NIST SPECIAL PUBLICATION 1800-25B

Data Integrity:

Identifying and Protecting Assets Against Ransomware and Other Destructive Events

Volume B:
Approach, Architecture, and Security Characteristics

Jennifer Cawthra
National Cybersecurity Center of Excellence
NIST

Michael Ekstrom
Lauren Lusty
Julian Sexton
John Sweetnam
The MITRE Corporation
McLean, Virginia

December 2020

FINAL

This publication is available free of charge from
https://doi.org/10.6028/NIST.SP.1800-25.

This publication is available free of charge from
https://www.nccoe.nist.gov/projects/building-blocks/data-integrity/identify-protect.

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.

National Institute of Standards and Technology Special Publication 1800-25B, Natl. Inst. Stand. Technol. Spec. Publ. 1800-25B, 50 pages, (December 2020), 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 ds-nccoe@nist.gov.

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

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, 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 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

Ransomware, destructive malware, insider threats, and even honest user mistakes present ongoing threats to organizations. Organizations' data, such as database records, system files, configurations, user files, applications, and customer data, are all potential targets of data corruption, modification, and destruction. Formulating a defense against these threats requires two things: a thorough knowledge of the assets within the enterprise, and the protection of these assets against the threat of data corruption and destruction. The NCCoE, in collaboration with members of the business community and vendors of cybersecurity solutions, has built an example solution to address these data integrity challenges.

Multiple systems need to work together to identify and protect an organization's assets against the threat of corruption, modification, and destruction. This project explores methods to effectively identify assets (devices, data, and applications) that may become targets of data integrity attacks, as well as the vulnerabilities in the organization's system that facilitate these attacks. It also explores methods to protect these assets against data integrity attacks using backups, secure storage, integrity checking mechanisms, audit logs, vulnerability management, maintenance, and other potential solutions

KEYWORDS

attack vector; asset awareness; data integrity; data protection; malicious actor; malware; ransomware.

ACKNOWLEDGMENTS

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

Name	Organization
Kyle Black	Bay Dynamics
Sunjeet Randhawa	Broadcom Inc.
Peter Romness	Cisco Systems
Matthew Hyatt	Cisco Systems
Hans Ismirnioglou	Cryptonite
Sapna George	Cryptonite
Justin Yackoski	Cryptonite
Steve Petruzzo	GreenTec USA
Steve Roberts	Micro Focus
Timothy McBride	NIST

Name	Organization
Christopher Lowde	Semperis
Thomas Leduc	Semperis
Darren Mar-Elia	Semperis
Kirk Lashbrook	Semperis
Mickey Bresman	Semperis
Jim Wachhaus	Tripwire
Humphrey Christian	Symantec Corporation
Jon Christmas	Symantec Corporation
Kenneth Durbin	Symantec Corporation
Matthew Giblin	Symantec Corporation
Nancy Correll	The MITRE Corporation
Chelsea Deane	The MITRE Corporation
Sallie Edwards	The MITRE Corporation
Milissa McGinnis	The MITRE Corporation
Karri Meldorf	The MITRE Corporation
Denise Schiavone	The MITRE Corporation

Name	Organization
Anne Townsend	The MITRE Corporation

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	Build Involvement
Symantec Corporation	Symantec Data Loss Prevention v15.1
Cisco Systems	Cisco ISE v2.4, Cisco Web Security Appliance v10.1
GreenTec USA	GreenTec WORMdisk v151228
Tripwire	Tripwire Log Center v7.3.1, Tripwire Enterprise v8.7, Tripwire IP360 v9.0.1
Micro Focus	Micro Focus ArcSight Enterprise Security Manager v7.0 Patch 2
Cryptonite	CryptoniteNXT v2.9.1
Semperis	Semperis Active Directory Forest Recovery v2.5, Semperis Directory Services Protector v2.7

Contents

List of Figures

List of Tables

1 Summary

Businesses face a near-constant threat of destructive malware, ransomware, malicious insider activities, and even honest mistakes that can alter or destroy critical data. These types of adverse events ultimately impact data integrity (DI). It is imperative for organizations to be able to identify assets that may be impacted by a DI attack and to protect their enterprise against such attacks.

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) built a laboratory environment to explore methods to identify and protect assets from a data corruption event in various information technology (IT) enterprise environments. The example solution outlined in this guide describes the solution built in the NCCoE lab. It encourages identification of vulnerabilities and assets that may be present in the enterprise, as well as several protections that can significantly mitigate the effects of DI attacks before they occur.

The goals of this NIST Cybersecurity Practice Guide are to help organizations confidently:

- identify systems, users, data, applications, and entities on the network
- identify vulnerabilities in enterprise components and clients
- baseline the integrity and activity of enterprise systems, in preparation for an attack
- create backups of enterprise data in advance of an attack
- protect these backups and other potentially important data against alteration
- manage enterprise health by assessing machine posture

For ease of use, a short description of the different sections of this volume follows.

- Section 1: Summary presents the challenge addressed by the NCCoE project, with an in-depth look at our approach, the architecture, and the security characteristics we used; the solution demonstrated to address the challenge; benefits of the solution; and technology partners that participated in building, demonstrating, and documenting the solution. The Summary also explains how to provide feedback on this guide.
- Section 2: How to Use This Guide explains how readers—business decision makers, program managers, and IT professionals (e.g., systems administrators)—might use each volume of the guide.
- Section 3: Approach offers a detailed treatment of the scope of the project and describes the assumptions on which the security platform development was based, the risk assessment that informed platform development, and the technologies and components that industry collaborators gave us to enable platform development.
- Section 4: Architecture describes the usage scenarios supported by project security platforms, including Cybersecurity Framework [1] functions supported by each component contributed by our collaborators.

- Section 5: Security Characteristics Analysis provides details about the tools and techniques we used to perform risk assessments.
- Section 6: Future Build Considerations is a brief treatment of other Data Security implementations NIST considers consistent with Framework Core Functions: Identify, Protect, Detect and Respond, and Recovery.

1.1 Challenge

Thorough collection of quantitative and qualitative data is important to organizations of all types and sizes. It can impact all aspects of a business, including decision-making, transactions, research, performance, and profitability. When these data collections sustain a DI attack caused by unauthorized insertion, deletion, or modification of information, the attack can affect emails, employee records, financial records, and customer data, rendering them unusable or unreliable. Some organizations have experienced systemic attacks that caused a temporary cessation of operations. One variant of a DI attack—ransomware—encrypts data and holds it hostage while the attacker demands payment for the decryption keys.

Before DI events occur, organizations should identify their assets and vulnerabilities and have defenses and preparations in place to preemptively mitigate the events. This reduces the workload of actions to take during and after an attack occurs, as well as the enterprise's data loss and number of successful attacks.

1.2 Solution

The NCCoE implemented a solution that incorporates appropriate actions before the start of a DI event. The solution comprises systems working together to identify and protect assets against a data corruption event in standard enterprise components. These components include mail servers, databases, end user machines, virtual infrastructure, and file share servers. Essential to protection of assets is understanding of what those assets are and what vulnerabilities they have.

The NCCoE sought existing technologies that provided the following capabilities:

- **inventory**
- **policy enforcement**
- **logging**
- **backups**
- **vulnerability management**
- **secure storage**
- **integrity monitoring**

In developing our solution, we used standards and guidance from the following sources, which can also provide your organization with relevant standards and best practices:

- NIST *Framework for Improving Critical Infrastructure Cybersecurity* (commonly known as the NIST Cybersecurity Framework) [1]
- NIST Interagency or Internal Report (NISTIR) 8050: *Executive Technical Workshop on Improving Cybersecurity and Consumer Privacy* [2]
- NIST Special Publication (SP) 800-30 Rev. 1: *Guide for Conducting Risk Assessments* [3]
- NIST SP 800-37 Rev. 1: *Guide for Applying the Risk Management Framework to Federal Information Systems: A Security Life Cycle Approach* [4]
- NIST SP 800-39: *Managing Information Security Risk* [5]
- NIST SP 800-40 Rev. 3: *Guide to Enterprise Patch Management Technologies* [6]
- NIST SP 800-53 Rev. 4: *Security and Privacy Controls for Federal Information Systems and Organizations* [7]
- Federal Information Processing Standard 140-3: *Security Requirements for Cryptographic Modules* [8]
- NIST SP 800-86: *Guide to Integrating Forensic Techniques into Incident Response* [9]
- NIST SP 800-92: *Guide to Computer Security Log Management* [10]
- NIST SP 800-100: *Information Security Handbook: A Guide for Managers* [11]
- NIST SP 800-34 Rev. 1: *Contingency Planning Guide for Federal Information Systems* [12]
- Office of Management and Budget, Circular Number A-130: Managing Information as a Strategic Resource [13]
- NIST SP 800-61 Rev. 2: *Computer Security Incident Handling Guide* [14]
- NIST SP 800-83 Rev. 1: *Guide to Malware Incident Prevention and Handling for Desktops and Laptops* [15]
- NIST SP 800-150: *Guide to Cyber Threat Information Sharing* [16]
- NIST SP 800-184: *Guide for Cybersecurity Event Recovery* [17]

1.3 Benefits

The NCCoE's practice guide can help your organization:

- develop a plan for identifying assets and vulnerabilities and protecting these assets from a cybersecurity event
- facilitate detection, response, and recovery from a DI event by collecting information about the enterprise before an attack occurs

- maintain integrity and availability of data critical to supporting business operations and revenue-generating activities
- manage enterprise risk (consistent with the foundations of the NIST Cybersecurity Framework)

2 How to Use This Guide

This NIST Cybersecurity Practice Guide demonstrates a standards-based reference design and provides users with the information they need to replicate the DI identify-and-protect solution. This reference design is modular and can be deployed in whole or in part.

