SANDIA REPORT

SAND2017-3386
Unlimited Release
Printed 07 March 2017

Guide for Cyber Assessment of Industrial Control Systems Field Devices

Jason Stamp, Ph.D., Jennifer Stinebaugh, and Daniel Fay, Ph.D.
Sandia National Laboratories

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from:

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/scitech

Available to the public from:

U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/search

SAND2017-3386
Unlimited Release
Printed 07 March 2017

Guide for Cyber Assessment of Industrial Control Systems Field Devices

Jason Stamp, Ph.D.
Special Cyber Initiatives Department

Jennifer Stinebaugh
Executive Support Division

Daniel Fay, Ph.D.
Assurance Tech and Assessments Department

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1188

Abstract

Programmable logic controllers (PLCs) and other field devices are important components of many weapons platforms, including vehicles, ships, radar systems, etc. Many have significant cyber vulnerabilities that lead to unacceptable risk. Furthermore, common procedures used during Operational Test and Evaluation (OT&E) may unexpectedly lead to unsafe or severe impacts for the field devices or the underlying physical process. This document describes an assessment methodology that addresses vulnerabilities, mitigations, and safe OT&E.

Acknowledgements

The authors would like to acknowledge the funding and technical support from the Office of the Director, Operational Test and Evaluation (DOT&E) for the development of this paper. Also, there were key contributions by other Sandia National Laboratories (SNL) personnel supporting the analysis, particularly from Mitch Martin, Tricia Schulz, Chris Davis, and Nick Pattengale, and from Pacific Northwest National Laboratory (PNNL), especially Chris Bonebrake, Jim Brown, and Katy Bragg.

Executive Summary

Industrial control system (ICS) field devices like programmable logic controllers (PLCs) play a critical role in the safe and reliable operation of Department of Defense (DOD) platforms and weapon systems operations. Unfortunately, these sorts of devices are often rife with cyber security vulnerabilities that can lead to significant risks for mission performance, or even unsafe conditions during routine Operational Test and Evaluation (OT&E). The cyber security issues faced by ICS differ from typical information technology (IT), and this requires a different and more specific approach to assess, test, and mitigate ICS vulnerabilities.

In a typical IT system, data confidentiality and integrity are the primary concerns. In an ICS, mission operations, safety, public health, and avoiding equipment damage are the primary concerns. ICS devices directly control time critical processes and have little margin for delay. Outages or interruptions (even something as simple as a reboot) might not be acceptable, and if unplanned can result in significant risk to mission. Unlike IT system updates or patches, which can be done using automated server-based tools and are widely applicable, ICS updates are specific to the equipment vendor.

OT&E on ICS field devices (on deployed platforms, or in high value test rigs) is often a necessary requirement, but this causes significant concern within the DOD ICS community. The concern is that implementing routine cyber security measures and testing on active ICS components and systems may damage the ICS or even underlying physical systems. Of particular concern are ICS field devices, which encompasses the specialized hardware that covers the boundary between the cyber and physical domains. Examples of field devices include PLCs, electric power relays, remote terminal units (RTUs), and other embedded devices.

The goals of the Field Device Assessment Methodology (FDAM) are to research and rank field device vulnerabilities to be tested, summarize associated mitigations, and determine cyber test concerns by summarizing potential OT&E test damage/safety issues. The FDAM primarily supports the cooperative assessment stage of OT&E, although the results can also support adversarial assessments.

This document provides guidance on tools and procedures that have been developed by Sandia National Laboratories (SNL) that are used to implement the FDAM approach, including an assessment framework, quantitative risk calculation, and ranked access/procedure pairs (APPs); these are described in Chapters 2 through 4. The FDAM process itself is presented in Chapters 5 through 7 – from initial research and discovery, to standalone lab testing, through to compiling the final report.

As cyber security testing is inherently complex and detail-oriented, personnel executing the FDAM will generally differing knowledge and experience, and this is difficult to fully document or simplify into a step by step process. Different testing situations will influence the testing process and its execution; therefore, it is not necessary to follow every step in the document as laid out.

The FDAM is intended to support operational test agencies (OTAs), cyber protection teams (CPTs), and other organizations within DOD that support OT&E on weapons platforms and systems, but it can also be applied to ICS used within DOD installations and other bases, particularly for infrastructure support. The Director, Operational Test and Evaluation (DOT&E) FDAM is applicable for mission platforms, which are heavily reliant on ICS, including naval shipboard systems (electrical plant management, machinery control, aircraft launch/recovery, radar, fire control, and others), advanced ground vehicle management, and aircraft/avionics. The FDAM also supports a range of DOD assessment requirements [1, 2] and the approach is suitable to varying classification levels, as application details and close-held government information can be included when desirable (and useful).

Contents

Appendices

List of Tables

Abbreviations and Acronyms

A availability (for **A** by itself)

A access (for **A** followed by a number)

APP access/procedure pair

C confidentiality

CANBUS controller area network bus

CP communications processor

CPT cyber protection team

CPU central processing unit

CVE common vulnerabilities and exposures

CVSS Common Vulnerability Scoring System

DOD Department of Defense

DOS denial of service

DOT&E Director, Operational Test and Evaluation

DRAM dynamic random-access memory

EEPROM electrically erasable programmable read-only memory

EM electromagnetic

EW electronic warfare

FDAM Field Device Assessment Methodology

FTP file transfer protocol

G goal (for **G** followed by a number)

HART highway addressable remote transducer

I integrity

I/O input/output

ICS industrial control system

ICS-CERT Industrial Control Systems Cyber Emergency Response Team

IT information technology

JTAG Joint Test Action Group

LED light emitting diode

LOE level of effort

M mitigation (for **M** followed by a number)

MMC MultiMediaCard

NIST National Institute of Standards and Technology

OPFOR opposing force

OS operating system

OSD Office of the Secretary of Defense

OT operational technology

OTA operational test agency

OT&E Operational Test and Evaluation

P procedure (for **P** followed by a number)

PCB printed circuit board

PCMCIA Personal Computer Memory Card International Association

PHY physical transceiver

PLC programmable logic controller

PNNL Pacific Northwest National Laboratory

RBAC role-based access control

RE reverse engineering

RF radio frequency

ROE rules of engagement

RTU remote terminal unit

SCADA supervisory control and data acquisition

SDRAM synchronous dynamic random-access memory

SERDES serializer/deserializer

SNL Sandia National Laboratories

SoC system-on-a-chip

SRAM static random-access memory

SSH secure shell

TCP Transmission Control Protocol

TCP/IP Transmission Control Protocol/Internet Protocol

US United States

US-CERT United States Computer Emergency Readiness Team

USB universal serial bus

V vulnerability (for **V** followed by a number)

VME Versa Module Europa

VPN virtual private network

V/C verification/characterization

WAN wide area network

Chapter 1

Introduction

This document provides an overview of the Field Device Assessment Methodology (FDAM). The FDAM approach encapsulates the experience that has been developed over many years by analysts at Sandia National Laboratories (SNL) to provide a safe, economical, and efficient approach that can be used to analyze and test potential mitigations to industrial control system (ICS) cyber security vulnerabilities, both in laboratory testing and later in operational settings.

Over the last 20 years there has been increasing recognition of cyber vulnerabilities in ICS, starting with concerns relating to United States (US) critical infrastructure. Initially, ICS vendors developed highly specialized technology that had a severely constrained operating envelope (very low bitrate networks, analog signaling, and minimal software). However, the ICS industry has increasingly adopted conventional information technology (IT) capabilities and approaches; for example, ICS control center software is installed on conventional (often Windows-based) computers, and the ubiquitous Transmission Control Protocol/Internet Protocol (TCP/IP) protocol stack now supports a lot of ICS networking. The adoption of IT technologies for increased ICS capabilities has led to security problems since ICS and IT systems are designed and implemented differently. Technologies borrowed from IT systems sometimes include their own vulnerabilities; in an IT environment, regular patching is expected and provides mitigation, but ICS owners are often reluctant to change a fielded system and therefore rarely apply patches. This leads to many ICS having known vulnerabilities that are not fixed once they become known, which stresses the importance of operational testing for systems or platforms that utilize ICS.

A second challenge is the lack of a strong cyber focus on the part of the ICS vendor and integrator communities. Since operational availability is the key motivating factor, decisions favor reliability over security. For example, in the design stage, hardcoded vendor backdoor passwords may facilitate troubleshooting (although awareness of this as a problem is improving) and ICS software packages run with highest privileges to avoid any potential problems caused by incorrect permissions. Some ICS networking protocols are extraordinarily permissive, up to allowing direct memory access to facilitate simple, reliable control and troubleshooting. Moreover, ICS networks rarely include network defense-in-depth, because this can introduce additional communication delays and add more potential points of failure. Furthermore, ICS component designs often assume a rigidly bounded set of operational conditions that are expected for any deployment. As a result, implementations of key services on ICS devices may not support reasonable error checking and capture. Unexpected IT network traffic (like port/device scans, broadcast traffic, or malformed packets) could lead to interruptions or crashes. Potential device fragility limits the techniques available to monitor and secure ICS.

According to an Office of the Secretary of Defense (OSD) memorandum regarding "Procedures for Operational Test and Evaluation of Cybersecurity in Acquisition Programs," operational test agencies (OTAs) will "include cyber threats... with the same rigor as other threats" [3]. The purpose of cyber security operational test and evaluation is to evaluate the ability of a unit equipped with a system to support assigned missions in the expected environment. The "system" in this case is considered to encompass hardware, software, user operators, etc. This memorandum also specifies the procedures to be used for testing oversight systems. The purpose of this document is to introduce a FDAM that parallels (with some differences due to the focus on ICS hardware and not the entire system) the procedures suggested in the memorandum. The FDAM approach is not intended to cover the entire oversight system as referenced in the memorandum; rather, it explains the procedures necessary to evaluate the ICS hardware devices. This focused approach on the hardware subset of the system is warranted because ICS field devices face very different issues than IT systems, and the risks associated with ICS cyber vulnerabilities can be significant.

Due to these issues, there is significant concern within the Department of Defense (DOD) ICS community that routine cyber security measures that actively engage components and systems may prove damaging to the ICS or even underlying physical systems when testing is performed on operational systems or high-value test rigs. The effects that are of concern include:

- Unexpected or unpredictable physical system behavior
- Damage to physical system components
- Denial of service (DOS) for local automated or manual control
- DOS to supervisory control
- Unrecoverable (or nearly so) faults for field devices

The FDAM has been developed to summarize vulnerabilities, prioritize risk, evaluate mitigation measures, and develop measures for safe Operational Test and Evaluation (OT&E). The FDAM manages the above effects of concern by developing insight and limitations for field device cyber security issues in a lab setting before performing tests or implementing cyber security mitigations on operational devices.

1.1 Overview of the FDAM

A high level overview of the FDAM approach is listed below, with associated chapter numbers where applicable:

- Literature review and device physical examination (Chapter 5)
- Verification/characterization (V/C) testing (Chapter 6)
- Exploration testing (Chapter 6)
- FDAM summary (Chapter 7)
  - Conclusions
  - Recommendations

The FDAM is a process-oriented encapsulation of experience from decades of work by analysts at SNL. The methodology should provide a safe, economical, and efficient approach to analyze and test potential mitigations to ICS field device vulnerabilities, both in laboratory testing and operational settings.

