP GPO with the following settings:

    a. From the **Enforcement Properties** dialog (Figure 2-82):

        i. Select the **All Software Files** radio button.

        ii. Select the **All Users** radio button.

Figure 2-82 Setting Enforcement Properties

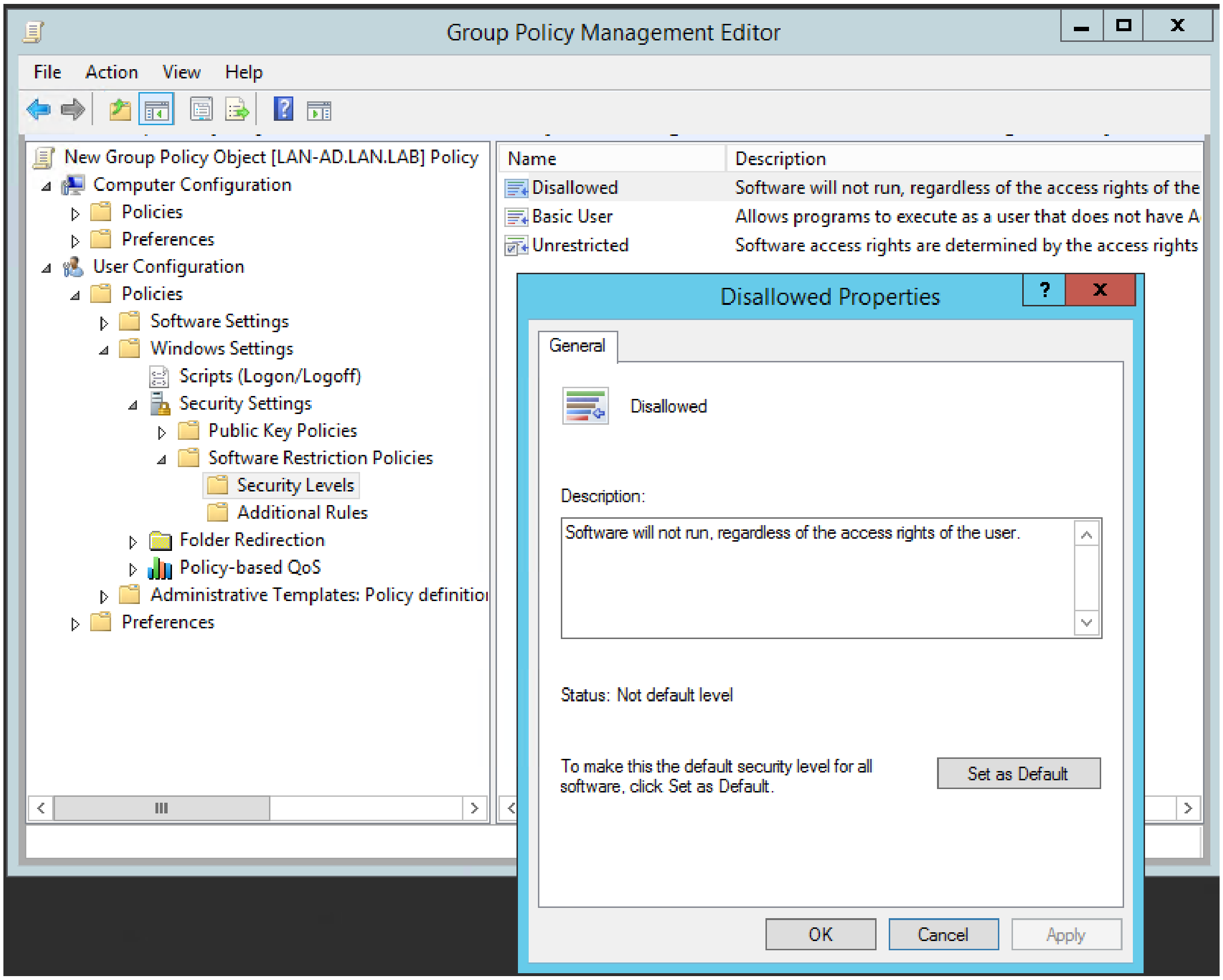


b. In the **Group Policy Management Editor**, in the **Security Levels** folder:

   i. Double-click the **Disallowed** security level to open the **Disallowed Properties** window.

   ii. Click the **Set as Default** radio button (Figure 2-83) to configure SRP in allowlist mode. After completing this step, only programs in the paths specified by the environment variables SYSTEMROOT (typically C:\Windows), PROGRAMFILES (C:\Program Files), and PROGRAMFILES(x86) (C:\Program Files (x86)) are permitted to execute. These path rules are automatically added when the "Disallowed" security level is set as the default.