This guide contains three volumes:

- NIST SP 1800-25A: *Executive Summary*
- NIST SP 1800-25B: *Approach, Architecture, and Security Characteristics* – what we built and why **(you are here)**
- NIST SP 1800-25C: *How-To Guides* – instructions for building the example solution

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

Business decision makers, including chief security and technology officers, will be interested in the *Executive Summary,* NIST SP 1800-25A, which describes the following topics:

- challenges that enterprises face in identifying assets and protecting them from DI events
- example solution built at the NCCoE
- benefits of adopting the example solution

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in this part of the guide, NIST SP 1800-25B, which describes what we did and why. The following sections will be of particular interest:

- Section 3.4.1, Risk, provides a description of the risk analysis we performed.
- Section 3.4.2, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices.

You might share the *Executive Summary,* NIST SP 1800-25A, with your leadership team members to help them understand the importance of adopting a standards-based solution to identify and protect assets from DI attacks.

IT professionals who want to implement such an approach will find the whole practice guide useful. You can use the how-to portion of the guide, NIST SP 1800-25C, to replicate all or parts of the build created in our lab. The how-to portion of the guide provides specific product installation, configuration, and integration instructions for implementing the example solution. We do not re-create 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 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 a DI identify-and-protect solution. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope 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, 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 ds-nccoe@nist.gov.

Acronyms used in figures can be found in the Acronyms appendix.

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

Based on key points expressed in NISTIR 8050, *Executive Technical Workshop on Improving Cybersecurity and Consumer Privacy* (2015), the NCCoE is pursuing a series of DI projects to map the Core Functions of the NIST Cybersecurity Framework. This project is centered on the Core Functions of Identify and Protect, which consist of identifying assets and protecting them from DI attacks. For instance, the first step in building a strategy requires an organization to inventory its assets. This involves identifying systems, applications, data sources, users, and other relevant entities that may be targets or facilitators of DI attacks. Once this exercise is complete, an organization can then create a customized strategy to protect the identified assets against the possibility of data corruption, modification, and destruction. NCCoE engineers working with a community of interest (COI) defined the requirements for this DI project.

Members of the COI, which include participating vendors referenced in this document, contributed to development of the architecture and reference design, providing technologies that meet the project requirements and assisting in installation and configuration of those technologies. The practice guide highlights the approach used to develop the NCCoE reference solution. Elements include risk assessment and analysis, logical design, build development, test and evaluation, and security control mapping. This guide aims to provide practical guidance to any organization interested in implementing a solution for identifying and protecting assets against a cybersecurity event.

3.1 Audience

This guide is intended for individuals responsible for implementing security solutions in organizations' IT support activities. Current IT systems, particularly in the private sector, often lack the ability to comprehensively identify enterprise assets that need protection from integrity attacks, as well as the protections themselves. The platforms demonstrated by this project, and the implementation information provided in these practice guides, permit integration of products to implement a data identification and protection system. The technical components will appeal to system administrators, IT managers, IT security managers, and others directly involved in the secure and safe operation of business IT networks.

3.2 Scope

The guide provides practical, real-world guidance on developing and implementing a DI solution consistent with the principles in the NIST *Framework for Improving Critical Infrastructure Cybersecurity*, Version 1.1 [1], specifically the Core Functions of Identify and Protect. The Identify Function emphasizes the development and implementation of the appropriate activities to discover and manage an organization's assets, services, and the threats to these assets and services. The Protect Function emphasizes development and implementation of activities that protect these assets and services from

cybersecurity events. Examples of outcomes within these Functions include asset inventory, logging, backups, vulnerability management, policy enforcement, and file/system integrity management.

3.3 Assumptions

This project is guided by the following assumptions:

- The solution was developed in a lab environment. The environment is based on a generic organization's IT enterprise—it uses services found commonly across typical enterprises, such as a database, a domain controller, a mail/web server, etc. It does not reflect the complexity of a production environment, for example, building across numerous physical locations, accommodating for extreme working conditions, or configuring systems to meet specific network/user needs. These demands can all increase the level of complexity needed to implement a DI solution.
- An organization has access to the skills and resources required to implement an asset identification and protection system.
- An organization is seeking to preemptively mitigate the damage a DI event would cause.

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 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.

We performed two types of risk assessments:

- Initial analysis of the risk factors discussed with financial, retail, and hospitality institutions: this analysis led to creation of the DI project and desired security posture. See NISTIR 8050, *Executive Technical Workshop on Improving Cybersecurity and Consumer Privacy,* for additional participant information.

- Analysis of how to secure the components within the solution and minimize any vulnerabilities they might introduce: see Section 5, Security Characteristic Analysis.

3.4.1 Risk

Using the guidance in NIST's series of publications concerning risk, we worked with financial institutions and the Financial Sector Information Sharing and Analysis Center to identify the most compelling risk factors encountered by this business group. We participated in conferences and met with members of the financial sector to define the main security risks to business operations. From these discussions came identification of an area of concern—DI. We produced the practice guide *Data Integrity: Recovering from Ransomware and Other Destructive Events,* which primarily focused on the recovery aspect of DI. From responses to the recovery project, we also identified a need for guidance in identifying and protecting assets from DI attacks.

When considering risk from the perspective of identifying and protecting assets prior to a cybersecurity event, we must consider not only the impact of an event on an organization's assets but also the threats to those assets and the potential vulnerabilities these threats could exploit.

When discussing threats to an organization's assets from the perspective of DI, we consider the following factors:

- malware
- insider threats
- accidents caused by human error
- compromise of trusted systems

Types of vulnerabilities we consider in relation to these threats are:

- zero-day vulnerabilities
- vulnerabilities due to outdated or unpatched systems
- custom software vulnerabilities/errors
- social engineering and user-driven events
- poor access control

Finally, we consider the potential impact on an organization from a DI event:

- systems incapacitated
- modification/deletion of organization's assets
- negative impact on the organization's reputation

Analyses of the threats, vulnerabilities, and potential impact to an organization give us an understanding of the risk to an organization with respect to DI. NIST SP 800-39, *Managing Information Security Risk,* focuses on the business aspect of risk, namely at the enterprise level. This understanding is essential for any further risk analysis, risk response/mitigation, and risk monitoring activities. The following summary lists the strategic risk areas we identified and their mitigations:

- Impact on system function: ensuring the availability of accurate data or sustaining an acceptable level of DI reduces the risk of systems' availability being compromised.
- Cost of implementation: implementing asset identification and protection from DI events once and using it across all systems may reduce system continuity costs.
- Compliance with existing industry standards contributes to the industry requirement to maintain a continuity of operations plan.
- Maintenance of reputation and public image helps reduce level and likelihood of impact as well as facilitates the information required for impact reduction.
- Increased focus on DI includes not just loss of confidentiality but also harm from unauthorized alteration of data (per NISTIR 8050).

We subsequently translated the risk factors identified to security Functions and Subcategories within the NIST Cybersecurity Framework. In Table 3-1, we mapped the categories to NIST SP 800-53 Rev. 4 controls.

3.4.2 Security Control Map

As explained in Section 3.4.1, we identified the Cybersecurity Framework Functions and Subcategories that we wanted the reference design to support, through a risk analysis process. This was a critical first step in designing the reference design and example implementation to mitigate the risk factors. Table 3-1 lists the addressed Cybersecurity Framework Functions and Subcategories and maps them to relevant NIST standards, industry standards, and controls and best practices. The references provide solution validation points in that they list specific security capabilities that a solution addressing the Cybersecurity Framework Subcategories would be expected to exhibit. Organizations can use Table 3-1 to identify the Cybersecurity Framework Subcategories and NIST SP 800-53 Rev. 4 controls they are interested in addressing.

When cross-referencing Functions of the Cybersecurity Framework with product capabilities used in this practice guide, it is important to consider:

- This practice guide, though primarily focused on Identify/Protect Functions also uses DE.CM-8 and RS.MI-3, Detect and Respond Subcategories respectively. This is primarily because these two Subcategories deal with vulnerability discovery and mitigation, which are techniques used to prevent future damage and are not as useful for preventing attacks previously exploited a given vulnerability. Often, it is unlikely that an organization will be able to resolve a newly

discovered vulnerability during an attack; for attacks where patches are available, it can be dangerous to allow updates on a compromised system.

- Not all the guidance of Cybersecurity Framework Subcategories can be implemented using technology. Any organization executing a DI solution would need to adopt processes and organizational policies that support the reference design. For example, some of the Subcategories within the Cybersecurity Framework Function known as Identify are processes and policies that should be developed prior to implementing recommendations.