It should be noted that cyber security is inherently complex and detail-oriented. The people that perform testing and analysis often have a wealth of knowledge and experience that in some cases cannot be documented or simplified into a step-by-step process. The FDAM tries to be as thorough as possible, but it should be noted that in every testing situation, the experience and knowledge of the testers may influence part or all of the process. For example, a tester that is intimately familiar with supervisory control and data acquisition (SCADA) systems and has had years of experience in the ICS management field might choose to do literature review and vulnerability scoring concurrently with lab testing. A person with limited experience might take a more conservative, methodical approach and follow the procedures presented here to the letter. The goal of the approach is to help testers optimize assessment scoping in order to ensure efficiency, safety, and highest return on testing approaches.

Several conceptual frameworks and tools have been developed that are essential to understanding the FDAM approach. These are described prior to the actual procedures in order to to provide a necessary foundation and ground subsequent discussions. These tools, described in Chapters 2-4 include:

- Cyber risk framework
- Prioritized access/procedure pairs
- Vulnerability risk calculation

The FDAM activities are organized according to a cyber risk framework, comprised of five categories (Chapter 2). Each of these five category consists of several components, which are used to prioritize the cyber analysis and testing:

- Goals – what is the adversary trying to achieve
- Procedures – what the adversary is doing during cyber attack
- Access – the avenue to the device environment
- Vulnerabilities - includes both device and system vulnerabilities
- Mitigations - cyber and physical, for defend/detect/manage

A key concept of the FDAM approach is the introduction of prioritized APPs developed to ensure optimal testing scenarios are chosen (refer to Chapter 3). These APPs, combined with the risk ranking calculation, allow the tester to optimally select the vulnerabilities to be tested by narrowing down the scope of the testing landscape, ensuring the highest degree of improved cyber security without compromising ICS performance or safety. Each APP also includes examples of tests and potential (or likely) vulnerabilities.

The vulnerability risk calculation has been transposed from an IT-based Common Vulnerability Scoring System (CVSS) system to be more applicable to ICS systems (Chapter 4). This calculation assesses the risk associated with vulnerabilities uncovered by the literature review and research, and is used as a guideline (in conjunction with the APPs described above)to optimally select what is to be tested. If a particular vulnerability has a low risk score, but could be very difficult to test or if mitigation is onerous, it might be removed from the assessment. Conversely, if a vulnerability carries with it a high risk score, and the mitigation approach is simple, it should strongly be considered for further testing. An example of this selection process is given in Appendix A.

1.2 Applicability of the FDAM

While high capability threats are within scope (up to and including advanced adversaries), many ICS implementations are vulnerable to much lower-capability adversaries and techniques. Simpler vulnerabilities that require lower level of effort (LOE) are of higher concern, as are ones that have a greater effect on device operations (which could affect the underlying physical process). The degree of difficulty for detection or prevention of the vulnerability is also a factor. Furthermore, to best analyze ICS cyber security, some level of opportunity (in the form of adversary access and capability) must be assumed. Therefore, access denial for ICS field devices will attenuate risks and complement mitigations described in a completed FDAM assessment report. Assuming reasonable access is also most attuned for understanding operational test safety concerns.

The DOD has a mature and proven approach for cyber security [4, 5] for networking and higher level system support environments. The FDAM approach instead focuses on mitigating risks specifically for fielded ICS devices, which in general have a less mature security framework given the motivation and necessity to keep the devices running. Thus, the FDAM approach ensures that mitigation efforts considered will:

1. Preserve system operation (to best support mission performance)
2. Prevent damage or degradation to associated physical systems
3. Deny attractive information from an ICS to an adversary

In general, the FDAM approach should be amenable to different deployment situations, including:

1. A given component in a specific application (e.g. a water pump control system)
2. A given component in a typical application (e.g. machinery control)
3. A given component in any application

The application may help characterize the technical configuration of the component or ICS, although some variability is always expected given refresh cycles for weapons platforms. Of course, less specificity may enable more generally useful conclusions; however, as specificity increases there is more opportunity to understand vulnerabilities and mitigations (as an example, possible physical domain mitigation steps that guard the controlled process against malfunctioning PLC). In the Director, Operational Test and Evaluation (DOT&E) process for understanding cyber risk, the results from analyses derived from this methodology will inform future structured cyber assessments [3] by providing an important foundational characterization.

The FDAM is focused on mission platforms, which are heavily reliant on ICS. Examples include many naval shipboard systems (electrical plant management, machinery control, aircraft launch/recovery, radar, fire control, and others), advanced ground vehicle management (engine control/optimization, CANBUS-based networks, etc.), and aircraft/avionics (often based on VME and 1553 bus). Although the FDAM is intended to support OTAs, cyber protection teams (CPTs), and other organizations within DOD that support OT&E on weapons platforms and systems, but it can also be applied to ICS used within DOD installations and other bases, particularly for infrastructure support (like electric power, water, physical security,lighting systems for flight lines, radar, water treatment, access denial systems, heating/cooling plants, etc).

1.3 Intended FDAM Results

The FDAM supports a range of DOD assessment requirements [1, 2] and the approach is suitable to varying classification levels, as application details and close-held government information can be included when desirable (and useful). The information gathered by using the FDAM approach should be put into a final report (see Chapter 7) that summarizes the vulnerabilities and mitigations that are to be lab tested (see Appendix A), results of the testing, and recommendations regarding potential OT&E test damage/safety concerns and whether potential mitigations should be implemented on an operational systems.

Chapter 2

Framework for Analyzing ICS Field Devices

The FDAM is based on a generalized framework which has been developed to capture the cyber security elements necessary to successfully use the approach. There are five categories for consideration – adversary goals (G), procedures for cyber attack (P); access to the device environment (A); device and system vulnerabilities (V); and cyber/physical mitigations (M). These categories, with the subcomponents of each, are listed below:

- Adversary Goals (G)
  - G1: Change behavior
  - G2; Misrepresent state
  - G3: Obtain or change data
  - G4: Deny operation
  - G5: Use as intermediate attack platform
- Procedures for Cyber Attack (P)
  - P1: Achieve device management access
  - P2: Gain privileged operational relationship
  - P3: Impersonate the device
  - P4: Affect device operation via atypical stimuli
- Access to the Device Environment (A)
  - A1: Physical
  - A2: Network
  - A3: Secondary
  - A4: Peer device
  - A5: Engineering workstation
  - A6: Supply chain

- Device and System Vulnerabilities (V)
  - V1: Common problems/networking and communications
  - V2: Common problems/architecture and management
  - V3: Equipment-specific problems
  - V4: Context- or application-specific problems
- Cyber and Physical Mitigations (M)
  - M1: Defend – take actions that prevent unwanted access
  - M2: Detect – monitor for unauthorized system access attempts
  - M3: Manage – react appropriately to incidents and ensure successful recovery

Goals are achieved by procedures, which in turn are made possible via access, and involve the exploitation of vulnerabilities to be successful. Vulnerabilities are best understood in the context of an adversaries' goals or intent, which may be specific or general to a desired scenario – this is the threat. Cyber risks can be denied or attenuated by proper mitigation strategies (which could encompass safeguards that limit the risk – for example, a purely mechanical pressure relief valve would not allow an embedded device to catastrophically over-pressurize a pipe).

Each of the five categories is described in detail below. Note that in the access and procedure categories, the descriptions include a notation that describes its prioritization within the FDAM (the overall framework itself is general and applicable to many assessment situations, and so the notation is necessary to describe its application for this work). This prioritization is key part of the FDAM testing approach, and will be described in detail in Chapter 3 as prioritized APPs.

2.1 Adversary Goals

Goals for cyber attacks against embedded devices in ICS generally fall into five categories. An adversary will seek one or more goals to achieve a desired effect in the underlying physical architecture (weapons systems, electric supply, machinery control, oil/gas pipelines, manufacturing, building automation, etc.).

- **Goal G1 – Change behavior:**

  Changing the control settings or logic can lead to behavior that is advantageous to an adversary. In extreme cases, this could cause or contribute to equipment damage, dangerous hazards, or systems degradation/failure.

- **Goal G2 – Misrepresent state:**

  Embedded devices form the boundary between control and the physical process. Vulnerabilities might facilitate the device misrepresenting the process state (or its own), either directly (the device itself is misrepresenting) or indirectly (the device can be easily impersonated).

- **Goal G3 – Obtain or change data:**

  Any embedded device may be more or less amenable to reporting potentially sensitive data about itself or the underlying process to unauthorized actors, or toggling control based on correctly-formatted protocol directions.

- **Goal G4 – Deny operation:**

  Embedded devices are almost always expected to have very high uptime, particularly to manage potentially dangerous physical processes. Therefore, a short-term loss of functionality – even partial – could be important. Straightforward vulnerabilities can permanently disable some devices. Note that the denied operations are being analyzed from the perspective of the field device itself; therefore, denial of communications external to the device would be an example of a procedure that would be out-of-scope.

- **Goal G5 – Use as intermediate attack platform:**

  An adversary can theoretically use any computing device as a network or systems beachhead for further malevolent activities. This will usually take the form of a high-privilege interactive session for the adversary, with the download and hosting of additional software tools desirable. Very often, the architecture of embedded devices makes this goal unlikely, but it remains reasonable for some scenarios – particularly if the embedded device is leveraging more conventional hardware/software or has interesting connectivity.

Individual analysis for the targeted equipment will determine the likelihood of its susceptibility to these goals. Risk will depend on the chosen attack procedures, the access available to an adversary, and vulnerabilities/mitigations for the equipment.

2.2 Procedures for Cyber Attack

An adversary has several generalized operations that can support realization of their goals. Procedures can be thought of as the net results of overall scenarios that adversarial manipulation of the targeted equipment (a series of attack steps). For this category and the next (“access”), there is a descriptor in each category that describes the analysis priority for a FDAM assessment (“for the purposes of FDAM, this is listed as Priority #X”). This prioritization is key part of the testing approach, and will be used as a foundation for the discussions in Chapter 3.

- **Procedure P1: Achieve device management access**

  Embedded devices can be thought of as running one or more software processes subject to configuration parameters. In most devices, both software and parameters are configurable and reprogrammable remotely, and nearly all support modification via the device's serial port or local interface. Typical configuration parameters include management-specific information, like networking addresses, accounts (to the extent that the device supports them), logging frequency and parameters, etc. Other configuration data affects the equipment's control and measurement of physical data, like control setpoints, measurement intervals and scaling, ladder logic, etc. Device firmware encompasses all software outside of configuration, including boot loaders, operating systems, software, and other data. Note that, in a modular device, each component can contain configuration and firmware. Modifying the device's behavior could allow an adversary to achieve any goal (although G5 might be difficult if malware or tools are not available for the targeted equipment's architecture). The device hardware may also be changed to suit an adversary. Given the poor state of forensics for many existing embedded systems, P1-type scenarios could easily lend themselves to minimal probability of detection or attribution and because embedded devices have long operational deployments, adversary-driven changes in configuration could have a long-term presence.