Figure 2-83 Setting Security Level Default



c. Customize the Allowlist Rules to enhance security by disallowing specific subfolders in the default allowed paths and to support organization application requirements.

i. Click the **Additional Rules** folder and apply the rules shown in Figure 2-84. This figure combines the NSA recommended path settings in addition to lab application requirements and for disabling installers and other executable content as indicated in the comments. *Organizations should audit their environments to determine the appropriate rules to define within the policy*.

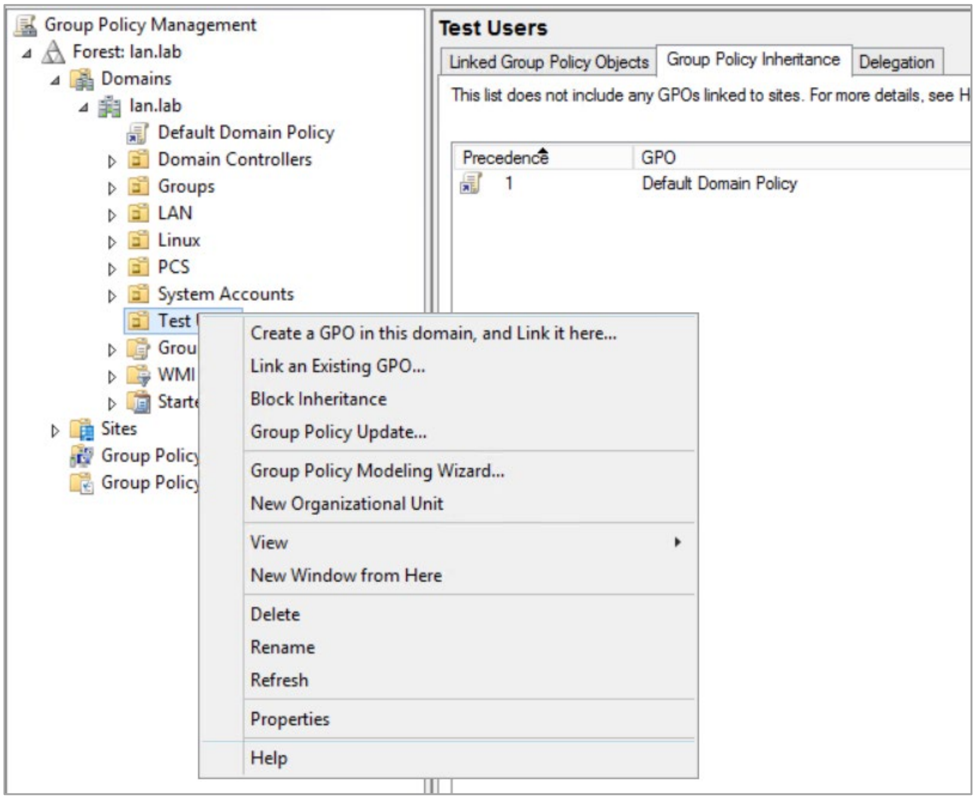
Figure 2-84 Additional Rules Defined for Lab Environment

Name	Type	Security Level	Description
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%	Path	Unrestricted	Default System Root Allow Rule
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\Debug	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\PCHEALTH\ERRORREP	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\Registration	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\catroot2	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\com\dmp	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\FxsTmp	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\spool\drivers\c...	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\spool\PRINTERS	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\System32\Tasks	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\Systme32\spool\SERVERS	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\SysWOW64\com\dmp	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\SysWOW64\FxsTmp	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\SysWOW64\Tasks	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\Tasks	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\Temp	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%\tracing	Path	Disallowed	Deny execution per NSA Guidance
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ProgramFilesDir (x86)%	Path	Unrestricted	Allow 32-bit Program Files on 64 bit systems.
%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ProgramFilesDir%	Path	Unrestricted	Default Program Files Directory Allow Rule
%USERPROFILE%\AppData\Local\Microsoft\OneDrive\OneDrive.exe	Path	Unrestricted	Temp rule for Workstations Allow OneDrive
%USERPROFILE%\Forescout Console 8.2.1	Path	Unrestricted	Temporary Rule to Allow Forescout Console
*.lnk	Path	Unrestricted	Allow Links to executables
*.msi	Path	Disallowed	Prevent installers from executing
\\%USERDNSDOMAIN%\Sysvol\	Path	Unrestricted	Allow Domain Login Scripts
C:\TwinCAT	Path	Unrestricted	Added to support CRS PLC Programming
E:\Program Files	Path	Unrestricted	Approved alternate Program Files Location
E:\Program Files (x86)	Path	Unrestricted	Approved alternate 32-bit Program Files location
runas.exe	Path	Disallowed	Deny execution per NSA Guidance