Table 3-1 DI Reference Design Cybersecurity Framework Core Components Map

Cybersecurity Framework v1.1				Standards and Best Practices	
Function	**Category**	**Subcategory**	**NIST SP 800-53 R4**	**ISO/IEC 27001:2013**	**NIST SP 800-181**
IDENTIFY (ID)	Asset Management (ID.AM)	ID.AM-1: Physical devices and systems within the organization are inventoried.	CM-8, PM-5	A.8.1.1, A.8.1.2	OM-STS-001
		ID.AM-2: Software platforms and applications within the organization are inventoried.	CM-8, PM-5	A.8.1.1, A.8.1.2, A.12.5.1	OM-STS-001
	Risk Assessment (ID.RA)	ID.RA-1: Asset vulnerabilities are identified and documented.	CA-2, CA-7, CA-8, RA-3, RA-5, SA-5, SA-11, SI-2, SI-4, SI-5	A.12.6.1, A.18.2.3	PR-VAM-001
		ID.RA-2: Cyber threat intelligence is received from information sharing forums and sources.	SI-5, PM-15, PM-16	A.6.1.4	CO-OPL-002
		ID.RA-5: Threats, vulnerabilities, likelihoods, and impacts are used to determine risk.	RA-2, RA-3, PM-16	A.12.6.1	SP-SYS-001

Cybersecurity Framework v1.1				Standards and Best Practices	
Function	**Category**	**Subcategory**	**NIST SP 800-53 R4**	**ISO/IEC 27001:2013**	**NIST SP 800-181**
PROTECT (PR)	Access Control (PR.AC)	PR.AC-1: Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes.	AC-1, AC-2, IA-1, IA-2, IA-3, IA-4, IA-5, IA-6, IA-7, IA-8, IA-9, IA-10, IA-11	A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3	SP-DEV-001, OV-PMA-003
		PR.AC-3: Remote access is managed.	AC-1, AC-17, AC-19, AC-20, SC-15	A.6.2.1, A.6.2.2, A.11.2.6, A.13.1.1, A.13.2.1	SP-SYS-001, OM-ADM-001
		PR.AC-4: Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties.	AC-1, AC-2, AC-3, AC-5, AC-6, AC-14, AC-16, AC-24	A.6.1.2, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5	OM-STS-001
		PR.AC-5: Network integrity is protected (e.g., network segregation, network segmentation).	AC-4, AC-10, SC-7	A.13.1.1, A.13.1.3, A.13.2.1, A.14.1.2, A.14.1.3	OM-NET-001
	Data Security (PR.DS)	PR.DS-1: Data-at-rest is protected.	MP-8, SC-12, SC-28	A.8.2.3	OM-DTA-002
		PR.DS-2: Data-in-transit is protected.	SC-8, SC-11, SC-12	A.8.2.3, A.13.1.1, A.13.2.1, A.13.2.3, A.14.1.2, A.14.1.3	OM-DTA-002, PR-CDA-001

Cybersecurity Framework v1.1				Standards and Best Practices	
Function	**Category**	**Subcategory**	**NIST SP 800-53 R4**	**ISO/IEC 27001:2013**	**NIST SP 800-181**
		PR.DS-6: Integrity checking mechanisms are used to verify software, firmware, and information integrity.	SC-16, SI-7	A.12.2.1, A.12.5.1, A.14.1.2, A.14.1.3, A.14.2.4	OM-DTA-001
	Information Protection Processes and Procedures (PR.IP)	PR.IP-1: A baseline configuration of information technology/industrial control systems is created and maintained, incorporating security principles (e.g., concept of least functionality).	CM-2, CM-3, CM-4, CM-5, CM-6, CM-7, CM-9, SA-10	A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4	SP-ARC-001
		PR.IP-3: Configuration change control processes are in place.	CM-3, CM-4, SA-10	A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4	SP-DEV-001, OM-ANA-001
		PR.IP-4: Backups of information are conducted, maintained, and tested.	CP-4, CP-6, CP-9	A.12.3.1, A.17.1.2, A.17.1.3, A.18.1.3	SP-SYS-001
		PR.IP-9: Response plans (Incident Response and Business Continuity) and recovery plans (Incident Recovery and Disaster Recovery) are in place and managed.	CP-2, CP-7, CP-12, CP-13, IR-7, IR-8, IR-9, PE-17	A.16.1.1, A.17.1.1, A.17.1.2, A.17.1.3	PR-CIR-001
		PR.IP-10: Response and recovery plans are tested.	CP-4, IR-3, PM-14	A.17.1.3	SP-SYS-001

Cybersecurity Framework v1.1				Standards and Best Practices	
Function	**Category**	**Subcategory**	**NIST SP 800-53 R4**	**ISO/IEC 27001:2013**	**NIST SP 800-181**
		PR.IP-12: A vulnerability management plan is developed and implemented.	RA-3, RA-5, SI-2	A.12.6.1, A.14.2.3, A.16.1.3, A.18.2.2, A.18.2.3	SP-RSK-002
	Maintenance (PR.MA)	PR.MA-1: Maintenance and repair of organizational assets are performed and logged, with approved and controlled tools.	MA-2, MA-3, MA-5, MA-6	A.11.1.2, A.11.2.4, A.11.2.5, A.11.2.6	OM-ADM-001
		PR.MA-2: Remote maintenance of organizational assets is approved, logged, and performed in a manner that prevents unauthorized access.	MA-4	A.11.2.4, A.15.1.1, A.15.2.1	SP-TRD-001
	Protective Technology (PR.PT)	PR.PT-1: Audit/log records are determined, documented, implemented, and reviewed in accordance with policy.	AU Family	A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1	OV-LGA-002
		PR.PT-3: The principle of least functionality is incorporated by configuring systems to provide only essential capabilities.	AC-3, CM-7	A.9.1.2	PR-CDA-001, OM-ANA-001

Cybersecurity Framework v1.1				Standards and Best Practices	
Function	**Category**	**Subcategory**	**NIST SP 800-53 R4**	**ISO/IEC 27001:2013**	**NIST SP 800-181**
		PR.PT-4: Communications and control networks are protected.	AC-4, AC-17, AC-18, CP-8, SC-7, SC-19, SC-20, SC-21, SC-22, SC-23, SC-24, SC-25, SC-29, SC-32, SC-36, SC-37, SC-38, SC-39, SC-40, SC-41, SC-43	A.13.1.1, A.13.2.1, A.14.1.3	SP-ARC-002
DETECT (DE)	Security Continuous Monitoring (DE.CM)	DE.CM-8: Vulnerability scans are performed.	RA-5	A.12.6.1	SP-TRD-001
RESPOND (RS)	Mitigation (RS.MI)	RS.MI-3: Newly identified vulnerabilities are mitigated or documented as accepted risks.	CA-7, RA-3, RA-5	A.12.6.1	PR-CIR-001

3.5 Technologies

Table 3-2 lists all the technologies used in this project and provides a mapping among the generic application term, the specific product used, and the security control(s) the product provides. Refer to Table 3-1 for an explanation of the NIST Cybersecurity Framework Subcategory codes.

Please note that PR.AC-4 is not included in this table. Access controls are detailed more thoroughly in other NCCoE practice guides [18], [19]. For the purposes of this practice guide, we assume a minimal Active Directory setup with an administrator and several users.

Table 3-2 Products and Technologies

Component	Product	Function	Cybersecurity Framework Subcategories
Inventory	Cisco ISE v2.4 Symantec Data Loss Prevention (DLP) v15.1	• Identification and status information for users • Identification and status information for devices • Identification and status information for software • Identification and status information for data assets	ID.AM-1, ID.AM-2, PR.AC-1, PR.PT-2
Vulnerability Management	Tripwire IP360 v9.0.1	• Identification for vulnerabilities on various systems in the enterprise • An interface for managing/prioritizing vulnerabilities, based on organizational needs	ID.RA-1, ID.RA-5, PR.IP-12, DE.CM-8, RS.MI-3
Policy Enforcement	Cisco ISE v2.4	• Enforce machine posture across an enterprise • Quarantine machines that do not comply with organizational policy	ID.RA-1, PR.AC-3, PR.MA-1, PR.MA-2, RS.MI-3
Integrity Monitoring	Tripwire Enterprise v8.7 Semperis Directory Services Protector (DSP) v2.7	• Baselines integrity activity for data • Baselines integrity activity for Active Directory • Provides file hashes and integrity baselines for files and software, regardless of file type	PR.DS-6, PR.IP-3, PR.PT-1
Logging	Micro Focus ArcSight Enterprise Security Manager (ESM) v7.0 Patch 2	• Provides auditing and logging capabilities configurable to corporate policy • Provides logs of baseline network operations	PR.IP-1, PR.IP-3, PR.PT-1

Component	Product	Function	Cybersecurity Framework Subcategories
	Tripwire Log Center v7.3.1	• Provides logs of database activity and database backup operations • Provides logs of integrity changes • Provides logs of some user activity of monitored systems	
Backups	Semperis Active Directory Forest Recovery (ADFR) v2.5 FileZilla v0.9.60.2 OPEN SOURCE Duplicati v2.0.3.3 OPEN SOURCE	• Backs up Active Directory information • Backs up systems • Backs up configurations • Backs up organizational data	PR.DS-1, PR.IP-3, PR.IP-4, PR.IP-9, PR.IP-10
Secure Storage	GreenTec WORMdisk v151228	• Provides immutable storage • Provides configurable prevention of backup modification	PR.DS-1, PR.IP-4
Network Protection	CryptoniteNXT v2.9.1	• Prevents unapproved network communication • Prevents malicious reconnaissance • Quarantines unauthorized machines on the network	ID.AM-1, PR.AC-1, PR.AC-3, PR.AC-5, PR.DS-2, PR.PT-4
Denylisting	Cisco Web Security Appliance v10.1	• Provides capability to denylist websites • Provides capability to denylist communication with malicious or disallowed IP addresses	PR.AC-3, PR.AC-5, PR.DS-2, PR.PT-4