  For the purposes of FDAM, procedure P1 is included as ***Priority #2****. This clearly has the potential for extreme impacts to ICS devices and underlying physical systems.*

- **Procedure P2: Gain privileged operational relationship**

  Embedded devices often have weak or nonexistent capabilities to authenticate their peers in operational relationships. As an example, a PLC running the Modbus protocol will reply with current data to any well-formed data request that accesses a configured data register, and would toggle process control similarly. Therefore, depending on the situation and access, an adversary could easily gain valuable data and potentially affect processes (G3) by injecting control data. This might involve accessing existing peer devices, adding customized hardware to the network, or simply injecting network traffic. The ability to view device traffic on a network could allow for unauthorized access to data being sent to or from an embedded device (depending on the protocols and safeguards used), which is a type of privileged relationship.

  For the purposes of FDAM, procedure P2 is included as ***Priority #3****. The unsecured nature of automation protocols used by ICS devices lead to the likelihood than unauthorized parties may issue false but authentic-looking requests for data or control, to whatever extent allowed by the device configuration. This is of lower priority for additional analysis because of its straightforward nature, and would only be investigated in depth if there were some unique risks or technical details (although it clearly can have significant effects on the ICS and underlying physical systems).*

- **Procedure P3: Impersonate the device**

  The scenario described in P2 above also extends to the possibility of an attacker providing or consuming data in place of a targeted piece of equipment. The authentic data messaging for the targeted device could be blocked, changed, or ignored depending on the adversary's needs.

  *For the purposes of FDAM, procedure P3 is included as **Priority #4**. Impersonating the ICS device to peer devices or supervisory control is also straightforward, similar in nature to gaining a privileged operational relationship (P2). Therefore, this is also a relatively low priority compared to other procedures.*

- **Procedure P4: Affect device operation via atypical stimuli**

  This type of procedure covers tactics not previously mentioned. The critical distinction for this procedure is that all "atypical stimuli" are outside of the on the intended administration or operations of the field device once deployed. Examples would include DOS attacks, susceptibility to radio frequency (RF) that affects operation or management, flaws in protocol or server implementations that could crash the device, adding unauthorized hardware to the system, leveraging debugging capabilities left over from the device development, etc. Another example: affecting the device programming using the vendor's configuration software to achieve goal G1 is an example of procedure P1, but leveraging a flaw in the device's network stack to overwrite firmware via a network connection is within procedure P4.

  *For the purposes of FDAM, procedure P4 is included as **Priority #1**. Although achieving device management access conventionally (P1) has more degrees of mapping with respect to goals, affecting device operation via atypical stimuli (P4) is a slightly higher priority for investigation because of its associations with inadvertent effects to the device (both changing behavior and denying operation – G1 and G4 respectively). Potential fragility or mistakes in ICS network interface design, implementation, or configuration may lead to significant impacts to the device from minimal stimuli, like aggressive networks scans or small sequences of network traffic from attackers. Again, atypical stimuli includes any attacks outside of the normal operational envelope or management via vendor-approved approaches.*

2.3 Access to the Device Environment

Access allows the potential execution of steps in one or more selected procedures. Access represents a fundamental requirement for any planned procedure and, if unavailable, would negate the feasibility of even attempting some procedures. However, access can be gained in many different ways over diverse timeframes. As with the "Procedures" section, there is a descriptor describing the priority of the access for the FDAM analysis. This prioritization will support developments in Chapter 3.

- **Access A1 – Physical**

  Physical access refers to an adversary having the opportunity to touch the device. This affords opportunities to change hardware or access non-networked communication ports. The latter may be particularly sensitive. Many embedded devices include two primary types of communications/networking interfaces; the first is for interacting with other system operational elements (such as servers in a control center or other embedded devices) for data exchange and control, and the second is often a high-privilege connection used for device configuration. (Often the functions of the second are replicated on the first for convenience, and in some cases there is only a single interface for configuration and operations.) However, if there is a dedicated configuration interface, then this represents a potentially attractive opportunity for an adversary if it can be physically accessed, because it typically has the highest privilege levels available. If there are authentication challenges, these can be often defeated by jumpers or other physical interfaces on the equipment. An adversary could also leverage physical access to introduce additional equipment, either in the local network, on a configuration interface, within the embedded device itself, or added to a modular chassis (like many PLCs have). These can provide varying opportunities for procedures with low potential for detection (depending on the site's configuration control and monitoring scheme). Physical access can also easily affect device configuration if the firmware is stored on removable media. Another particularly insidious type of access would be to achieve net physical access using social engineering vectors (like the classic case of the USB drive in the parking lot). Similarly, an adversary could game operators to manipulate ICS equipment to affect a number of procedures and achieve goals. Many physical access avenues could be very persistent and stealthy.

  *For the purposes of FDAM, access A1 is listed as **Priority #2**. Many severe attacks are possible given this level of access. The focus will be on achieving device management access (P1) and causing atypical stimuli (P4); privileged relationships (P2) and impersonating the device (P3) are not considered.*

- **Access A2 – Network**

  Embedded devices are very often networked via their operational interfaces to participate in control with peer or remote systems. Networks can be based on legacy serial technology or modern systems like Ethernet and TCP/IP. Local access would be network or communications access to the equipment with physical proximity. Typically, this will involve less monitoring given the potential for physical access controls that are usually presumed to be effective. Remote network access would be from physically distant systems over a permanent data connection, like through an ICS wide area network (WAN) or the cloud, and usually includes greater monitoring and lower bandwidth than local.

  *For the purposes of FDAM, access A2 is listed as **Priority #1**. The network avenue has large force multiplier for adversaries and enables stealthy remote attack. Also, the DOD is concerned about potential CPT or opposing force (OPFOR) network auditing and testing which could apply significant cyber stress to ICSs devices. Depending on configurations, all procedures are likely feasible; however, gaining a privileged operational relationship (P2) and impersonating the device (P3) have been regularly demonstrated and are of lower priority barring some key differentiating factors for the equipment or situations under test.*

- **Access A3 – Secondary communication**

  Permanent or periodic remote access can be intended for the equipment-owning entity to gather data or manage configuration, or could be used via trusted partners like outsourced technical management or vendors (many of which often require highly privileged access to their equipment). This is separate from continuous network access (A2), in that if both A2 and A3 are present, they are on different networks with different technology and distinct opportunities for adversaries. Often the secondary access will apply to a device's dedicated configuration (physical) port.

  *For the purposes of FDAM, this access avenue **not in scope**. The FDAM is focused on devices, and data about deployment patterns (which would describe secondary access capabilities) is likely absent. Therefore, it is unclear what opportunities there would be for remote network access to device configuration ports. Subsequent analysis can consider relevant analysis using physical access (A1) to inform risk from A3.*

- **Access A4 – Peer device**

  An adversary can access the functionality on a targeted device by successfully gaining privileges on a peer device, and then exploit the existing trusted data relationships that support the operations of the operational technology (OT) or ICS system. This would be a straightforward way to execute scenarios that depend on P2 (privileged operational relationship). This may or may not allow for additional attack opportunities compared to A2 (network access).

  *For the purposes of FDAM, access A4 is listed as **priority #4**. Since the analyst often does not have comprehensive insight into deployment patterns, and the focus is on device-level concerns, this access approach is included only as they relate to designated remote input/output (I/O) or coordinated control architectures for the field device under test. The best approach for analysis will be to assume physical access to the peer devices or slave I/O (but not to the master field device itself); this assumption may be reconsidered in light of device usage and environment if that information is available.*

- **Access A5 – Engineering workstation**

  As noted in the "Procedures" discussion, unauthorized device management is a key problem for embedded systems. One extremely attractive access opportunity would therefore be via the platform that is normally used to manage and configure the targeted equipment (typically called the engineering workstation – one or more computing platforms used by technicians for authorized configuration control of embedded systems). Either the management software itself or other hardware/software elements of the platform could be targeted. This is a type of lifecycle access for the embedded device given the engineering workstation's key role over the deployment history for embedded systems.

  *For the purposes of FDAM, access A5 is listed as **priority #3**. This vector applies only to gaining configuration access (P1), and is concerned with possible effects on the PLC given attacks against the authorized copy of the device management software, assuming they are connected at some point.*

- **Access A6 – Supply chain**

  Another interesting avenue for access is to consider the possibility that an adversary has affected the embedded devices' hardware or software as it was being developed or sourced. Trojan code or modified silicon may require very high investments to achieve, but for certain classes of adversaries the possibilities are reasonable. Another avenue for access in this category involves an adversary affecting the supply chain for product updates, particularly for firmware. Although OT and ICS devices are rarely updated or patched by current practice, there appears to be more acceptance coming for field updates to embedded devices. Under those conditions, Trojan code may be introduced into systems for later use. Also, many vendors retain privileged access via presumed private communications as a condition of their equipment sale, and those channels present attractive targets for adversaries.

  *For the purposes of FDAM, this is **not in scope**, given all of the other vulnerabilities for ICS devices. However, it is relevant for an advanced threat and should be considered separately as needed. Also, the scope of these assessments does not include information regarding intended vendor patch plans for DOD platforms (for either the device or its configuration software), which is an important consideration for this access approach.*

2.4 Device and System Vulnerabilities

For the planned analysis, consideration for vulnerabilities is needed to understand the potential risks to operations from adversary access to embedded systems. Often, embedded systems – particularly for ICS – are susceptible to classes of common vulnerabilities based on the technology space and the particular application. These common problems are investigated for the equipment being analyzed within the categories shown. Other vulnerabilities may be specific to proprietary design characteristics or implementation flaws, and those can be subcategorized into "known" (publicly available information) or "close-held" (known only to small audiences, like the manufacturer, user groups, or government). Lastly, specific information about the device within the target context may illuminate important vulnerabilities.

- **Vulnerabilities V1 – Common problems/networking and communications**

  Vulnerabilities in this category relate to services that are supported by the embedded equipment that are necessary to its function within common ICS or OT environments. Due to legacy development focus solely on availability, protocols used by embedded devices lack security services, and this is not a fault particular to any device.

- **Vulnerabilities V2 – Common problems/architecture and management**

  These types of vulnerabilities relate to the devices themselves, and the way that they are built and operated according to expectations within an established industry. An example for this category would be the lack of account-based access; although this would be an attractive security feature for role-based access control (RBAC), it is simply not a value-added feature within the ICS market.

- **Vulnerabilities V3 – Equipment-specific problems**

  This category of vulnerabilities includes those from the design or implementation of the device or equipment. There are two subcategories: well-known problems (information is available publicly) and closely-held problems (known only to limited groups or within the government). There are three primary sources for known vulnerabilities: information from the manufacturer, vulnerability reporting authorities like Industrial Control Systems Cyber Emergency Response Team (ICS-CERT), and relevant user communities including ones within government. While equipment-specific close-held problems may range from simplistic to extraordinarily sensitive, they are enormously useful information for this sort of cyber analysis.

- **Vulnerabilities V4 – Context- or application-specific problems**

  Finally, the device vulnerabilities can be highlighted by considering the actual application or deployment environment. This would be particularly useful to better understand the discussion of common vulnerabilities, and the existence of specific vulnerabilities within the specific context (i.e. a known software patch level would clearly settle any issue of specific vulnerabilities being present).