6. Link the GPO to the Test User OU:

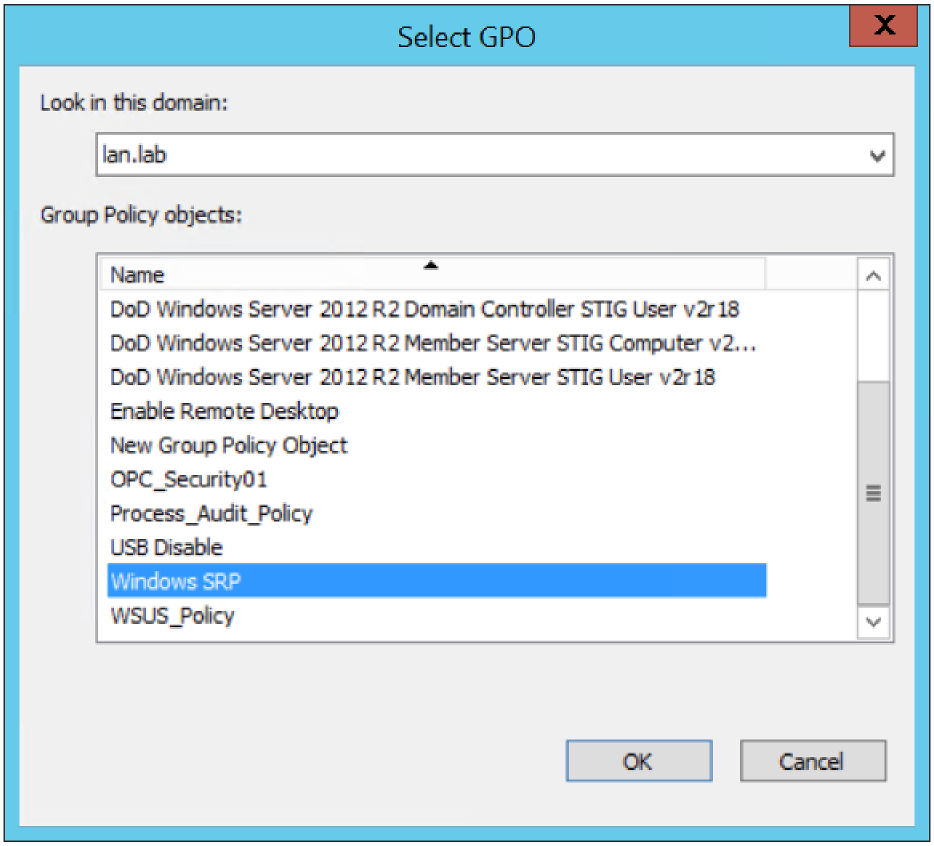
   a. In the Group Policy Management tool, right click the "Test User" OU and select **Link an Existing GPO** from the pop-up menu (**Figure** 2-85).

Figure 2-85 Menu Options for Accessing the Link an Existing GPO Option



b. In the dialog box, select the **Windows SRP GPO Object** from the list and click **OK** (Figure 2-86).

Figure 2-86 Dialog Box for Selecting GPO to Link



(Optional) Install GPO as the local policy on non-domain systems; for systems that are not joined to the domain, the nccoeUser and nccoeAdmin accounts are created as local user and administrator accounts, respectively. Additionally, the Windows SRP GPO is manually applied to the local system using the LGPO.exe application contained in the ZIP file from Step 3.

c. Create a Backup of the Windows SRP GPO Object:

   i. From the Group Policy Manager, select the **Group Policy Objects** folder and right-click on the Windows SRP GPO object.

   ii. Select the **Back Up…** option from the pop-up menu.

   iii. In the dialog box, choose a destination location such as *C:\Backup GPO Folder* or some other convenient location to place the files and click **Back Up**.

d. Copy the LGPO.exe along with the files created in the previous step to the non-domain computer system.

e. Login as an administrator on the non-domain computer and navigate to the **{GUID}\DomainSysvol\GPO\User** folder, which should contain the **registory.pol** file for the GPO.