4 Architecture

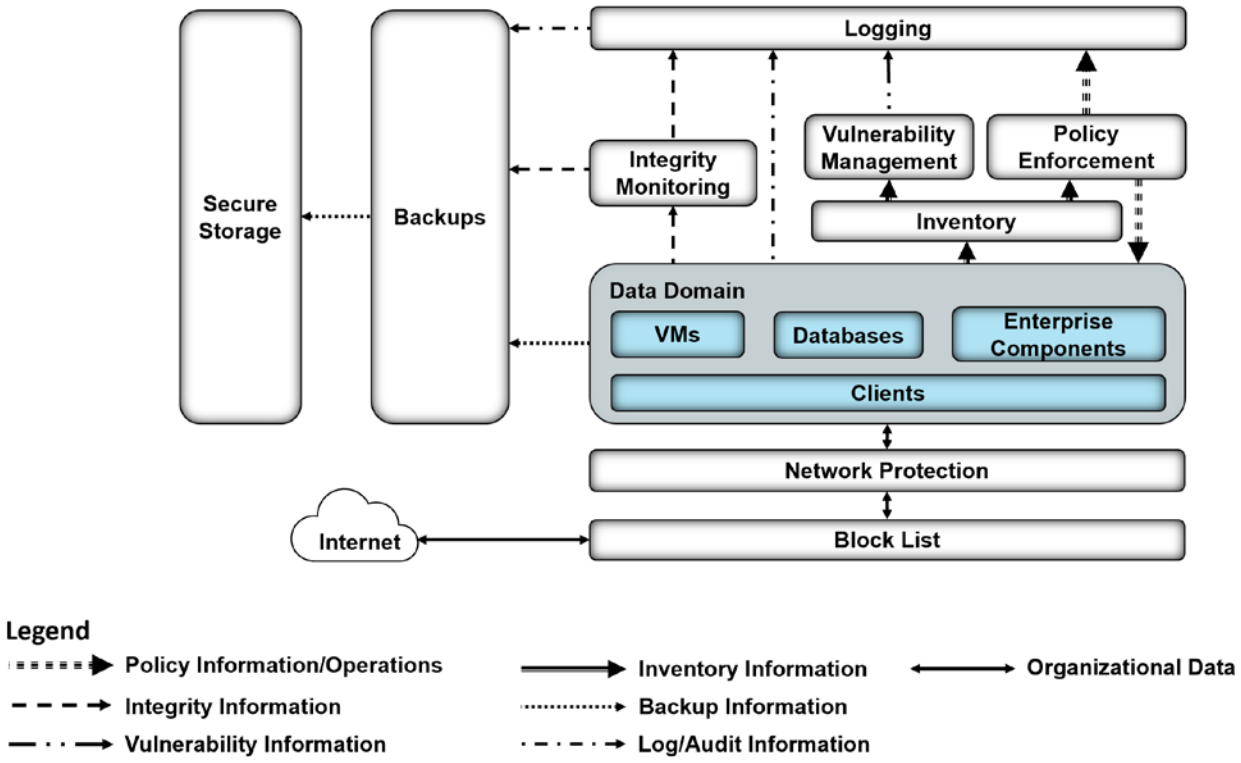
This section presents the high-level architecture used for implementation of a DI solution that identifies and protects assets from ransomware and other destructive events.

4.1 Architecture Description

4.1.1 High-Level Architecture

The DI solution is designed to address the security Functions and Subcategories described in Table 3-1 and is composed of the capabilities illustrated in Figure 4-1.

Figure 4-1 DI Identify and Protect High-Level Architecture



- **Inventory** allows discovering and keeping track of devices connected to the enterprise.
- **Vulnerability management** provides a mechanism for analyzing various system and network components, for a better understanding of resolved and unresolved vulnerabilities in the enterprise.
- **Policy enforcement** uses feedback from logs and vulnerability management to target machines with unresolved vulnerabilities and maintain overall enterprise health.

- **Integrity monitoring** establishes baselines of file/system integrity.
- **Logging** records and stores all the log files produced by the components within the enterprise.
- **Backups** allow components within the enterprise to produce backups.
- **Secure storage** allows data storage with additional data protection measures, such as Write Once Read Many (WORM) technologies. Data encryption can also be used, but this will not inherently protect data against corruption.
- **Network protection** can defend an enterprise network against both intrusion and lateral movement of malicious actors and programs.
- **Denylisting** can filter allowed programs or network communications. Often, this may be provided in the form of a firewall or even an allowlist, but products exist that allow finer-grained control over these filters.

These capabilities work together to provide the functions of Identify and Protect for the reference architecture. The inventory capability allows accurate and complete discovery and status reporting of all network assets. The inventory capability feeds into vulnerability management, which analyzes the assets and network for vulnerabilities. Vulnerability management feeds its information into the logging capability, which aggregates and collects logs from various sources for use as a baseline of normal system operations. Policy enforcement uses information from logging and vulnerability management, to repair vulnerabilities found in the enterprise and maintain the system with up-to-date patches. Integrity monitoring records normal file/system integrity information to be used as a baseline in the event of an attack and forwards this information to the logging capability as part of the organization's baseline. Backups create periodic backups of organizational data to be used in a cybersecurity event. Secure storage allows storing files—such as backups, gold images, logs, or configuration files—in a format that cannot be corrupted, because files cannot be altered or changed while in storage.

4.1.2 Architecture Components

4.1.2.1 Inventory

The inventory capability allows discovering and visualizing the enterprise's network as well as the present network devices. This component also informs the other components in the enterprise, providing information such as what systems to monitor, back up, and scan for vulnerabilities. This component provides the basic knowledge of what assets there are to protect.

For the inventory capability, we use a combination of two products: Cisco ISE and Symantec DLP. Cisco ISE provides inventory capabilities for machines, devices, and users on its network and can use that information in tandem with other capabilities. Symantec DLP provides data asset inventory, allowing organizations to identify potentially sensitive data.

4.1.2.2 Vulnerability Management

The vulnerability management capability allows scanning and managing vulnerabilities across the enterprise. It provides a priority system for these vulnerabilities, as well as logs on existing vulnerabilities and potentially resolved vulnerabilities. The information produced by this capability informs the policy enforcement capability, which aims to fix the discovered vulnerabilities or quarantine the machine until they are fixed.

For the vulnerability management capability, we use Tripwire IP360. Tripwire IP360 is a vulnerability scanner and management tool, which can scan a variety of hosts for known vulnerabilities and report on the results. Furthermore, the tool can manage and assign risk levels to these vulnerabilities, allowing security teams to effectively manage vulnerabilities throughout the enterprise.

4.1.2.3 Policy Enforcement

Through various mechanisms, the policy enforcement capability maintains the health of the enterprise. Policy enforcement acts on log information provided by the inventory and vulnerability management capabilities, often with the help of a security team, to ensure the health and compliance of enterprise systems. This can include mechanisms such as pushing software updates, resolving vulnerabilities, or quarantining noncompliant machines, but the capabilities of policy enforcement tools vary from product to product.

For policy enforcement, we use Cisco ISE. Cisco ISE can identify machines on its network and perform a posture check on these machines. This can entail checking that certain services are enabled, that anti-malware is installed, or that certain files are present. Using this information, Cisco ISE can then disable network access to noncompliant machines.

4.1.2.4 Integrity Monitoring

Integrity monitoring provides the ability to test, understand, and measure attacks that occur on files and components within the enterprise. When considering DI from the perspective of protecting assets prior to an attack, it is important to establish an integrity baseline for files and systems across the enterprise, to be used in comparison with daily operations. The value of integrity monitoring becomes clear both during and after an attack. Alerts can be set to notify the security team to act when abnormal changes are detected to a file or system, such as changes made at abnormal times or by users who typically do not make changes to these assets. Furthermore, the information produced by integrity monitoring systems can be used to inform a recovery process; they provide information about what changes happened, when changes began to take place, as well as what programs were involved in the changes.

For integrity monitoring, we use a combination of two tools: Tripwire Enterprise and Semperis Directory Services Protector. Tripwire Enterprise is a file integrity monitoring tool that establishes a baseline for integrity activity within the enterprise. This baseline is used in the event of an attack, to detect and alert on changes within the enterprise as well as aid recovery should it be necessary. Semperis Directory

Services Protector also provides integrity monitoring, but for Active Directory it allows granular rollbacks of Active Directory changes and provides a baseline for any attacks on the enterprise account configuration.

4.1.2.5 Logging

Logging from each enterprise component serves several functions in an architecture that aims to identify and protect assets. Logs are produced through integrity monitoring, which aids in establishing a baseline for the enterprise's daily activity. Logs are also produced through vulnerability scanning and asset inventory, which inform policy enforcement: maintaining up-to-date systems requires information about what systems exist in the enterprise and their status.

For logging, we use a combination of two tools: Micro Focus ArcSight and Tripwire Log Center (TLC). While TLC's purpose in this build is primarily to collect, transform, and forward logs from Tripwire IP360 and Tripwire Enterprise to ArcSight, ArcSight performs a wider function. ArcSight collects logs from various sources in the enterprise, such as vulnerability management, backups, network protection, denylisting, inventory, integrity monitoring, as well as Windows event logs and Ubuntu syslogs. This widespread collection aims to provide a baseline for activity throughout the enterprise. ArcSight can analyze and alert, which can be used in the event of an attack, but it requires thorough log collection from all components of the enterprise.

4.1.2.6 Backups

The backups capability backs up both the organization's data and data from other components, such as logs and integrity information. These backups are most often used as part of the Recover Function as part of the restoration process. Backups must be taken prior to an event to be useful, though; the restoration process requires backups from before the event to adequately restore a system.

The configuration of this capability needs to align with the tempo of the enterprise. For example, if an enterprise performs thousands of transactions per hour per day, then a backup solution that performs a backup only once a day would not adequately provide for the enterprise. This type of configuration would allow a potentially large data loss. If backups occur every morning and a loss of DI happened at the end of the day, then a full day's worth of transactions would be lost. The decision for the correct configuration of backups is determined by an organization's risk tolerance.

For the backups capability, we use a combination of two open-source tools: FileZilla and Duplicati. FileZilla is a user-based File Transfer Protocol (FTP) server with the option to force FTP over Transport Layer Security (TLS). It allows control over where individual users/groups store files, and its primary purpose in this build is as a receptacle for backups produced by Duplicati. Duplicati is a client-based backup system configured on individual hosts to back up to a provided FTP server. It packages and encrypts backups before sending them to the FTP server, potentially on a schedule.

We also use Semperis ADFR to provide more fine-grained backups for Active Directory. As Active Directory is often critical to enterprise operations, Semperis ADFR is designed to work off-site in the event of a disaster.

4.1.2.7 Secure Storage

Secure storage stores the most critical files for an enterprise. These include backup data, configuration files, logs, golden images, and other files critical to both system operation and the organization's mission. Additional measures need to be applied to provide increased security to these files so they are not subject to attacks or corruption.

For secure storage, we use GreenTec's WORMdisk, a transparent hard disk that can prevent any data deletion and modification at a firmware level. WORMdisks provide a user-friendly graphical user interface and a command line interface for automating locking and disk rotation. In this architecture they are used primarily to store backups to prevent any damage to the backups, but they can be used at the discretion of the organization to store other critical files.