2.5 Cyber Risk Mitigation

Mitigations are a key focus for FDAM. However, they are often less emphasized during device analysis than vulnerability discovery. Therefore, their introduction is somewhat more detailed than the prior framework elements. As described previously, mitigations in the FDAM framework are categorized as follows:

- M1: Defend vulnerabilities (reduce exposure)
- M2: Detect attacks (monitor and alarm)
- M3: Manage incidents (attenuate impacts and recover)

It is important to note that in *for the purposes of this analysis, mitigations do not refer to access denial, which is a separate category of countermeasures, and an area where DOD already has numerous policies and procedures*. Access denial will certainly help to manage risk. Mitigations might be commonly applied either to manage common vulnerabilities via cyber controls (an example of this might be configuration monitoring) or to guard against physical effects via physical controls (like pressure relief valves or fuses, which are not attackable via cyber means but could lead to extended equipment outages). Mitigations may be specific to an ICS or a particular application or context.

Some mitigations, particularly those in M3, can be best analyzed by considering the actual deployment environment. An example could be an electrical fuse in a power system, where an embedded device (a relay) has the primary responsibility for detecting and clearing short circuits, but in the event of relay failure the fuse will blow. Preferably, the short circuit is managed by the embedded device, because it allows for easier restoration of service (perhaps recloser control for a transient issue) compared to needing to dispatch a crew to replace the fuse. However, the potential overall effect is greatly reduced by adding a fuse, instead of an adversary compelling relays to completely ignore a fault and leading to possible damage and safety impact.

2.5.1 M1 – Defend

Vulnerabilities can be defended by deploying compensatory measures: configuration access should be logged and monitored by additional overhead software; firewalls can be used to create ICS enclaves and defend against abnormal networking traffic; unauthenticated open protocols for communication and substandard configuration authentication can be protected; and jumpers to reset passwords can be physically protected. However, the reality of ICS are that they are designed for reliability, and to date poor security is not seen as a force that could reduce reliability. Unfortunately, this makes the addition of security devices that add possible failures an unlikely option, although the balance between greater numbers of failures caused by non-adversarial stimuli versus the actual reliability of an unprotected ICS should be considered.

The nature of ICS supports a whitelisting approach, which starts with effective characterization of allowed network communications. There are two major options to consider when considering defensive mitigation strategy:

- Blocking unallowable types of activity (perhaps based on network protocols)
- Blocking unallowable types of activity and also unreasonable settings or data within allowed activities

The former is relatively straightforward to implement, while the latter (a more behavioral approach) is more challenging, but provides better protection. Consider a system using Modbus/TCP for communications. There may be attacks that attempt to change setpoints outside allowable system bounds (which can effect the capabilities of the physical process being controlled, or might be associated with expectations based on current operations). R&D efforts suggest that machine learning techniques for behavioral monitoring are feasible and useful (although false positives would be a threat to impact legitimate operation).

Alternatively, cyber vulnerabilities can be left in place, with some potential issues managed by physical devices. Locked cabinets or physical barriers can help protect jumper settings and removable memory from tampering and limit RF or EM exposure. Unfortunately, strong physical protections may be compromised by poorly considered convenience features. An example would be a PLC locked within a cabinet, but with the device's serial configuration port extended outside of the enclosure via a cable.

Depending on the specific ICS device, there might be both well-known and closely-held mitigations, although mitigation steps are often less sensitive than vulnerabilities. Equipment-specific mitigations could be cyber (using vendor-specific security services) or physical (using key-based run/program selectors to limit device reconfiguration opportunities – assuming they cannot be bypassed via software, and this is often the case). Another vendor specific mitigation is to ensure that all patching and updates are in place for specific equipment. However, unlike the patching / update approach used in IT, this might not happen in ICS systems due to downtime associated with implementation and testing/recertification necessary to bring the system back to operational use. However, a patching / update approach is vital to ICS security, and should be strongly considered, (as long as business/warranty issues are adequately addressed). Careful planning (and possible testing) can ensure patching is done with minimal effects on the system.

Other opportunities for defense include leveraging vendor-supplied security features to the greatest extent allowed by the device's application. Examples include:

- Configuration authentication procedures (which often have significant weaknesses)
- Network encryption or authentication (granted, this is unlikely to be supported by ICS devices)
- Setting read-only permissions (possibly using physical settings, although these can sometimes be circumvented via cyber attack)

2.5.2 M2 – Detect

Given the reluctance of ICS stakeholders for potentially degrading reliability by inserting components to defend systems, attack/incident *detection is the key opportunity to reduce cyber risk*. Detection is especially attractive if it is accomplished by sensors that unobtrusively monitor signals without adding security equipment in-line with systems and operational data flows.

Monitoring can be categorized into the same two major options from the defense discussion. Clearly, any communications protocols that are outside of the narrowly defined allowed set are evidence of misoperation and potentially a cyber intrusion. However, the behavioral model allows stronger detection. As an example, consider network traffic associated with remote configuration of a PLC. Although this may be necessary to the proper functioning and administration of a system, a traffic signature showing an active configuration session that is outside normal maintenance intervals would be cause for concern. Monitoring for reasonable setpoints along the lines of the M1 discussion for defense would also be very useful.

Because of their recognized importance, ICS field devices may include some state-of-health monitoring that could detect cyber attack. Also, tracking device configuration is an excellent defense, particularly given the small amounts of memory in typical field devices. For the majority of devices, very little (if any) change for most of the memory space (except operational data registers) is expected when the device is operating normally. However, the great range of variation in the architecture of field devices will require customization for any configuration monitoring countermeasures.

Another opportunity is independent cyber monitoring within the device. If the component is modular (with a backplane) or has reasonably accessible signal buses, a monitoring device listen to the conversations between the component's constituent modules. Such devices are under active development [6]. Because of the granularity in communications, a backplane/bus monitor can uniquely observe behavior of the control system independent of the processor and alert when the system is not operating within a specifically defined manner defined by whitelisting and behavioral heuristics. Because it alerts on effects of an attack in progress, and not on signatures of prior attacks, countermeasures like backplane or bus monitoring can detect zero-day exploits [7].

The concepts that support the backplane/bus monitoring approach also can be applied on the physical process side of the ICS field device. This may be realized using a second, independent system to monitor the protected device's output signals to ensure they are suitable for the process and context. However, this is far more resource intensive than a backplane/bus monitor approach, which itself is only one step behind the analog I/O equipment within the device.

Physical security monitoring is complementary to defense measures. Countermeasures like cabinet and access door alarms are almost always good suggestions for mitigation. In contrast with IT systems, early R&D results for ICS suggest that monitoring power draw for components (a physical signature) may be a reliable indicator of off-nominal operation that potentially indicates an attack) [8].

2.5.3 M3 – Manage

Cyber attacks against ICS can cause damage to the physical processes being monitored and controlled. Defense and detection based mitigations (M1 and M2) that prevent such damage are preferred but may be infeasible or unreasonable, and so mitigations intended to manage incidents often play an important and necessary role. Mitigations in this category can minimize impact to the physical processes and aid in efficient and safe device recovery in order to restore both local and supervisory functionality while preserving needed forensic indicators. Device recovery can occur over varying time frames and require varying levels of administrative intervention (in ascending order of impact to ICS operations):

- Automatic: the device restores itself to operational status after an error condition is detected
- Remote: a qualified admin can restore the device
- Local: qualified admins and/or technicians require physical access to repair/restore the device

Given these parameters, hierarchies of recovery can be described (again, in ascending order of impact to operations):

- Automatic self-recovery with short, acceptable delay
- Restore remotely with quick procedure (e.g. toggle a setting)
- Restore remotely with involved procedure (e.g. reload configuration/firmware)
- Restore locally with quick procedure (e.g. rotate key without reset, power toggle)
- Restore locally with involved procedure (e.g. reload configuration/firmware)
- Partially or completely unrecoverable (replace hardware and restore firmware)

Some equipment includes watchdog timers, either in software or hardware, which can monitor device operation and restore/restart critical components if needed. Care is needed to ensure that important incident response forensics data are not sacrificed by these countermeasures.

Another defensive approach is to include reasonable amounts of physical protection measures for critical components, such as fuses in an electrical circuit or tank and pipe overpressure valves, to prevent damage even with a fully successful attack. This may require additional overhead to maintain the physical system, but could be minimized with an effective design.

A final countermeasure (that does not have precedent in modern ICS) is to fit the analog outputs from ICS field devices with limiter circuits to ensure a minimal safe operating envelope for the physical system. There are two drawbacks to this approach that need to be considered: first, this increases the complexity of the overall control system (and thus raises the testing/certification burden while reducing reliability); and second, analog circuits inherently experience drift and must often be tuned. However, an effectively-designed and simple analog circuit could minimize impacts and damage even if it could not entirely prevent them, which would be an overall improvement for managing cyber risk.

Chapter 3

Applying the Framework to Develop and Prioritize Access/Procedure Pairs

A unique feature of the FDAM is the introduction of APPs. The APPs have been developed based on empirical data and experience over the last several decades. Out of 16 potential APPs (4 types of access– six total less 2 out-of-scope – and 4 procedures), nine are used in the FDAM approach. Some are excluded given the device-centric theme of the FDAM, and others are emphasized based on several decades of ICS device testing as the most effective to achieve overall risk reduction.

As part of the FDAM approach, vulnerabilities and mitigations are found via a literature review and device physical examination, and also through lab testing. Because not every vulnerability can be tested due to time and resource limitations, the set of possibilities is down-selected for testing. Part of the down selection process includes associating each vulnerability and related mitigation with the appropriate APP – see Appendix A for a detailed example of this test scoping process.

The APPs are presented in prioritized order in Table 3.1, followed by in-depth descriptions including approaches and testing tactics that can be used to verify vulnerabilities that fall under each APP – clearly this is the "verification" aspect of the V/C testing introduced in Chapter 1 (and detailed later in Chapter 6). In addition to using the tactics and approaches listed to verify vulnerabilities, other vulnerabilities might also be discovered in the process – resulting in the "characterization" of the device being tested.

The mapping between APP and goals is also shown in Table 3.1 and is similar for most embedded devices. Procedure P1 (achieving device management access) supports all goals, because it is a critical and fundamental issue. Otherwise, privileged relationships (P2) allow for obtaining/changing data (G3), device impersonation (P3) enables misrepresented state (G2), and atypical stimuli (P4) support denied operation (G4) and behavior change (G1). Given the device-centric approach for this analysis, individual goals are not stressed, but clearly P1 and P4 represent interesting avenues to impacting the device itself (G1 and G4). The other goals remain important, but more in the context of systems instead of individual device assessment. (Note that slave remote I/O for a PLC, coordinated automated control among PLCs, and supervisory control is considered as part of the assessment, under access P4.)