f. Execute the following commands to apply the settings to the local nccoeUser and nccoeAdmin accounts:

```
lgpo.exe /u:nccoeUser registory.pol
lgpo.exe /u:nccoeAdmin registory.pol
```

Appendix A List of Acronyms

AAL	Application Allowlisting
AD	Active Directory
AF	Asset Framework
BAD	Behavioral Anomaly Detection
CRS	Collaborative Robotic System
CRADA	Cooperative Research and Development Agreement
CSF	NIST Cybersecurity Framework
CSMS	Cybersecurity for Smart Manufacturing Systems
DMZ	Demilitarized Zone
DNAT	Destination Network Address Translation
FOIA	Freedom of Information Act
GPO	Group Policy Object
HDD	Hard Disk Drive
ICS	Industrial Control System
IIS	Internet Information Services
IoT	Internet of Things
IT	Information Technology
LAN	Local Area Network
MFA	Multifactor Authentication
MTD	Moving Target Defense
NAT	Network Address Translation
NCCoE	National Cybersecurity Center of Excellence
NIST	National Institute of Standards and Technology
NISTIR	NIST Interagency or Internal Report
NSA	National Security Agency
NTP	Network Time Protocol
OT	Operational Technology

OU	Organizational Unit
PCS	Process Control System
PI	Process Information
PLC	Programmable Logic Controller
POU	Program Organizational Unit
RDP	Remote Desktop Protocol
SP	Special Publication
SPAN	Switch Port Analyzer
SRP	Software Restriction Policy
VDI	Virtual Desktop Interface
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network