4.1.2.8 Network Protection

Network protection defends the network against threats that require network movement. This should preemptively protect against lateral movement, in which malware or a malicious actor attempts to spread across machines in the network. Furthermore, it should also protect against external threats attempting to gain access to the network.

For network protection, we use CryptoniteNXT. CryptoniteNXT provides zero-trust moving-target defense for the network it protects. This means that all enterprise communication goes through the CryptoniteNXT device, which provides granular access control for allowed types of communication. This allows defense against lateral propagation. Furthermore, as internet protocol (IP) addresses are dynamic and managed by CryptoniteNXT, reconnaissance is significantly more difficult for attackers on and outside the network.

4.1.2.9 Denylisting

Denylisting enables control of allowed communications and applications within an enterprise. This may include restricting installed software on enterprise machines to a predefined list or specifically disallowing software. Furthermore, it should restrict network communication with websites, servers, or external actors as well as restrict based on protocol or port usage. Some of these capabilities are covered by firewalls, but further control can allow more complex policies based on the organization's needs.

For the denylisting capability we use Cisco Web Security Appliance (WSA). Cisco WSA enables enterprises to denylist web traffic through a proxy. This allows for prevention of malware downloads from known malicious websites as identified by site reputation updates from Cisco Talos threat

intelligence. These websites can also be identified through the implementation of a Detect and Respond build and can also be provided by an integration with other information sharing services.

5 Security Characteristic Analysis

The purpose of the security characteristic analysis is to understand the extent to which the project meets its objective of demonstrating a DI identify-and-protect solution. 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. It is assumed that devices are hardened. Testing these devices would reveal only weaknesses in implementation that would not be relevant to those adopting this reference architecture.

5.2 Build Testing

The purpose of the security characteristic analysis is to understand the extent to which the building block meets its objective of identifying enterprise assets and vulnerabilities. Furthermore, the project aims to protect these assets prior to the start of an attack. In addition, it seeks to understand the security benefits and drawbacks of the reference design. To accomplish this, we created a set of use cases—each an individual attack on DI with different aspects to test various parts of the build.

When doing this, we aim not to test individual components for their capabilities but rather for the ability of the architecture to deal with these use cases. Furthermore, as this architecture is focused on defending against attacks before they happen, the resolutions to these use cases are primarily preventative rather than responsive.

5.3 Scenarios and Findings

One aspect of our security evaluation involved assessing how well the reference design addresses the security characteristics it was intended to support. The 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 Cybersecurity Framework Subcategories as a basis for organizing our analysis allowed us to systematically consider how well the reference design supports the intended security characteristics.

Below is a list of the scenarios created to test various aspects of this architecture. More detailed resolutions and mappings of these scenarios' requirements to the Cybersecurity Framework can be found in Appendix D.

5.3.1 Ransomware via Web Vector and Self-Propagation

5.3.1.1 Scenario

The following scenario was simulated to test the architecture's defense against ransomware.

A user mistakenly downloads ransomware from an external web server. When the user executes this malicious software, it generates a cryptographic key, which is sent back to the external web server. The malware then utilizes a privilege escalation exploit to propagate across the network. The malicious software encrypts files on the machines it propagated to, and it demands payment in exchange for decrypting these files.

5.3.1.2 Resolution

This build provides a significant defense in depth against this use case to prevent the majority of its functions from taking place.

The **denylisting** capability is used to prevent the user from reaching the malicious site that hosts the ransomware, preventing the download before it happens.

The **vulnerability management** capability is used to detect the vulnerability exploited by the ransomware to propagate, allowing resolution before the attack occurs.

The **network protection** capability is used to prevent the ransomware's propagation by disallowing network traffic between computers on the network, through a traffic allowlist policy.

The **inventory** capability is used to identify the enterprise's assets for backup and monitoring.

The **backups** capability is used to take backups of potential ransomware targets before the attack hits, nullifying the effects of potential attacks on files.

The **integrity monitoring** capability, in tandem with the **logging** capability, is used to take a baseline of the file system, so that an attack on the file system is detected and the scope can be identified.

5.3.1.3 Other Considerations

Malware comes in many forms and from many places, and as a result, requires a defense in depth against it. For example, though preventing a piece of malware from getting on enterprise systems may be possible through denylisting a website, it is often impossible to have full knowledge of all malicious websites before an attack happens. Because of this, other tools are necessary to prevent the effects of malware at every step of its potential execution, and preparation is necessary to mitigate effects.

It is important to improve upon these capabilities over time by learning from attacks on the enterprise and from attacks on other enterprises. Both information-sharing technologies and after-the-fact analysis of attacks can inform capabilities to prevent future attacks.

5.3.2 Destructive Malware via USB Vector

5.3.2.1 Scenario

The following scenario was simulated to test the architecture's defense against destructive malware.

A user finds an unmarked Universal Serial Bus (USB) device and inserts it into their system. The USB device contains malicious software that may run automatically or with user interaction. The malicious software modifies and deletes the user's files, removing text from text files and entirely deleting any media files it finds. The software does not offer a recovery mechanism as ransomware might, aiming only to corrupt files.

5.3.2.2 Resolution

This build provides two main layers of defense against this scenario: backups and Integrity baselining.

The **integrity monitoring** capability provides a baseline for file system activity as a point of comparison post-modification/deletion.

The **logging** capability provides a baseline for events across the enterprise, including typical USB and file modification activity.

The **backups** capability provides the ability to take backups of the file system, allowing restoration of files after the incident is resolved.

5.3.2.3 Other Considerations

A use case involving USBs is often best prevented through organizational training. In some cases, just the action of inserting the USB is enough to destroy an entire system on a physical level. Furthermore, not all malicious USBs will be file systems with auto-run malware on them—they can come disguised as keyboards or use lower-level attacks. Because of this, it is important for organizations to educate members on the dangers of unknown USB insertion, while also preparing if the attack occurs anyway.

5.3.3 Accidental VM Deletion via Maintenance Script

5.3.3.1 Scenario

The following scenario was simulated to test the architecture's defense against DI events that occur on virtual machines (VMs).

A routine maintenance script on the system causes an error. During a move operation in the Hyper-V system, the script deletes an important VM. A maintenance script with an error of this type could be a side effect of a normal system function or an error made by a member of the organization. The build is expected to mitigate the damage caused to VMs in such an incident.

5.3.3.2 Resolution

This build provides two main layers of defense against this scenario: backups and Integrity baselining.

The **integrity monitoring** capability provides a baseline for virtual machine activity, as a point of comparison post-deletion.

The **logging** capability provides a baseline for events across the enterprise, including typical Hyper-V activity.

The **backups** capability enables backups of entire VMs. In the event of a deletion, these backups can be used to restore the VMs.

5.3.3.3 Other Considerations

The backups capability can also be installed on individual VMs, given proper networking, to back up the contents of VMs if desired. This will likely depend on the needs of the organization.

5.3.4 Backdoor Creation via Email Vector

5.3.4.1 Scenario

The following scenario was simulated to test the architecture's defense against malicious email attachments.

A user unknowingly opens a malicious attachment they received in an email. When opened, the attachment quietly fetches files from an external web server. It then creates several unapproved backdoor accounts on the authentication server. The build is expected to mitigate the impacts of such an incident.

5.3.4.2 Resolution

The build provides several layers of defense against this use case. The **integrity monitoring** capability provides a baseline for Active Directory as a point of comparison against a compromised system. Furthermore, it also provides a baseline of the file system, to aid in identifying the malicious file during and after the attack has happened.

The **logging** capability provides a baseline for activity across the enterprise, including the name of the account used to create the backdoors.

Lastly, the **denylisting** capability is used to prevent web requests to the malicious web server. This capability is informed by capabilities in the Respond Category of the Cybersecurity Framework.

5.3.4.3 Other Considerations

Note that for this scenario, prevention of the downloads before an attack happens requires organizations to know what web servers are "known bad." Organizations can acquire this knowledge in two ways: through threat-sharing services and through self-information as part of the Respond Category of the Cybersecurity Framework. The former refers to services that collect the names of malicious domains and share them with customers. The latter refers to the addition of known-bad websites to the denylist after they are detected as malicious through the organization's own logs and analytics during or after an event. This build allows protecting against attacks given this knowledge, but the knowledge must be gained in some way first.

Another defense that can partially prevent this use case is by denylisting the sender of the phishing email or sorting it into spam. However, as this is typically a function of the email provider and not a separate security solution, it is out of scope for this build.

5.3.5 Database Modification via Malicious Insider

5.3.5.1 Scenario

The following scenario was simulated to test the architecture's defense against unwanted database modification.

A malicious insider has access to an enterprise database through a web page. The insider leverages a vulnerability in the web page to delete a large portion of the database. Though this scenario deals with a web vulnerability, other vulnerabilities could be used to modify the database undesirably. The build is expected to mitigate a user's potential impact on the database.

5.3.5.2 Resolution

This build provides two main layers of defense against this scenario: backups and Integrity baselining.

The **integrity monitoring** capability provides a baseline for database activity as a point of comparison post-deletion.

The **logging** capability provides a baseline for events across the enterprise, including typical database activity.

The **backups** capability enables backups of the entire database. In the event of a deletion, these backups can be used to restore the database.

5.3.5.3 Other Considerations

Creating backups of the entire database may, in some cases, be undesirable, particularly for enterprises that heavily use the database. For these cases, we recommend built-in database backups. Microsoft Structured Query Language databases have built-in backups that can be more granular than a full database backup.

For many applications, though, a periodic backup of the entire database is sufficient and potentially can be used in tandem with built-in database backups.

5.3.6 File Modification via Malicious Insider

5.3.6.1 Scenario

The following scenario was simulated to test the architecture's defense against malicious file and backup modification.