Access	Procedure	Pair	Goals
Network	Atypical stimuli	A2/P4	Change behavior (G1) or deny operation (G4)
Network	Device management	A2/P1	Achieve any goal (G1-5)
Physical	Atypical stimuli	A1/P4	Change behavior (G1) or deny operation (G4)
Physical	Device management	A1/P1	Achieve any goal (G1-5)
Engineering workstation	Device management	A5/P1	Achieve any goal (G1-5)
Peer device	Atypical stimuli	A4/P4	Change behavior (G1) or deny operation (G4)
Peer device	Device management	A4/P1	Achieve any goal (G1-5)
Network	Privileged operational relationship	A2/P2	Obtain or change data (G3)
Network	Device impersonation	A2/P3	Misrepresent state (G2)

Table 3.1: Prioritized APPs with associated goals: these APPs are listed in order of priority, and are an integral part of the FDAM; they have been shown empirically over assessment experience to be the most effective in identifying potential cyber risk

3.1 Network Access (A2) and Atypical Stimuli (P4) to Change Behavior (G1) or Deny Operation (G4)

Within this category, one key tactic is network scanning (e.g. nmap/Nessus, either regular or aggressive, and generic or tailored for the device), spoofing, or flooding in order to attempt to get the ICS to fault, go offline, or slow down. Another important vector is the use of single- or few-packet attacks that have unreasonably large effects to device operation. These would have the same goals, but instead leverage network services (like an unpatched FTP server, vulnerable automation protocols, etc.) or packet replay/custom packets against overly permissive or faulty PLC network interfaces, preferably including reasonable variation in tactics. In some extreme examples, vulnerabilities might allow for direct memory access. Note that these attacks may affect firmware or memory, but this is distinct from A2-P1 (see the following section) in tactics; here, the usage of the P4 designator indicates abnormal network traffic outside of intended vendor operational/configuration procedures attempting to exploit vulnerabilities (which can affect device operation and/or configuration).

There may be special concerns when attempting to affect firmware via atypical stimuli (which by definition is being done outside using conventional vendor management procedures since those are encompassed by P1). Many PLCs have configuration information (ladder logic, I/O memory mapping, etc) compiled together with application code within the configuration software, resulting in one or more segments of code being subsequently sent to the device. Under those circumstances, an unsophisticated attack from the network without vendor configuration software might not discriminate between application and configuration information, leading to unpredictable results.

3.2 Network Access (A2) and Device Management (P1) to Achieve Any Goal (G1-5)

This attack avenue is primarily concerned with network access and vendor software or procedures (for example, some use FTP) to affect firmware (including the device bootloader, executables, and/or configuration). Analysts should include the following approaches when testing vulnerabilities or mitigations that are associated with this APP.

- Using legitimate vendor software (modeled as adversary-owned – not the legitimate copy used for the PLC deployment, but one without any a priori device relationships – in order to simulate any vulnerabilities associated with the device/software relationship)
- Vendor-implemented access methods like network services offered by the device's network interface (e.g. FTP, Telnet, SSH, etc.)

Firmware tactics include (in increasing order of difficulty): reading (which may supply a starting point for modification, represent a theft of sensitive information, or an opportunity to search for vulnerabilities and hardcoded credentials), blanking, altering configuration, altering application code, or altering the bootloader. Since all components of a modular PLC include some sort of programming or firmware, the preferred order of targeting is: central processing unit (CPU) or processing, networking/communications, and finally I/O (note that I/O modules are last because there is little expectation of a significant attack surface, although they are very interesting target). Often, not all devices have all layers (most likely there are not bootloaders on I/O).

Some vendors use separate processes for changing application software versus configuration information on a device's processor module, and under those circumstances the attacker may have more granularity for targeting. Also, some other modules (possibly including the network card) may have configuration information supplied directly (i.e. without compiling), which may allow for higher-discrimination attacks.

Network/communications modules are seemingly attractive for A2/P1 attacks, since they often run conventional operating systems (VxWorks, WinCE, etc.). The CPU firmware is generally quite opaque, which will make some of the more advanced attacks (ones targeting altered firmware) quite difficult as they normally require much more reverse engineering. Changing application code or configurations on the network card may be a low-cost path for an adversary seeking specific effects. For example, the network card might be directed to stop at an assigned time, send erroneous settings along the PLC backplane to I/O, communicate illegitimate network automation messaging to other PLCs to affect a process, help exfiltrate data, etc. Simple P1 actions from the network (A2) may knock the PLC out of a subnet and stop all control and messaging with other devices.

3.3 Physical Access (A1) and Atypical Stimuli (P4) to Change Behavior (G1) or Deny Operation (G4)

Tactics include reasonable, atypical physical stimuli to the PLC done in order to affect operation or operation (any effect that allows G1 carries the same caveat as A2-P4):

- Power cycling
- Effects in device electrical supply (like abnormal power quality)
- Effects caused by RF/EM stimuli
- Remove/exchange/add modules or submodules
- Remove memory cards/chips while powered
- Access JTAG-like functions

The investigation should proceed from simpler vectors to more complex. For example, the assessment should note the effects when power is cycled, including the recovery time, effects on operations, re-establishing communications with peer devices or sub-chassis, or possible problems with physical processes that might be sensitive to the delays. Similarly, analysts should evaluate effects from pulling a module (including the power supply, if it is removable) or perhaps the terminal block on the front of an I/O card, particularly if physical damage is feasible. If possible, the analyst can consider the possibility that an adversary adds a hacked module or something else to the backplane or to a local port in order to achieve nefarious goals (specifically for P4, in order to deny or alter device operation without utilizing intended vendor configuration procedures). Also, if memory is removable (like a chip or card), then the assessment should measure the effects. Finally, JTAG connections or other test/debug ports offer enormous attack opportunity and could be investigated in this stage of the analysis as presumably any aspect of firmware is potentially targetable in this manner.

3.4 Physical Access (A1) and Device Management (P1) to Achieve Any Goal (G1-5)

For this section, some key attack opportunities include:

- Move jumpers or change switch positions
- Exchange memory cards/chips during power cycles
- Inserting USB/peripherals
- Accessing local configuration ports (that are distinct from the primary network port)

The analysis should check whether there are reset switches or jumpers that could blank firmware or reset security access authorization, or perhaps switches (like RUN/PROGRAM or REMOTE/LOCAL) that affect operation. Also, if memory is removable, then the assessment should evaluate the possibility of changing device configuration via the exchange, and the needed procedures to accomplish that (power cycling, logic reload, automatic operational change, etc.). In addition, given their potential to circumvent air gap perimeters, USB key vectors, electronic warfare (EW), and RF-enabled cyber may be included if feasible.

There is a critical avenue for investigation if any of the PLC modules offer a local configuration or access port that is distinct from the primary network port. These local access ports are often serial DB9 connections. They may offer better access to configurations – meaning there might be less access authorization (or perhaps the bootloader is only readable/changeable via local serial ports). The scope of tactics for local device access is the same as for A2-P1, including legitimate vendor software (modeled as adversary-owned), vendor-described access methods, attacks against vulnerable services on local ports, or other hacks that may affect firmware.

3.5 Engineering Workstation Access (A5) and Device Management (P1) to Achieve Any Goal (G1-5)

The target of investigation in these attacks is the authorized copy (or copies) of the vendor-designated configuration capability intended for use on the given PLC. Very often, this takes the form of a laptop with various connectivity options and some enforcement of user security. Other times, there is an "engineering workstation" that is permanently connected to an automation network that is used for the task. In both A2- and A1-P1, the effects of attacks using copies of the vendor software (but not the authorized copies themselves) were considered. This is a critical distinction, as there may be severe technical difficulties achieving P1 without the legitimate copy of the authorized software. The engineering workstation copy may already include the target PLC configuration (which could be critical if they are not easily read from the device) that may be a necessary first step for modifying firmware. Alternatively, there might be stored security credentials that allow for easier access to device firmware (although this is unlikely – in many cases, a user authenticates to the local configuration software, authorized or unauthorized, and not the PLC). Finally, A2 or A1 may not be achievable for an adversary, and achieving A5 (possibly via social engineering or attacking the host OS) might be the best alternative.

Therefore, the tactics to be investigated for the A5 target include the ones previously described in A2-P1 or A1-P1, but emphasizing the additional opportunities of stealing PLC firmware, configurations, or access tokens. The analysis should also (as feasible) investigate the possibility of changing local PLC firmware or configurations on the authorized configuration management platform in a way that will eventually make it to the operating device (which may amount to a significant amount of reverse engineering (RE) against the OS and vendor software).

3.6 Peer Device Access (A4) and Atypical Stimuli (P4) to Change Behavior (G1) or Deny Operation (G4)

A4 refers to a peer device, and in this case references a subchassis for the primary PLC (given that it does operate autonomously to at least some extent) or some other device with an existing privileged relationship to target field device (perhaps another PLC or an ICS server) that affords opportunities for P4 that are different than those achievable via simple network access (A2). One key concern is that a PLC subchassis might be more accessible than the primary PLC, so the analysis should assume full access – including physical if necessary – to the subchassis in order to gauge the vulnerability to the main unit via A4. These connections are typically either serial or TCP/IP (often separate from the main ICS WAN, covered in A2), although highway addressable remote transducer (HART) or others might be encountered. When the connections are similar to the primary PLC network connection, then it is a straightforward argument that the results from the A2-P4 investigations are relevant. However, if these connections employ different TCP/IP networking modules, then the analysis process should be repeated. Also, if A4 access is via a TCP/IP network card, then the same issues for A2-P4 need to be considered to try and get the

PLC to fault, go offline, or slow down significantly: scans (whatever is applicable to the interface); single- or few-message attacks leveraging port services (likely confined to the designated port protocol, but the possibility of others should be considered) or faulty port software, etc.; this may also include fuzzing. The results may affect firmware or memory, but this is again considered distinct from A4-P1 (as these are characterized as atypical stimuli).

3.7 Peer Device Access (A4) and Device Management (P1) to Achieve Any Goal (G1-5)

This attack scenario is relevant only if the port used for the peer or subchassis communications allows for device management. Indirect attack via an adversary accessing the designated supervisory control platform is also included. For TCP/IP networks, this is unlikely to be different from A2-P1, or from A1-P1 for serial networks. Any differences should be noted.

3.8 Network Access (A2) and Privileged Operational Relationship (P2) to Obtain/Change Data (G3)

Using network access, direct device operation using seemingly legitimate commands and/or read data from device. Given the known poor state of ICS protocol cyber security, this access/procedure pair is not novel, so only limited analysis is expedient to avoid expending project resources on known problems. However, successful commands via the network can still cause the field device to innocently execute control changes that can affect the underlying process.

3.9 Network Access (A2) and Device Impersonation (P3) to Misrepresent State (G2)

Similarly to A2-P2, known problems with protocols allow an attacker to intercept device communications and send false messaging upstream. Barring any unique aspects to this problem, minimal analysis is warranted.

Chapter 4

Cyber Risk Quantification

System and device vulnerabilities can be uncovered by several approaches detailed in the next two chapters (literature review, device physical examination, and two stages of lab testing). It is not uncommon for a single device to have multiple associated vulnerabilities, and so the tester will most likely come up with a long list of potential vulnerabilities when researching a specific ICS device. However, not all vulnerabilities are created equal – some are associated with much higher risk that others. The risk calculation presented here is a tool used to quantify and score the risk posed by each vulnerability. Once each vulnerability is scored, it can be further ranked by associating it with its associated APPs presented in Chapter 3 to determine if it should be tested later (testing is covered in Chapter 6).