Appendix B Build Architecture Diagrams

Figure B-1 Build 1 Architecture Diagram

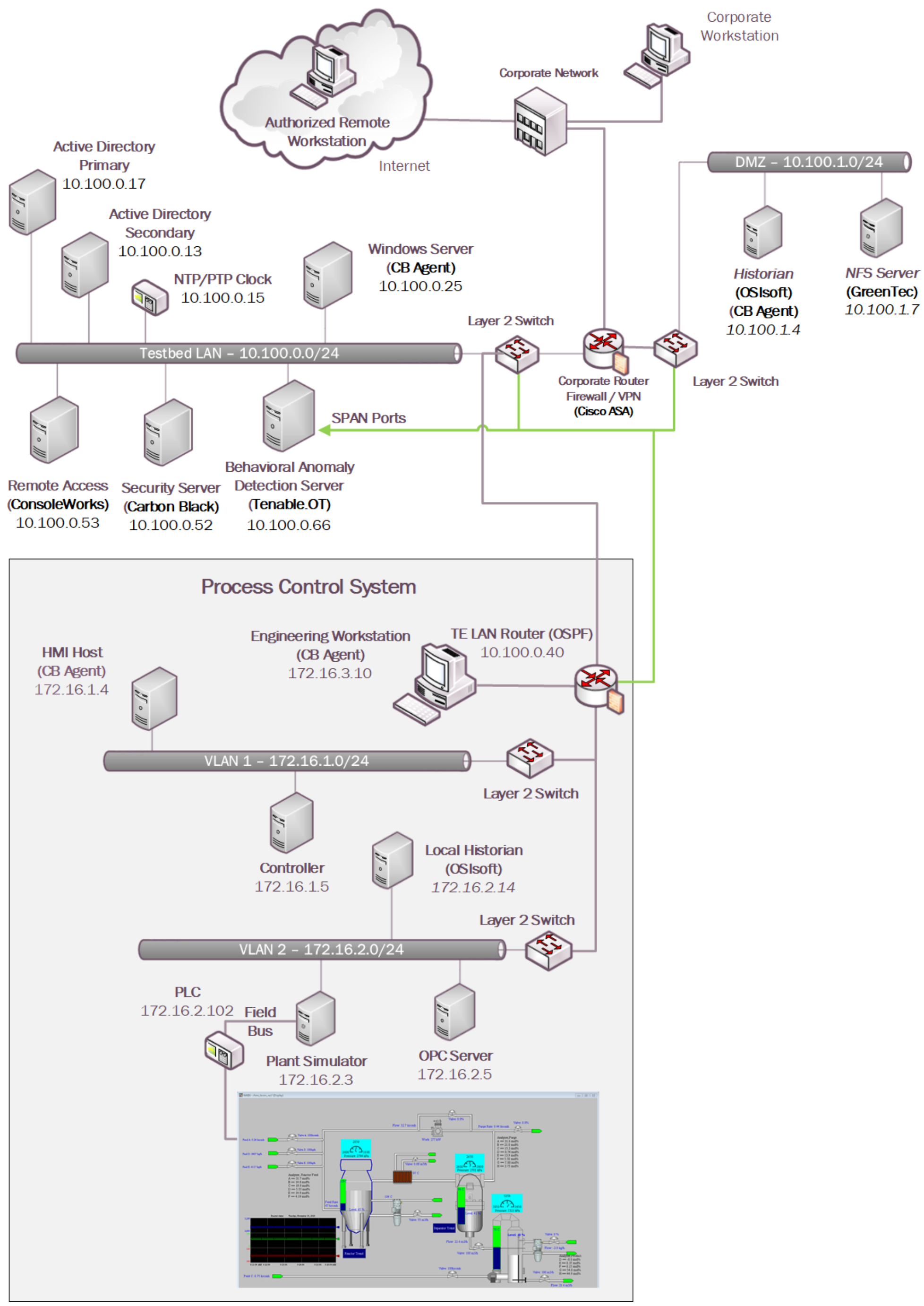
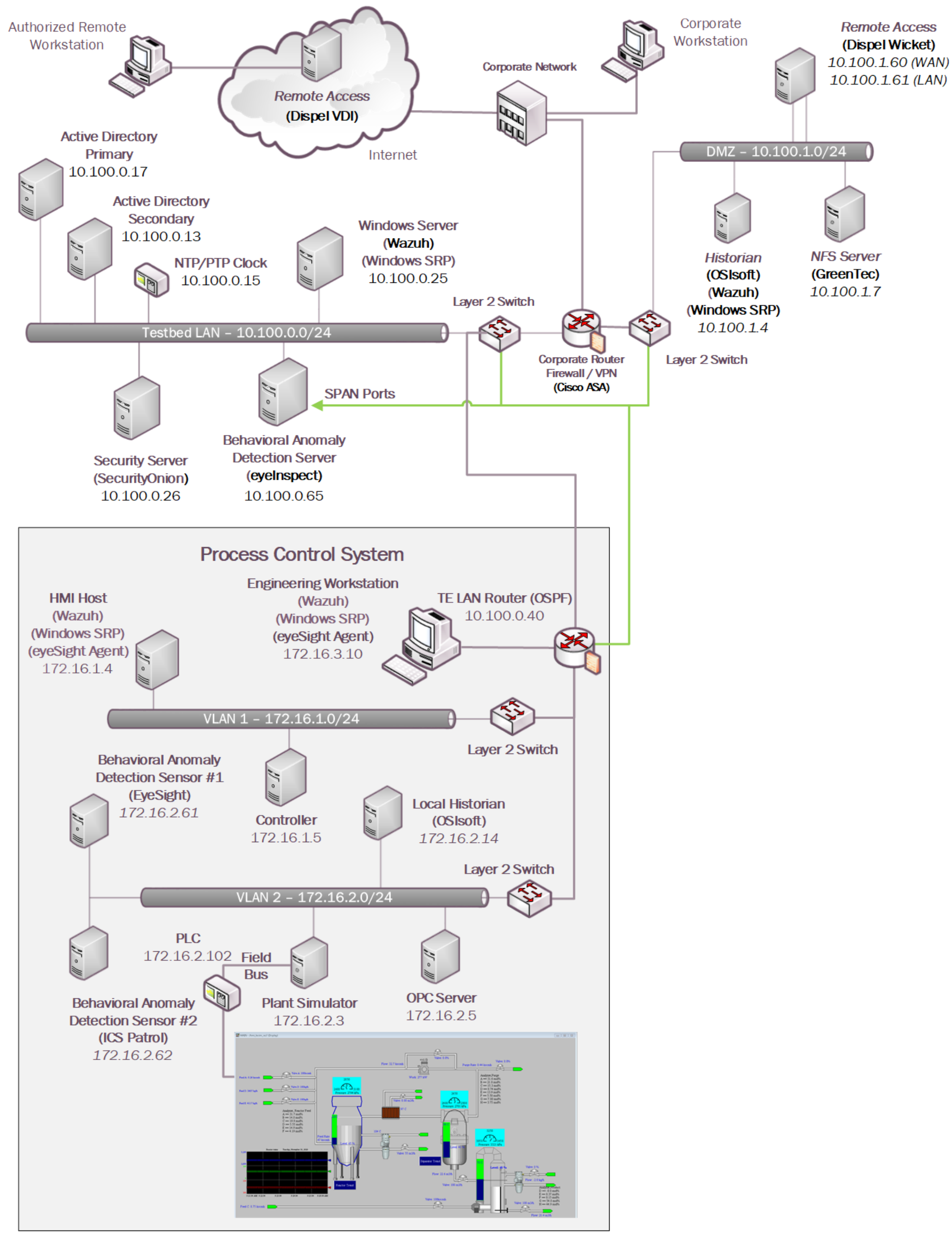


Figure B-2 Build 2 Architecture Diagram



This publication is available free of charge from: https://doi.org/10.6028/NIST.SP.1800-10.