A malicious insider is assumed to have stolen administrator-level credentials through nontechnical means. The insider, using these credentials, uses remote Windows PowerShell sessions to uniformly modify employee stock information across several machines, to the insider's benefit. This attack will also target the enterprise's backups system, to modify all records of the previous stock information. The aspects of the build described above are expected to mitigate the ability of the user to target and modify enterprise data and backups. The method of securing administrator credentials will be considered out of scope for this solution.

5.3.6.2 Resolution

The build provides several layers of defense against this use case. Because this use case specifically targets the backups, the solution includes mechanisms for protecting and monitoring the backups.

The **inventory** capability is used to identify potentially sensitive information across the enterprise.

The **integrity monitoring** capability is used to baseline file activity, both for backups and for organizational files.

This information is forwarded to the **logging** capability for analysis.

The **backups** capability is used to take encrypted backups of the file system, preventing targeted attacks against information in the backups.

The **secure storage** capability is used to prevent write-access to the backups once taken, allowing a guarantee of modification/deletion protection for backups stored on the disk.

5.3.6.3 Other Considerations

A significant trade-off between memory and frequency of backups occurs when implementing a secure storage solution for backups. As WORM space may be limited by the number of disks purchased or by a cloud service's limitations, it is important for organizations to consider the cost of storing all backups in secure storage, especially for organizations that frequently take backups to reduce the loss of data.

5.3.7 Backdoor Creation via Compromised Update Server

5.3.7.1 Scenario

The following scenario was simulated to test the architecture's defense against compromised update servers.

An update server that services an enterprise machine is compromised and provides an update to the enterprise machine that contains a backdoor. The update contains a vulnerable version of vsftpd, allowing a malicious actor root access into the machine updated by the compromised server. The build is expected to mitigate the impact of a compromised update server.

5.3.7.2 Resolution

The build provides several layers of defense against this use case. The **integrity monitoring** capability is used to baseline the integrity of both files and programs, as an intrusion via compromised update server can potentially affect both. This aids in early detection and recovery.

The **backups** capability is used to back up the file system, to preemptively mitigate the damage done by the intrusion.

The **denylisting** capability is used to denylist the compromised update server, to prevent use of the update server by other machines.

5.3.7.3 Other Considerations

To prevent updates through denylisting, organizations should either use their denylisting capability as a transparent proxy or ensure that the update mechanism uses the proxy; the process for configuring this will differ between update mechanisms. The denylisting and network protection capabilities are especially important in the event of a breach, as these two can help prevent the spread of the intrusion.

5.3.8 New Employee

5.3.8.1 Scenario

The following scenario was simulated to test the architecture's identification capabilities with respect to machines and vulnerabilities.

A new employee joins the organization and connects their machine to the network. The machine, however, is not up-to-date on its patches and poses a security risk to the organization. The build is expected to be able to identify the machine and its noncompliance with organizational maintenance policy.

5.3.8.2 Resolution

The build provides several layers of defense against this use case. The **inventory** capability provides logs and information about newly connected machines, including operating system, MAC address, IP address, and date of login. It also generates logs for the **logging** capability to collect and use for comparison against a baseline in the event of an incident.

The **policy enforcement** capability provides the ability to grant or deny network access based on the machine's posture—essentially, this verifies existence of security software and machine update status before the machine is ever allowed to use the network.

Lastly, the **Vulnerability Management** capability detects and keeps track of vulnerabilities on the newly discovered machine, allowing better understanding of the machine's vulnerabilities before and after it is allowed onto the network.

5.3.8.3 Other Considerations

Though this use case primarily targets desktops, similar considerations should be taken for enterprises that aim to include employee-owned mobile devices. These devices should be inventoried and scanned for relevant security posture, before being allowed to join the network.

6 Future Build Considerations

The NCCoE is creating an overarching guide to combining the architectures of the various DI projects: Identify and Protect, Detect and Respond, and Recover. These architectures have some commonalities, such as integrity monitoring, as well as some potential integrations and cycles that could not be expressed in just one of the practice guides. The different functions of the Cybersecurity Framework are intended to prepare and inform one another, and the overarching guide addresses those issues.

The NCCoE is also considering additional data security projects that map to the Cybersecurity Framework Core Functions of Identify, Protect, Detect, Respond, and Recover. These projects will focus on data confidentiality—the defense of enterprise systems from attacks that would compromise the secrecy of data.

Appendix A List of Acronyms

COI	community of interest
DI	data integrity
DSP	Directory Services Protector
ESM	Enterprise Security Manager
IT	Information Technology
ISO/IEC	International Organization for Standardization/International Electrotechnical Commission
NCCoE	National Cybersecurity Center of Excellence
NIST	National Institute of Standards and Technology
NIST IR	NIST Interagency Report
RMF	Risk Management Framework
SP	Special Publication
TLC	Tripwire Log Center
TLS	Transport Layer Security
USB	Universal Serial Bus
VM	Virtual Machine
vsftpd	Very Secure File Transfer Protocol Daemon
WORM	Write Once Read Many
WSA	Web Security Appliance

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
Audit	Independent review and examination of records and activities to assess the adequacy of system controls and ensure compliance with established policies and operational procedures SOURCE: CNSSI 4009-2015
Backdoor	An undocumented way of gaining access to a computer system. A backdoor is a potential security risk. SOURCE: National Institute of Standards and Technology (NIST) Special Publication (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
Compromise	Disclosure of information to unauthorized persons, or a violation of the security policy of a system in which unauthorized intentional or unintentional disclosure, modification, destruction, or loss of an object may have occurred SOURCE: NIST SP 800-32

Continuous Monitoring	Maintaining ongoing awareness to support organizational risk decisions SOURCE: NIST SP 800-137
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)
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
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 Security Risk	The risk to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, other organizations, and the Nation due to the potential for unauthorized access, use, disclosure, disruption, modification, or destruction of information and/or information systems SOURCE: CNSSI 4009-2015 (NIST SP 800-30 Rev. 1)
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)
Insider	An entity inside the security perimeter that is authorized to access system resources but uses them in a way not approved by those who granted the authorization

	SOURCE: NIST SP 800-82 Rev. 2 (RFC 4949)
Kerberos	An authentication system developed at the Massachusetts Institute of Technology (MIT). Kerberos is designed to enable two parties to exchange private information across a public network. SOURCE: NIST SP 800-47
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
Privacy	Assurance that the confidentiality of, and access to, certain information about an entity is protected SOURCE: NIST SP 800-130
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
Vulnerability	Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source SOURCE: FIPS 200 (Adapted adapted from CNSSI 4009)

Appendix C References

[1] Sedgewick, *Framework for Improving Critical Infrastructure Cybersecurity,* Version 1.1, National Institute of Standards and Technology, Gaithersburg, Maryland, Apr. 2018, 55 pp. Available: https://www.nist.gov/cyberframework/framework.

[2] L. Kauffman, N. Lesser and B. Abe, *Executive Technical Workshop on Improving Cybersecurity and Consumer Privacy*, NISTIR 8050, National Institute of Standards and Technology, Gaithersburg, Maryland, April 2015, 155pp. Availabe: https://nccoe.nist.gov/sites/default/files/library/nistir-8050-draft.pdf.

[3] G. Stoneburner, *et al*., *Guide for Conducting Risk Assessments*, NIST Special Publication (SP), 800-30 Revision 1, National Institute of Standards and Technology, Gaithersburg, Maryland, September 2012, 95 pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-30r1.

[4] R. Ross, *et al*., *Guide for Applying the Risk Management Framework to Federal Information Systems*, NIST Special Publication (SP) 800-37, National Institute of Standards and Technology, Gaithersburg, Maryland, February 2010, 101pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-37r1.

[5] R. Ross *et al*., *Managing Information Security Risk*, NIST Special Publication (SP) 800-39, National Institute of Standards and Technology, Gaithersburg, Maryland, March 2011, 87pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-39.

[6] M. Souppaya *et al.*, *Guide to Enterprise Patch Management Technologies*, NIST Special Publication (SP) 800-40 Revision 3, National Institute of Standards and Technology, Gaithersburg, Maryland, July 2013, 25pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-40r3.

[7] R. Ross *et al.*, *Security and Privacy Controls for Federal Information Systems and Organizations*, NIST Special Publication (SP) 800-53 Revision 4, National Institute of Standards and Technology, Gaithersburg, Maryland, April 2013, 461pp. Available: https://doi.org/10.6028/NIST.SP.800-53r4.

[8] U.S. Department of Commerce. Security Requirements for Cryptographic Modules, Federal Information Processing Standards (FIPS) Publication 140-3, Mar. 2019, 65pp. Available: https://csrc.nist.gov/publications/detail/fips/140/3/final.

[9] K. Kent *et al.*, *Guide to Integrating Forensic Techniques into Incident Response*, NIST Special Publication (SP) 800-86, National Institute of Standards and Technology, Gaithersburg, Maryland, August 2006, 121pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-86.

[10] K. Kent and M. Souppaya, *Guide to Computer Security Log Management*, NIST Special Publication (SP) 800-92, National Institute of Standards and Technology, Gaithersburg, Maryland, September 2006, 72pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-92.

[11] P. Bowen *et al.*, *Information Security Handbook: A Guide for Managers*, NIST Special Publication (SP) 800-100, National Institute of Standards and Technology, Gaithersburg, Maryland, October 2006, 178pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-100.

[12] M. Swanson et al., Contingency Planning Guide for Federal Information Systems, NIST Special Publication (SP) 800-34 Revision 1, National Institute of Standards and Technology, Gaithersburg, Maryland, May 2010, 148pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-34r1.

[13] Office of Management and Budget (OMB), *Management of Federal Information Resources*, OMB Circular No. A-130, November 2000. Available: https://obamawhitehouse.archives.gov/sites/default/files/omb/assets/OMB/circulars/a130/a130revised.pdf.