The risk calculation presented here leverages the CVSS methodology, used by National Institute of Standards and Technology (NIST) and United States Computer Emergency Readiness Team (US-CERT). The CVSS system was designed for IT, and so it emphasizes different priorities and scoring options that are not applicable for ICS. However, the basic structure of the CVSS system is valid, and has been repurposed to be applicable to ICS devices (with the caveat that normalized ICS scores are only comparable with like scores, and are not one-to-one comparable with normal CVSS IT scores).

According to CVSS, a "base score" depends on quantitative scores for impact and exploitability according to the following formula:

$$Score = Exploitability + Impact \tag{4.1}$$

The CVSS formula for exploitability is:

$$8.22\,(Attack\,Vector)\,(Attack\,Complexity)\,(Privilege\,Required)\,(User\,Interaction) \tag{4.2}$$

And the confidentiality (C), integrity (I), availability (A) total impact score per CVSS is determined according to the following:

$$6.42\,(1-(\,(1-Impact.C)\,(1-Impact.I)\,(1-Impact.A)\,)\,) \tag{4.3}$$

Once calculated, the score in equation 4.1 is rounded up or down to one decimal place (and limited to the allowable range of 1 to 10 if needed). The CVSS formulas will be used as given for the FDAM ratings. The severities shown in Table 4.1 are assigned by CVSS, but these do not account for the importance of ICS and so should be carefully considered at the lower end.

Rating	**Risk Score**
None	0.0 - 0.1
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Table 4.1: Qualitative severity rating scale

The individual ratings and corresponding quantitative values for the metrics are shown in Table 4.2. The determination of each score is addressed in the subsequent sections. Each will introduce the metric and the original CVSS considerations, and then describe how they are used in the FDAM.

Metric	**Metric Value**	**Numerical Value**
Attack Vector	Network	0.85
	Adjacent	0.62
	Local	0.55
	Physical	0.20
Attack Complexity	Low	0.77
	High	0.44
Privilege Required	None	0.85
	Low	0.62
	High	0.27
User Interaction	None	0.85
	Required	0.62
C, I, A Impact	High	0.56
	Low	0.22
	None	0.00

Table 4.2: Metric values

4.1 Attack Vector

This category maps to the FDAM framework access category, and it will not be affected by procedure or mitigation considerations. Table 4.3 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM access categorization.

Metric Value	CVSS Standard Description	FDAM Description
Network (N): 0.85	A vulnerability exploitable with network access means the vulnerable component is bound to the network stack and the attacker's path is through OSI layer 3 (the network layer). Such a vulnerability is often termed "remotely exploitable" and can be thought of as an attack being exploitable one or more network hops away (e.g. across layer 3 boundaries from routers). An example of a network attack is an attacker causing a denial of service (DoS) by sending a specially crafted TCP packet from across the public Internet (e.g. CVE-2004-0230).	Includes any A2 or A4 that works across router boundaries.
Adjacent (A): 0.62	A vulnerability exploitable with adjacent network access means the vulnerable component is bound to the network stack, however the attack is limited to the same shared physical (e.g. Bluetooth, IEEE 802.11), or logical (e.g. local IP subnet) network, and cannot be performed across an OSI layer 3 boundary (e.g. a router). An example of an Adjacent attack would be an ARP (IPv4) or neighbor discovery (IPv6) flood leading to a denial of service on the local LAN segment. See also CVE-2013-6014.	Includes any A2 or A4 limited by a router boundary (a single collision domain).
Local (L): 0.55	A vulnerability exploitable with local access means that the vulnerable component is not bound to the network stack, and the attacker's path is via read/write/execute capabilities. In some cases, the attacker may be logged in locally in order to exploit the vulnerability; otherwise, she may rely on user Interaction to execute a malicious file.	A3 explicitly relies on user interaction.
Physical (P): 0.20	A vulnerability exploitable with physical access requires the attacker to physically touch or manipulate the vulnerable component. Physical interaction may be brief (e.g. evil maid attack) or persistent. An example of such an attack is a cold boot attack, which allows an attacker to access to disk encryption keys after gaining physical access to the system, or peripheral attacks such as Firewire/USB Direct Memory Access attacks.	Same as CVSS (A1 is an exact match for the CVSS definition).

Table 4.3: Risk scoring: attack vector

4.2 Attack Complexity

This category is a binary choice measure that will characterize the simplicity (or complexity) of the procedure (assuming some given access level), but not emphasizing the magnitude of the effort. Mostly, for FDAM, the attack complexity will map to the amount of prior knowledge (device information and situational reconnaissance) and/or specific conditions that must be met to successfully execute the vulnerability against the target system. Table 4.4 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM complexity categorization.

Metric Value	**CVSS Standard Description**	**FDAM Description**
Low (L): 0.77	Specialized access conditions or extenuating circumstances do not exist. An attacker can expect repeatable success against the vulnerable component.	Includes all severe vulnerabilities for P4 and P1.
High (H): 0.44	A successful attack depends on conditions beyond the attacker's control. That is, a successful attack cannot be accomplished at will, but requires the attacker to invest in some measurable amount of effort in preparation or execution against the vulnerable component before a successful attack can be expected. For example, a successful attack may depend on an attacker overcoming any of the following conditions: • The attacker must conduct target-specific reconnaissance. For example, on target configuration settings, sequence numbers, shared secrets, etc. • The attacker must prepare the target environment to improve exploit reliability. For example, repeated exploitation to win a race condition, or overcoming advanced exploit mitigation techniques. • The attacker must inject herself into the logical network path between the target and the resource requested by the victim in order to read and/or modify network communications (e.g. man in the middle attack).	Includes vulnerabilities allowing P1 or P4 with either significant reconnaissance or extenuating circumstances, and nearly all P2/P3 (as protocol-based vulnerabilities nearly always require some reconnaissance to learn the data scheme).

Table 4.4: Risk scoring: attack complexity

4.3 Privileges Required

This category is primarily concerned with a measure of the authorization level (after assuming the given access) needed to leverage the vulnerability within a procedure. Table 4.5 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM for this metric.

Metric Value	CVSS Standard Description	FDAM Description
None (N): 0.85	The attacker is unauthorized prior to attack, and therefore does not require any access to settings or files to carry out an attack.	Same as CVSS.
Low (L): 0.62	The attacker is authorized with (i.e. requires) privileges that provide basic user capabilities that could normally affect only settings and files owned by a user. Alternatively, an attacker with Low privileges may have the ability to cause an impact only to non-sensitive resources.	Same as CVSS (note that nearly all P2/P3 will be here).
High (H): 0.27	The attacker is authorized with (i.e. requires) privileges that provide significant (e.g. administrative) control over the vulnerable component that could affect component-wide settings and files.	Same as CVSS.

Table 4.5: Risk scoring: privileges required

4.4 User Interaction

This metric is a simple binary measure of the need (or not) of user action to allow an attacker to exploit the vulnerability. Table 4.6 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM for this metric.

Metric Value	CVSS Standard Description	FDAM Description
None (N): 0.85	The vulnerable system can be exploited without interaction from any user.	Same as CVSS.
Required (R): 0.62	Successful exploitation of this vulnerability requires a user to take some action before the vulnerability can be exploited. For example, a successful exploit may only be possible during the installation of an application by a system administrator.	Includes all A3 (Table 4.3) and other vulnerabilities that depend on user action (like replay attacks, USB insertion, etc.).

Table 4.6: Risk scoring: user interaction

4.5 Confidentiality Impact

The interpretation of confidentiality in FDAM is the extent that the vulnerability reveals data, including reconnaissance for later adversary action. A high rating indicates that information about the logic of a system is revealed, while a low rating represents a loss of operational data. Table 4.7 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM confidentiality impact.

Metric Value	**CVSS Standard Description**	**FDAM Description**
High (H): 0.56	There is total loss of confidentiality, resulting in all resources within the impacted component being divulged to the attacker. Alternatively, access to only some restricted information is obtained, but the disclosed information presents a direct, serious impact. For example, an attacker steals the administrator's password, or private encryption keys of a web server.	Loss of confidentiality for system logic or key device configuration parameters (including authentication information).
Low (L): 0.22	There is some loss of confidentiality. Access to some restricted information is obtained, but the attacker does not have control over what information is obtained, or the amount or kind of loss is constrained. The information disclosure does not cause a direct, serious loss to the impacted component.	Loss of confidentiality for system operational information.
None (N): 0	There is no loss of confidentiality within the impacted component.	Same as CVSS.

Table 4.7: Risk scoring: confidentiality impact

4.6 Integrity Impact

Integrity for ICS field devices refers to the accuracy for both device and system data as well as the operational configuration. In particular, integrity of the field device's operations refers to it executing its functions in a timely and accurate manner, both of which can be affected by adversary action. In an extreme case, this could lead to a loss of control functionality (clearly, this could also be an availability issue, but for this modified CVSS ICS formulation, it is noted here as a characteristic of integrity). Table 4.8 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM integrity impact.

Metric Value	**CVSS Standard Description**	**FDAM Description**
High (H): 0.56	There is a total loss of integrity, or a complete loss of protection. For example, the attacker is able to modify any/all files protected by the impacted component. Alternatively, only some files can be modified, but malicious modification would present a direct, serious consequence to the impacted component.	Modification of device configuration (via P1 or P4), including firmware, logic, addressing, or authentication information, including corresponding impacts to local control.
Low (L): 0.22	Modification of data is possible, but the attacker does not have control over the consequence of a modification, or the amount of modification is constrained. The data modification does not have a direct, serious impact on the impacted component.	Modification of operational information to/from the device/sub-chassis, including corresponding impacts to supervisory control. Although there is not a serious loss for the device, there may be significant impact to the controlled process.
None (N): 0	There is no loss of integrity within the impacted component.	Same as CVSS.

Table 4.8: Risk scoring: integrity impact

4.7 Availability Impact

Availability for ICS field devices refers to the duration of downtime, which is crucial for these systems. The downtime is directly associated with the effort involved in restoring the device to its intended operation. Only three levels and a simple criterion are used, to maintain compatibility with the CVSS system. Table 4.9 includes the rating, numeric score, original CVSS definition, and the mapping to the FDAM availability impact.

Metric Value	**CVSS Standard Description**	**FDAM Description**
High (H): 0.56	There is total loss of availability, resulting in the attacker being able to fully deny access to resources in the impacted component; this loss is either sustained (while the attacker continues to deliver the attack) or persistent (the condition persists even after the attack has completed). Alternatively, the attacker has the ability to deny some availability, but the loss of availability presents a direct, serious consequence to the impacted component (e.g., the attacker cannot disrupt existing connections, but can prevent new connections; the attacker can repeatedly exploit a vulnerability that, in each instance of a successful attack, leaks a only small amount of memory, but after repeated exploitation causes a service to become completely unavailable).	Requires physical user intervention to restore the field device to operation.
Low (L): 0.22	There is reduced performance or interruptions in resource availability. Even if repeated exploitation of the vulnerability is possible, the attacker does not have the ability to completely deny service to legitimate users. The resources in the impacted component are either partially available all of the time, or fully available only some of the time, but overall there is no direct, serious consequence to the impacted component.	Requires remote user intervention to restore the field device to operation.
None (N): 0	There is no impact to availability within the impacted component.	Same as CVSS.