Figure B-3 Build 3 Architecture Diagram

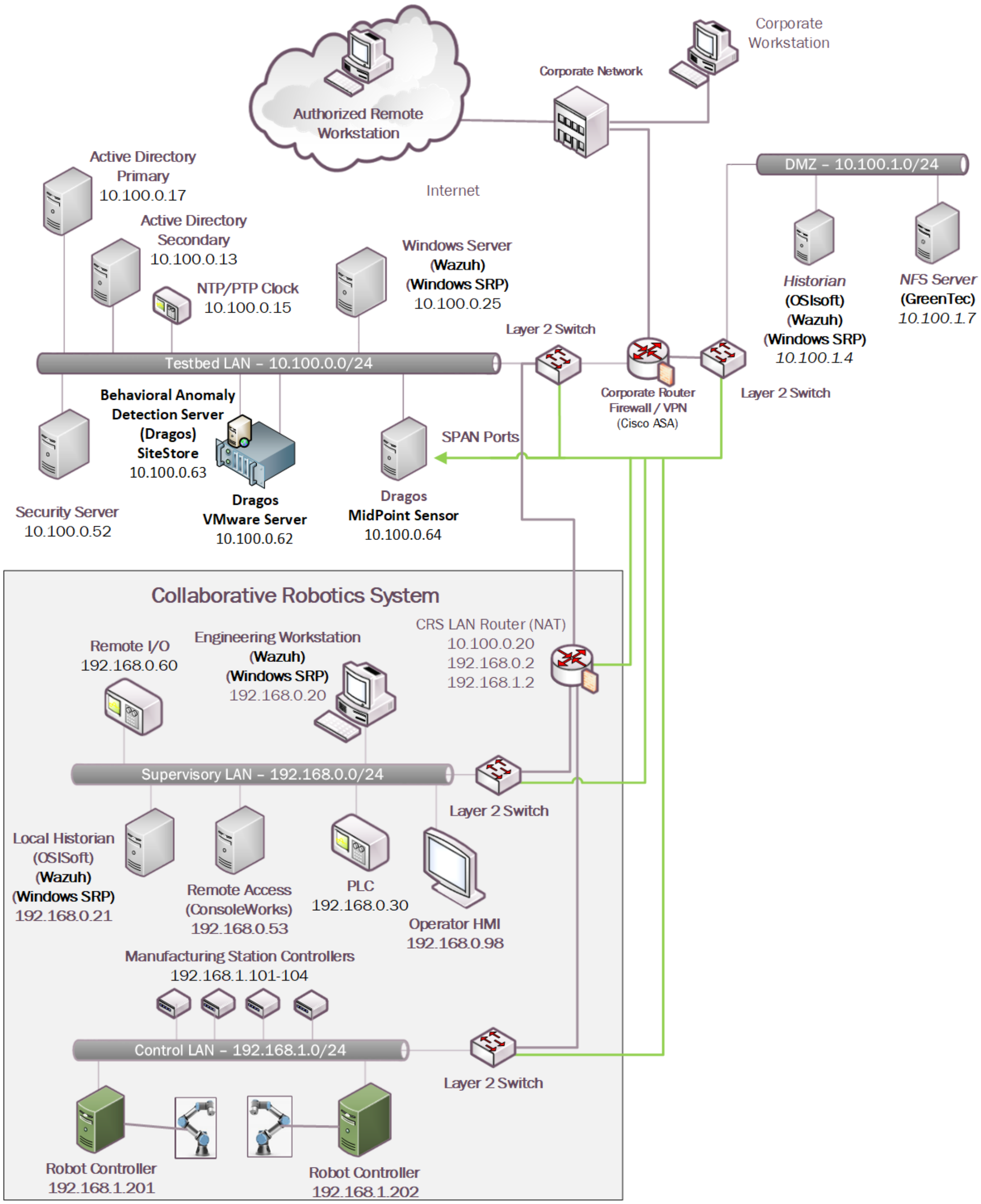


Figure B-4 Build 4 Architecture Diagram