[14] P. Cichonski *et al.*, *Computer Security Incident Handling Guide*, NIST Special Publication (SP) 800-61 Revision 2, National Institute of Standards and Technology, Gaithersburg, Maryland, August 2012, 79pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-61r2.

[15] M. Souppaya and K. Scarfone, *Guide to Malware Incident Prevention and Handling for Desktops and Laptops*, NIST Special Publication (SP) 800-83 Revision 1, National Institute of Standards and Technology, Gaithersburg, Maryland, July 2013, 46pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-83r1.

[16] C. Johnson *et al.*, *Guide to Cyber Threat Information Sharing*, NIST Special Publication (SP) 800-150, National Institute of Standards and Technology, Gaithersburg, Maryland, October 2016, 42pp. Available: http://dx.doi.org/10.6028/NIST.SP.800-150.

[17] M. Bartock *et al.*, *Guide for Cybersecurity Event Recovery*, NIST Special Publication (SP) 800-184, National Institute of Standards and Technology, Gaithersburg, Maryland, December 2016, 52pp. http://dx.doi.org/10.6028/NIST.SP.800-184.

[18] J. Banoczi *et al.*, *Access Rights Management,* NIST Special Publication (SP) 1800-9, National Institute of Standards and Technology, Gaithersburg, Maryland, October 2017. Available: https://www.nccoe.nist.gov/projects/use-cases/access-rights-management.

[19] B. Fisher *et al.*, *Attribute Based Access Control,* NIST Special Publication (SP) 1800-3, National Institute of Standards and Technology, Gaithersburg, Maryland, September 2017. Available: https://www.nccoe.nist.gov/projects/building-blocks/attribute-based-access-control.

Appendix D Functional Evaluation

A functional evaluation of the data integrity (DI) example implementation, as constructed in our laboratory, was conducted to verify that it meets its objective of identifying assets and vulnerabilities within the enterprise. Furthermore, the project aims to protect these assets prior to an attack. The evaluation verified that the example implementation could perform the following functions:

- discover assets on the network
- discover and mitigate vulnerabilities in assets on the network
- protect data from modification prior to an attack
- provide a baseline for daily activity and asset integrity

Section D.1 describes the format and components of the functional test cases. Each functional test case is designed to assess the capability of the example implementation to perform the functions listed above and detailed in Section D.1.

D.1 Data Integrity Functional Test Plan

One aspect of our security evaluation involved assessing how well the reference design addresses the security characteristics it was intended to support. The 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 that Subcategory. The cited sections provide validation points that the example solution is expected to exhibit. Using the Cybersecurity Framework Subcategories as a basis for organizing our analysis allowed us to systematically consider how well the reference design supports the intended security characteristics.

This plan includes the test cases necessary to conduct the functional evaluation of the DI example implementation, which is currently deployed in a lab at the National Cybersecurity Center of Excellence. The implementation tested is described in Section 4.

Each test case consists of multiple fields that collectively identify the goal of the test, the specifics required to implement the test, and how to assess the results of the test. Table 6-1 describes each field in the test case.

Table 6-1 Test Case Fields

Test Case Field	Description
Parent Requirement	Identifies the top-level requirement or the series of top-level requirements leading to the testable requirement
Testable requirement	Drives the definition of the remainder of the test case fields. Specifies the capability to be evaluated.

Test Case Field	Description
Description	Describes the objective of the test case
Associated Cybersecurity Framework Subcategories	Lists the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53 Rev. 4 controls addressed by the test case
Preconditions	The starting state of the test case. Preconditions indicate various starting state items, such as a specific capability configuration required or specific protocol and content.
Procedure	The step-by-step actions required to implement the test case. A procedure may consist of a single sequence of steps or multiple sequences of steps (with delineation) to indicate variations in the test procedure.
Expected results	The expected results for each variation in the test procedure
Actual results	The observed results
Overall result	The overall result of the test as pass/fail. In some test cases, determination of the overall result may be more involved, such as determining pass/fail based on a percentage of errors identified.

D.2 Data Integrity Use Case Requirements

Table 6-2 identifies the DI functional requirements addressed in the test plan and associated test cases.

Table 6-2 Capability Requirements

Capability Requirement (CR) ID	Parent Requirement	Sub requirement 1	Test Case
CR 1	The DI example implementation shall identify and protect assets against malware that encrypts files and displays notice demanding payment.		
CR 1.a		Vulnerability in Active Directory server is identified.	Data Integrity IP-1
CR 1.b		User is blocked from visiting malicious site.	Data Integrity IP-1

Capability Requirement (CR) ID	Parent Requirement	Sub requirement 1	Test Case
CR 1.c		Downloads from site are blocked.	Data Integrity IP-1
CR 1.d		Vulnerability is patched.	Data Integrity IP-1
CR 1.e		Ransomware cannot send information to home server.	Data Integrity IP-1
CR 1.f		Backups are taken.	Data Integrity IP-1
CR 1.g		File integrity information is baselined.	Data Integrity IP-1
CR 2	The DI example implementation shall identify and protect assets against malware inserted via Universal Serial Bus (USB) that modifies and deletes user data.		Data Integrity IP-2
CR 2.a		Backups are taken.	Data Integrity IP-2
CR 2.b		File integrity information is baselined.	Data Integrity IP-2
CR 3	The DI example shall identify and protect virtual machines against deletion.		Data Integrity IP-3
CR 3.a		Backups of virtual machines are taken.	Data Integrity IP-3
CR 4	The DI example implementation shall identify and protect assets against malware received via phishing email.		Data Integrity IP-4

Capability Requirement (CR) ID	Parent Requirement	Sub requirement 1	Test Case
CR 4.a		Downloads from the spreadsheet are blocked.	Data Integrity IP-4
CR 4.b		Backups of configurations are taken.	Data Integrity IP-4
CR 4.c		Configuration integrity information is baselined.	Data Integrity IP-4
CR 5	The DI example implementation shall identify and protect the database against changes made through a web server vulnerability in custom code.		Data Integrity IP-5
CR 5.a		Vulnerability is identified.	Data Integrity IP-5
CR 5.b		Vulnerability is resolved.	Data Integrity IP-5
CR 5.c		Backups of database are taken.	Data Integrity IP-5
CR 5.d		Database integrity information is baselined.	Data Integrity IP-5
CR 6	The DI example implementation shall identify and protect assets against targeted modification by malicious insiders with elevated privileges.		Data Integrity IP-6
CR 6.a		Backups are taken.	Data Integrity IP-6
CR 6.b		File integrity information is baselined.	Data Integrity IP-6

Capability Requirement (CR) ID	Parent Requirement	Sub requirement 1	Test Case
CR 6.c		Backups are encrypted.	Data Integrity IP-6
CR 6.d		Backups are stored securely.	Data Integrity IP-6
CR 7	The DI example implementation shall identify and protect assets against an intrusion via compromised update server.		Data Integrity IP-7
CR 7.a		Downloads from site are temporarily blocked.	Data Integrity IP-7
CR 7.b		Backups are taken.	Data Integrity IP-7
CR 7.c		Program integrity information is baselined.	Data Integrity IP-7
CR 7.d		File integrity information is baselined.	Data Integrity IP-7
CR 8	The DI example implementation shall identify new and unmaintained assets on the network.		Data Integrity IP-8
CR 8.a		Machines that are new to the network are identified.	Data Integrity IP-8
CR 8.b		Machines that are not up-to-date are identified.	Data Integrity IP-8

D.3 Test Case: Data Integrity IP-1

Table 6-3 Test Case ID: Data Integrity IP-1

Parent requirement	(CR 1) The DI example implementation shall identify and protect assets against malware that encrypts files and displays notice demanding payment.
Testable requirement	(CR 1.a) Vulnerability identification, (CR 1.b, 1.c, 1.e) Denylisting, (CR 1.d) Maintenance, (CR 1.f) Backups, (CR 1.g) Integrity Baselining
Description	Show that the DI solution can identify and resolve vulnerabilities and protect against ransomware.
Associated Cybersecurity Framework Subcategories	ID.AM-1, ID.AM-2, ID.RA-1, ID.RA-2, ID.RA-6, DE.CM-8, PR.IP-12, RS.MI-3, PR.IP-4, PR.DS-1, PR.DS-6, PR.PT-1, PR.MA-2
Preconditions	User navigates to a malicious website and clicks on an ad for a virus cleaner. The virus cleaner is actually ransomware, which propagates across the domain and encrypts user files.
Procedure	The **denylisting** capability is used to prevent access to and downloads from known malicious sites. The **inventory** capability is used to identify organizational assets and devices. The **network protection** capability is used to prevent the propagation of ransomware across the enterprise. The **vulnerability management** capability is used to identify vulnerabilities that allow malware to propagate. The **integrity monitoring** and **logging** collect integrity information and baseline the file system. The **backups** capability is used to take backups of the file system.
Expected Results (pass)	The vulnerability that allows the ransomware to propagate is identified (CR 1.a). The user cannot access the site when it is blocked (CR 1.b).

	The user cannot download the ransomware from the site when it is blocked (CR 1.c). The build can identify (and possibly execute) a fix for the vulnerability. When the fix is made, the ransomware is unable to propagate (CR 1.d). The ransomware is unable to communicate with its home server when the site is blocked (CR 1.e). The build can take backups of file systems (CR 1.f). The build can take and log integrity baselines of file systems (CR 1.g).
Actual Results	**Cisco WSA (denylisting)** stops the user from accessing the site when it is blocked. **Cisco ISE (inventory)** is used to identify devices on the network. **Symantec DLP (inventory)** is used to identify organizational data assets on monitored machines. **CryptoniteNXT (network protection)** prevents propagation of ransomware through an allowlist of approved communications in the enterprise. **Tripwire IP360 (vulnerability management)** detects vulnerabilities in Active Directory that allow ransomware to propagate. **Tripwire Enterprise (integrity monitoring)** and **ArcSight ESM (logging)** baseline critical data assets across the enterprise. **Duplicati** and **FileZilla (backups)** create backups of organizational data as a contingency, should ransomware be able to affect any systems.
Overall Result	Pass. All requirements for this use case are met.