Table 4.9: Risk scoring: availability impact

Chapter 5

Literature Review and Device Physical Examination

Prior to any testing, the analyst should uncover as many vulnerabilities and potential mitigations as possible for the ICS device in question. Two approaches can be considered:

- Literature review: Perform a comprehensive literature review to determine known vulnerabilities and mitigations
- Device physical examination: Examine the key components of the system if possible, which may also suggest vulnerabilities or mitigations

Background research and testing feed into each other – knowing about existing vulnerabilities can lead to uncovering other similar vulnerabilities.

If lab testing is planned, the results of the literature review and device examination should conclude with a general discussion of the best candidates (both vulnerabilities and mitigations) for demonstration. The process to scope the vulnerabilities and mitigations to be tested is detailed in Chapter 6, and is based on combining the APP scores from Chapter 3 with the risk scores presented in Chapter 4. An example of this scoping/down-selection process is shown in Appendix A.

Vulnerabilities should include common problems (like well-known issues with ICS technology and architectures) as well as specific risks (for example, a vulnerability to crashing when scanned from the network, or devices freezing if communication is lost). All available information about the device and its intended deployment, as well as a range of access avenues and tactics that cyber adversaries could employ, and the device's physical environment should be examined. The attack surface for these devices may include firmware, settings, networking, configuration platforms, peer devices, and potential physical vectors. Countermeasures range from general cyber fixes (e.g. external security enhancement like VPN encapsulation) to specific cyber fixes (e.g. configuring onboard security measures).

5.1 Literature Review Process

The literature review is intended to gather background information about the device and research both known and potential vulnerabilities and mitigations. Of course, when gathering all available information about the ICS device it may be limited by the assessment scope (e.g. classified/unclassified data, public/government information, etc.).

When gathering background information, consider the following:

- Application space (historical and evolving)
- (optional) Device history
  - Production timeline
  - Vendor background
- Operational procedures, requirements, and architecture
  - Device administration/lifecycle
  - Configuration/programming procedures
  - Firmware update process
  - Vendor specific patching, etc
  - Web based applications
  - Vendor documentation and manuals
- Chassis communications
- Software architecture
- Cyber security capabilities, including recovery/restoration
- Communication protocol vulnerabilities
- (optional) Other
  - Application focus (specifically for this report, e.g. ship systems, manufacturing, etc.)
  - Known incidents (the sensitivity will impact the final report's sensitivity)

There are many cyber-security based websites that provide information about ICS device vulnerabilities and mitigations. Conference proceedings also provide a good avenue to research vulnerabilities. Following is a partial list of references to use for research into ICS device vulnerabilities and mitigations:

- SCADAhacker
- BlackHat/DefCon/TakeDown presentations
- Digital Bond website
- ICS-CERT
- Google Scholar
- IEEExplore
- Rapid7
- Manufacturers' sites and product documentation
- Common vulnerabilities and exposures (CVE) database
- S4 conferences
- Manufacturer's website
- Other relevant subject matter discussion boards

5.2 Device Physical Examination

If at all possible, the device should be examined physically. The hardware subcomponents that make up the device might have potential vulnerabilities not found in the initial literature review and would require their own subsequent literature review specific to the subcomponent. For example, a specific make and model of a PLC CPU module, the CPU chip, system-on-a-chip (SoC), communications processor (CP) module, or Ethernet chipset might have intrinsic vulnerabilities that could be uncovered with focused research. The following list is a good overview of potential components to consider when doing a physical examination of the device:

- Device construction
- CPU and/or CPU module - processor type/architecture, identifier, model
- Memory – SDRAM, DRAM, etc.
- Communications processors
- Chassis communications
- Debug ports
- Connection ports such as Ethernet ports, serial connectors, etc. (pins, pads, or connectors)
- Memory or storage cards like PCMCIA or MultiMediaCard MMC

The physical examination should first start with an assay of the major device components with potential cybersecurity components. Examples of major components can include the CPU and CP network card. It is also worth examining any medium that the PLC components use to communicate with each other and with other devices. Different communication mediums can include general-purpose network interfaces like Ethernet, Token Ring, RS-485, or specialized PLC interconnects, as well as the corresponding protocols (such as Modbus and Profibus). It is also worth gaining at least a cursory understanding of what the backplane bus is (VME, K-Bus/P-bus, etc.).

After performing a holistic assessment of the PLC hardware system, each component should then be examined. The rest of this section discusses the procedure for examining and characterizing the security of each of the major components. The first step is to examine the exterior of the device for any sort of debug or console ports. These ports may take on the form of exposed pins, pads, or even full-fledged connectors like DB9 ports for RS-232 connectors. Other possible connections include slots for inserting memory or storage cards like PCMCIA or MMC. The next step after examining the exterior of each module is to remove the outer housing with the goal of examining the printed circuit boards (PCBs) that make up the module. Any other possible debug ports or memory card slots should be noted here.

Upon gaining a full view of the board, it is now time to characterize the embedded hardware with the goal of determining the "what" and "how" of the hardware. Major components should be identified, including:

- *CPU/SoC:* Determine what the CPUs for the module are.
- *Volatile memory:* Determine whether it is static random-access memory (SRAM) or DRAM, and the amount of memory that are attached to the CPUs.
- *Non-volatile memory:* Determine whether there is any sort of non-volatile memory. This memory may be EEPROM, NOR Flash, or NAND flash. It is also important to note whether this storage is socketed, which can make it easier for an attacker to modify the module's firmware.
- *Transceiver chips:* Determine what sort of off-board communications components are on the PCB. Examples include Ethernet physical transceivers (PHYs), serializer/deserializers (SERDESs), level shifters, and bus interfaces.

Successfully identifying these components is essential to determine whether any access opportunities may exist (including debug interfaces, communications, memory sockets, bus interfaces, etc.). Examining the area around the CPU/SoC area for exposed pins or pads can also help find hardware access avenues.

5.3 Collating the Results

The vulnerabilities and potential mitigations discovered in the literature review and device physical examination need to be summarized into tables, and should be given unique labels to track them throughout the process. The example shown in Appendix A labels vulnerabilities as V_A, V_B, ... and mitigations are labeled M_A, M_B, ... For each vulnerability tested, the following information should be captured when feasible:

- Brief vulnerability description and unique identifier
- References – public or private/government material
- Dates of discovery and last analysis (approximate if necessary)
- Vulnerability risk score and ranking
- Access/procedure pair and goals supported
- Testing approaches used
- Safety/damage impact given the application (if known)
  - Unexpected or unpredictable physical system behavior
  - Damage to physical system components
  - Potential for DOS of local automated or manual control
  - Potential for DOS of supervisory control
- Additional notes and analysis, if any
- Applicable mitigations, if any, and the results of testing if available

These are later down-selected for testing using using the risk score and APP prioritization (Chapter 3). Testing is described in the next chapter.

Chapter 6

Testing the ICS Device

Testing the ICS device in a lab setting serves several purposes. Cyber security testing on operational devices might have adverse effects – especially for an ICS field device. Something as simple as scanning a port to look for vulnerabilities can cause a field device or ICS process to fail – and if the device is use and operational, this failure could have serious consequences. Mitigations need to be tested as well, as they can counterproductively lead to serious operational impacts (like a mitigation that could fail and block critical communications). Performing laboratory assessments tests the effects of vulnerability and mitigations testing, which allows the tester to develop requirements for operational tests that minimize the potential impacts to reliability on operational equipment.

The testing process is iterative in nature. As the device is tested, new vulnerabilities might be exposed, and new mitigations discovered that are more specific to the device itself. There are three aspects involved in testing – all or some of them may be incorporated depending on the experience of the testers, available resources, and desired outcomes. The two phases of lab assessment are:

- **Verification/characterization testing:**

  V/C testing determines the effects of the vulnerabilities and mitigations selected for testing on the system or device. V/C testing may also reveal other, equipment specific issues (not found in the literature review or device physical examination), which can be analyzed further via exploration testing. V/C testing also determines whether mitigations achieve a needed result, and can be implemented safely and reliably on an operational ICS. V/C testing relies on the approaches and testing tactics associated with each of the prioritized APPs (Chapter 3). The tactics are used initially to verify vulnerabilities that fall under each APP – clearly this is the "verification" aspect of the V/C testing. In addition to using the tactics and approaches listed to verify vulnerabilities, others might also be discovered in the testing process – resulting in the "characterization" of the device being tested.

- **Exploration testing:**

  Based on the results of V/C testing, further exploration of newly discovered potential device or application-specific vulnerabilities (V3/V4 from Chapter 2) might be required. Exploration testing focuses on key cyber/physical interactions, mission operations, and ensuring safe OT&E implementation.

6.1 Test Scoping: Deciding What to Test

The literature review and device research, and also the V/C testing itself, identify vulnerabilities and potential mitigations. However, resources and time do not allow for testing everything, and some vulnerabilities simply may not need to be tested in order to ensure a device is secure (i.e they score very low on the risk score calculation or are associated with a low priority APP). The goal is to focus efforts on testing the vulnerabilities and mitigations that will provide the greatest degree of security.

The FDAM approach includes guidance to determine the best set of tests to run. Prior to V/C testing, the APPs (Chapter 3) and the risk calculation (Chapter 4) are used to narrow down the vulnerabilities and mitigations found in the literature review and device physical examination (this sensibly limits the "verification" phase of the V/C testing). A similar, but slightly modified approach is used to narrow down vulnerabilities and mitigations that are discovered during the V/C testing and further analyzed in exploration testing.

A brief overview of the process is presented here, and example of how to use the process is presented in Appendix A:

- Identify and tabulate vulnerabilities and mitigations
- Calculate risk scores and rank all vulnerabilities
- Tabulate vulnerabilities, their risk scores/ranks, and mitigations into APPs bands
- Down-select the vulnerabilities to be tested (rules are shown in Table 6.1)
- Down-select mitigations associated with the vulnerabilities based on achievability and effectiveness
- Finalize list of vulnerabilities and mitigations to be tested

The analyst tabulates the vulnerabilities, and uses the selection rules from Table 6.1 to form the subset of vulnerabilities to be tested. Also, each vulnerability has associated mitigations, and these may also be tested (provided testing would are achievable, e.g. some mitigation steps are only applicable at the design phase, like conformal coatings for circuit boards). Mitigations selected for testing should include ones with the best likelihood of being effective (i.e. low operational impact, address more than one vulnerability, simple to configure, address high-risk issues, etc.). After the final list of vulnerabilities and mitigations is determined, then testing is performed.

Ordered APPs	Vulnerability Test Rule
A2-P4 (G1, G4) A2-P1 (G1–G5) A1-P4 (G1, G5) A1-P1 (G1–G5)	Test All
A5-P1 (G1–G5) A4-P4 (G1, G4) A4-P1 (G1–G5)	Test Critical, High, Medium
A2-P2 (G3) A2-P3 (G2)	Test Critical & High

Table 6.1: The APPs described in Chapter 3 are presented in order of importance, and banded into to high, medium, and low importance, with the risk scores to be tested within each band

6.2 Verification/Characterization Testing

Once the vulnerabilities and mitigations to be analyzed have been determined (for "verification" testing, based on the process above and shown in Appendix A), then the testing can begin. The goal of V/C testing is to demonstrate the discovered vulnerabilities and evaluate potential mitigations to determine if they are effective (and can be implemented both securely and safely). In the process of using the testing approaches suggested by the APP prioritization, other vulnerabilities specific to the equipment being tested might be discovered (the "characterization" part of V/C). It is assumed no new general or system level vulnerabilities will be discovered (vulnerability categories V1 and V2 from the assessment framework), as these should have been uncovered in the literature review and device physical examination. The newly identified vulnerabilities will then be examined in subsequent exploration testing (if desired and feasible).