D.4 Test Case: Data Integrity IP-2

Table 6-4 Test Case ID: Data Integrity IP-2

Parent requirement	(CR 2) The DI example implementation shall identify and protect assets against malware inserted via USB that modifies and deletes user data.

Testable requirement	(CR 2.a) Backups, (CR 2.b) Integrity Baselining
Description	Show that the DI solution can preemptively protect against destructive malware.
Associated Cybersecurity Framework Subcategories	PR.IP-4, PR.DS-1, PR.DS-6, PR.PT-1
Preconditions	A user inserts an unidentified USB drive into their computer. They click on a file on the drive, which immediately destroys any files on their machine.
Procedure	**Backups** schedules and creates backups of the user's documents. The **integrity monitoring** capability is used to take integrity baselines of the file system. **Logging** collects logs and baselines system activity.
Expected Results (pass)	The build can take backups of file systems (CR 2.a). The build can take and log integrity baselines of file systems (CR 2.b).
Actual Results	**Duplicati** and **FileZilla (backups)** are used to take and store backups of the user's documents. **Tripwire Enterprise (integrity monitoring)** is used to take an integrity baseline of the user's file system prior to the malicious USB drive being inserted into the computer. **ArcSight ESM (logging)** takes a baseline of system activity prior to the USB drive being inserted into the computer.
Overall Result	Pass. All requirements for this use case are met.

D.5 Test Case: Data Integrity IP-3

Table 6-5 Test Case ID: Data Integrity IP-3

Parent requirement	(CR 3) The DI example implementation shall identify and protect virtual machines against deletion.
Testable requirement	(CR 3.a) Backups
Description	Show that the DI solution can preemptively protect against data integrity events that involve virtual machines (VMs).

Associated Cybersecurity Framework Subcategories	PR.IP-4, PR.DS-1
Preconditions	A routine maintenance script contains an error that accidentally deletes a VM.
Procedure	The **backups** capability is used to schedule and create backups of a VM.
Expected Results (pass)	The build can take backups of VMs (CR 3.a).
Actual Results	**Duplicati** and **FileZilla (backups)** take and store backups of VMs.
Overall Result	Pass. All requirements for this use case are met.

D.6 Test Case: Data Integrity IP-4

Table 6-6 Test Case ID: Data Integrity IP-4

Parent requirement	(CR 4) The DI example implementation shall identify and protect against malware received via phishing email.
Testable requirement	(CR 4.a, CR 4.b) Denylisting, (CR 4.c) Backups, (CR 4.d) Integrity Baselining
Description	Show that the DI solution can identify phishing emails and protect against configuration changes made by malicious attachments.
Associated Cybersecurity Framework Subcategories	ID.AM-2, ID.AM-3, ID. RA-1, ID.RA-2, ID.RA-5, DE.CM-8, PR.IP-4, PR.DS-1, PR.PT-1
Preconditions	The user receives a phishing email with a malicious attached spreadsheet. The spreadsheet is downloaded and opened, causing account changes in Active Directory.
Procedure	The **integrity monitoring** capability is used to baseline Active Directory activity. This information is forwarded to the **logging** capability, along with other available Active Directory information. The **backups** capability is used to take backups of the Active Directory configuration.

	The malicious web server is added to the **denylisting** capability to prevent downloads.
Expected Results (pass)	The spreadsheet cannot download files (CR 4.a). The build can take backups of configurations (CR 4.c). The build can take and log integrity baselines of configurations (CR 4.d).
Actual Results	**Semperis DSP (integrity monitoring)** successfully baselines Active Directory activity. **ArcSight ESM (logging)** successfully logs activity from Active Directory, including log-ons and changes. When the external web server is added to the denylist, **Cisco WSA (denylisting)** prevents the Excel sheet from downloading malicious files. **Semperis ADFR (backups)** is used to successfully take backups of the Active Directory configuration.
Overall Result	Pass. All requirements for this use case are met.

D.7 Test Case: Data Integrity IP-5

Table 6-7 Test Case ID: Data Integrity IP-5

Parent requirement	(CR 5) The DI example implementation shall identify and protect the database against changes made through a web server vulnerability in custom code.
Testable requirement	(CR 5.c) Backups, (CR 5.d) Integrity Baselining
Description	Show that the DI solution can protect the database against a vulnerability in the custom code of a web server.
Associated Cybersecurity Framework Subcategories	PR.IP-4, PR.DS-1, PR.PT-1, PR.DS-6
Preconditions	A vulnerability in the source code of an intranet webpage is discovered by a malicious insider. The insider exploits this vulnerability to delete significant portions of the database.
Procedure	The **backups** capability is used to take backups of the database.

	The **integrity monitoring** and **logging** capabilities take baselines of the database, for comparison post-modification.
Expected Results (pass)	The build can take backups of the database (CR 5.c). The build can take and log integrity baselines of the database (CR 5.d).
Actual Results	**Duplicati** and **FileZilla (backups)** successfully backs up the database. **Tripwire Enterprise (integrity monitoring)** successfully detects changes in the database. **ArcSight ESM (logging)** successfully logs changes to the database.
Overall Result	Pass. All requirements for this use case are met.

D.8 Test Case: Data Integrity IP-6

Table 6-8 Test Case ID: Data Integrity IP-6

Parent requirement	(CR 6) The DI example implementation shall identify and protect assets against targeted modification by malicious insiders with elevated privileges.
Testable requirement	(CR 6.a) Backups, (CR 6.b) Integrity Baselining, (CR 6.c) Encrypted Backups, (CR 6.d) Secure Storage
Description	Show that the DI solution can protect assets and backups against targeted modification by malicious insiders.
Associated Cybersecurity Framework Subcategories	PR.IP-4, PR.DS-1, PR.PT-1, PR.DS-6
Preconditions	A malicious insider attempts to modify targeted information in both the enterprise systems and the backup systems, using elevated credentials obtained extraneously.
Procedure	The **inventory** capability is used to identify data assets. The **backups** capability provides encrypted backups. **Secure storage** prevents modification or deletion of backups. **Integrity monitoring** and **logging** collect integrity information and baseline the file system.

Expected Results (pass)	The build can take backups of the file system (CR 6.a). The build can take and log integrity baselines of the file system (CR 6.b). Backups are encrypted (CR 6.c). Backups are stored securely and cannot be modified or deleted (CR 6.d).
Actual Results	**Symantec DLP (inventory)** identifies critical data assets across the enterprise. **Duplicati** and **FileZilla (backups)** provide encrypted backups of the file system. **GreenTec WORMdisks (secure storage)** provide write-protection for backups, preventing them from being modified or deleted. **Tripwire Enterprise (integrity monitoring)** and **ArcSight ESM (logging)** baseline critical data assets across the enterprise.
Overall Result	Pass. All requirements of this use case are met.

D.9 Test Case: Data Integrity IP-7

Table 6-9 Test Case ID: Data Integrity IP-7

Parent requirement	(CR 7) The DI example implementation shall identify and protect assets against an intrusion via compromised update server.
Testable requirement	(CR 7.a) Denylisting, (CR 7.b) Backups, (CR 7.c, 7.d) Integrity Baselining
Description	Show that the DI solution can protect against compromised update server as well as intrusion made possible by vulnerable programs.
Associated Cybersecurity Framework Subcategories	ID.RA-1, ID.RA-2, ID.RA-5, DE.CM-8, PR.IP-12, RS.MI-3, PR.IP-4, PR.DS-1, PR.PT-1, PR.DS-6, PR.MA-2
Preconditions	An external update server has been compromised, and a user workstation attempts to update from this server.
Procedure	**Integrity monitoring** capability is used to take baselines of the integrity of both the programs and the file systems. The **backups** capability is used to back up the file system.

	The **denylisting** capability is used to prevent communication between the update server and the machine.
Expected Results (pass)	Machines cannot update from this site while it is denylisted (CR 7.a). The build can take backups of file systems (CR 7.b). The build can take integrity baselines of programs (CR 7.c). The build can take integrity baselines of file systems (CR 7.d).
Actual Results	**Tripwire Enterprise (integrity monitoring)** successfully takes an integrity baseline of both programs and files. **Duplicati** and **FileZilla (backups)** successfully takes backups of the file system. **Cisco WSA (denylisting)** successfully prevents communication between the update server and workstations.
Overall Result	Pass. All requirements for this use case are met.

D.10 Test Case: Data Integrity IP-8

Table 6-10 Test Case ID: Data Integrity IP-8

Parent requirement	(CR 8) The DI example implementation shall identify new and unmaintained assets on the network.
Testable requirement	(CR 8.a) Asset Identification, (CR 8.b) Vulnerability Identification
Description	Show that the DI solution can identify machines new to the network, as well as unpatched machines.
Associated Cybersecurity Framework Subcategories	ID.AM-1, ID.AM-2, ID.RA-1, ID.RA-2, ID.RA-5, DE.CM-8
Preconditions	A new machine with several critical patches missing is connected to the network for the first time.
Procedure	The **inventory** capability is used to identify various aspects about the machine.

	The **policy enforcement** identifies the existence of security solutions on the machine and grants/denies access to the network, based on their presence. The **vulnerability management** capability is used to scan for vulnerabilities on the new machine.
Expected Results (pass)	New machine is identified on the network (CR 8.a). New machine is identified as unmaintained, and required fixes are identified (CR 8.b).
Actual Results	**Cisco ISE (inventory)** successfully logs information about new connections, including the user, date, device, and network information. **Cisco ISE (policy enforcement)** successfully prevents the new machine without 50 security software from connecting to the network. **Tripwire IP360 (vulnerability management)** successfully identifies vulnerabilities on the new machine.
Overall Result	Pass. All requirements for this use case are met.