Following are examples of testing approaches used to *verify* the vulnerabilities discovered in the literature review and device physical examination, and also *characterize* additional vulnerabilities associated with each of the APPs (these were previously mentioned in Chapter 3):

1. Network access and atypical stimuli (A2/P4) to change behavior (G1) or deny operation (G4)
    - Network scanning
    - Single- or few-packet attacks
    - Replay attacks
    - Memory access via ICS protocols
2. Network access and device management (A2/P1) to achieve any goal (G1-5)
    - Via vendor software or remote access (e.g. FTP, SSH, etc.)
    - Target any device containing code, particularly CPU and communication modules
    - Attempt to read firmware
    - Blank or alter firmware
    - Change operating logic
    - "Brick" the device
3. Physical access and atypical stimuli (A1/P4) to change behavior (G1) or deny operation (G4)
    - Power cycling
    - Remove/exchange/add modules or submodules while powered on
    - Remove memory cards/chips while powered on
    - Access JTAG-like functions
    - Evaluate effects from device electrical supply (like abnormal power quality)
    - Evaluate effects from RF/EM
4. Physical access and device management (A1/P1) to achieve any goal (G1-5)
    - Move jumpers or change switch positions
    - Exchange memory cards/chips
    - Inserting USB/peripherals
    - Local configuration or access ports that are distinct from the primary network port
5. Engineering workstation access and device management (A5/P1) to achieve any goal (G1-5)
    - Target is a presumed authorized mobile computing platform for ICS management
    - Similar opportunities to A2/P1, but may assume that the device configuration will be known locally
    - Can allow access if A2 or A1 are unavailable
    - OS services may enable some new attacks against vulnerabilities
6. Peer device access and atypical stimuli (A4/P4) to change behavior (G1) or deny operation (G4)
    - Vulnerabilities may be similar to A2/P4 if the connections are TCP/IP
    - If serial, then similar questions apply, but likely using different approaches

7. Peer device access and device management (A4/P1) to achieve any goal (G1-5)
    - Only relevant if peer or subchassis communications ports allow for P1
    - Vulnerabilities may be similar to A2/P1
8. Network access and privileged operational relationship (A2/P2) to obtain data (G3)
    - Given the normal lack of security services for ICS protocols, this should be achievable
    - Vulnerabilities are well-known, so issues should be noted with minimal resources expended
9. Network access and device impersonation (A2/P3) to misrepresent state (G2)
    - Similarly to A2/P2, this should be achievable
    - Vulnerabilities are well-known, so issues should be noted with minimal resources expended

There are other important requirements to guide the testing, especially the development of a suitable test environment. The ICS device being tested should be set up using a configuration suitable for the application (subject to available information). Key details include:

- A varied set of components (including power supply, CPU, binary I/O, analog I/O, network/communications interfaces, and cyber security modules if applicable)
- Remote I/O and/or sub-chassis as reasonable and feasible (a common feature for ICS field devices)
- If needed, a minimal set of analog connections such as simple signal generators and meters (to show proper I/O and state-of-health for onboard logic; note that testers have often configured front panel LEDs to show functioning internal logic)
- Representative network connections for supervisory control (where applicable), including monitoring and control traffic
- Common access methods for device configuration (physical/local and network)
- Two copies of the vendor's management software (one with the device configuration stored to test the A3 access path, and a second to exercise other P1 vulnerabilities)

All testing should include responsible approaches and documentation, including:

- Hypotheses
- Test requirements
- Success/fail indicators
- Test plan and execution

These elements should support the testing, but not dominate the effort. The FDAM is exploratory, and not intended to be a checklist.

Important cyber security tools that can be used during V/C (and later) testing include:

- Vendor software
- Open-source ICS protocol implementations
- IDA Pro (disassembler/debugger)
- Wireshark (network analysis)
- NMap (network mapping)
- Nessus (vulnerability scanning)
- Metasploit, Immunity Canvas, Kali Linux (penetration testing)
- Samurai STFU (electric utility ICS based penetration testing)
- Cobalt Strike (threat emulation)
- Scapy (packet manipulation)
- tcpdump/tcpreplay (packet analyzer)
- binwalk (firmware analysis)

For each vulnerability tested, the following information should be captured when feasible (the same information suggested in Section 5.3):

- Brief vulnerability description and unique identifier
- References – public or private/government material
- Dates of discovery and last analysis (approximate if necessary)
- Vulnerability risk score and ranking

- Access/procedure pair and goals supported
- Testing approaches used
- Safety/damage impact given the application (if known)
  - Unexpected or unpredictable physical system behavior
  - Damage to physical system components
  - Potential for DOS of local automated or manual control
  - Potential for DOS of supervisory control
- Additional notes and analysis, if any
- Applicable mitigations, if any, and the results of testing if available

The results of the testing will be compiled and documented according to the final test report format (detailed in the next chapter).

6.3 Exploration Testing

Exploration testing proceeds similarly to V/C testing. But because the vulnerabilities being explored are a result of the V/C testing, they should all be equipment- or application-specific vulnerabilities (V3/V4). Therefore, in the exploration testing phase, the V3/V4 categories are incorporated. The exploratory testing proceeds from the new vulnerability/mitigation summary tables in a way identical to the down-selection for the V/C testing. The steps are listed in detail below (and also in Appendix A):

- Identify and tabulate vulnerabilities and mitigations discovered through V/C testing
- Calculate risk scores and rank all vulnerabilities (check that all are V3 or V4)
- Tabulate vulnerabilities, their risk scores/ranks, and mitigations into APPs bands
- Down-select the vulnerabilities to be tested (same rules as V/C testing)
- Down-select mitigations associated with the vulnerabilities based on achievability and effectiveness
- Finalize the list of vulnerabilities (all should be V3 or V4) and mitigations to be tested

The exploratory testing phase has many of the same requirements and suggestions as the V/C testing process. There is a need for a properly-configured test environment, suitable test plans, and similar software/hardware testing capabilities. The results from the exploratory testing will be captured using the same formats shown in the prior section. When exploration testing is completed, then the final step is to assemble the test report, which is detailed in the next chapter.

Chapter 7

Conclusion: Writing a Device Test Report

This final chapter of the FDAM describes the major sections of the device final report based on executing the FDAM approach. The FDAM accumulates a significant body of knowledge via the literature review, physical examination, risk scoring/ranking, test scoping, and test results are complete (which phases are included can be modified by the analyst based on personal experience and project resources).

The report is narrative in nature; the outline below is a suggestion as to how to present and compile that information to convey the thoroughness of the approach and ensure that reliable conclusions and recommendations have been reached. The narrative also captures testing details to allow for replication and reinvestigation. The report should emphasize the identification of vulnerabilities, reducing the cyber security risk, and finally managing operational test safety – by clearly delineating significant issues that could lead to damage or safety risks during operational testing.

Key report sections include:

- Introduction
  - FDAM overview
  - Background for the field device being analyzed
- Literature review and device physical examination
  - Literature review sources and results
  - Device physical examination (key observations, etc.)
  - Summary descriptions for vulnerabilities and mitigations discovered (per the suggested format toward the end of Section)
- Verification/characterization testing
  - Scoping; the down-selection process to identify vulnerabilities and mitigations to be tested (see Appendix A)
  - Testing setup and background
  - Testing results (also using the summary format then applicable; V/C realists may be selected for further exploratory testing)

- Exploration testing
  - Scoping; down-select the vulnerabilities and mitigations to be tested based on V/C testing (see Appendix A)
  - Testing setup and background (may be similar to V/C tests, e.g. the device configuration and lab equipment)
  - Testing results, scored and summarized
- Summary
  - Conclusions
    * Summarize all vulnerabilities, risk scores, and if they were tested
    * Summarize all mitigations, what vulnerabilities they address, if they were tested, and whether they can be added to operational systems
    * Summarize the concerns for operational testing, and applicable controls to manage damage/safety risks
  - Recommendations
    * Vulnerabilities in most need of being addressed, and necessary follow-on analysis
    * How mitigations should be implemented to reduce risk without unreasonable impacts to operation
    * Other tests that may be needed to fully characterize operational test safety

Each of the major sections is addressed individually in the subsequent sections.

7.1 Introduction

A brief overview of the FDAM process should be included in the final report. Also, a suggested outline for reporting the device background information includes:

- Background - what is being tested and why
- (optional) History of the device under test
  - Application space (historical and evolving)
  - Production timeline
  - Vendor background
- Basic overview of the device construction
  - Hardware modules
  - Chassis communications
  - Software architecture

- Administration/lifecycle procedures
  - Configuration/programming process
  - Firmware update process
  - Cyber security capabilities, including recovery/restoration
- (optional) Application focus (specifically for this report, e.g. ship systems, manufacturing, etc.)
- (optional) Known incidents (the specificity will impact the final report's sensitivity)

7.2 Literature Review and Device Physical Examination

This section is composed as a narrative of the activities that were pursued and the vulnerabilities and mitigations that were discovered as part of the literature review and device examination (Chapter 5). It should include the tools, sources, and methods that were employed as part of the process.

7.2.1 Literature Review

The literature review (Section 5.1) is aimed at finding all the vulnerabilities and mitigations available from relevant cyber security-based websites, conference proceedings, manufacturers websites, and other relevant subject matter discussion boards. This section of the report should briefly summarize all the vulnerabilities and mitigations found.

7.2.2 Device Physical Examination

If the device is able to be examined physically, this section will consist of a narrative detailing the approach used, and should include a detailed description of the major components (Section 5.2). These components include the CPU, CP/network cards, debug and console ports, other connection ports such as slots for memory or storage cards,volatile and non-volatile memory, printed circuit boards, etc. Vulnerabilities based on the devices physical components should be captured.

7.2.3 Vulnerabilities and Mitigations Discovered

The literature review and device physical examination leverages any available information (possibly limited by the assessment scope, e.g. classified/unclassified data, public/government information, etc.) to develop an overview of cyber vulnerabilities and possible mitigations. Once the

device background has been detailed, and a thorough physical examination has been performed (if possible), a table listing all vulnerabilities and mitigations discovered should be created per the suggested format toward the end of Section .

Of particular importance are the application context (if known) and known incidents for the specific device and application space. Vulnerabilities should include common problems (like well-known issues with ICS technology and architectures) as well as specific risks (for example, a vulnerability to crashing when scanned from the network, or devices freezing if communication is lost). Countermeasures range from general cyber fixes (e.g. external security enhancement like VPN encapsulation) to specific cyber fixes (e.g. configuring onboard security measures).

7.3 Verification/Characterization Testing

The focus in this section should be on the approach used to decide which vulnerabilities and mitigations should be further investigated through testing, the test approaches that were used, and how to leverage existing ICS testing capabilities whenever possible. The process used to scope the vulnerabilities and mitigations selected to be tested, and the testing setup and testing process should be described. Key to this report section is how the prioritized APPs guide the testing process. Recall from Chapter 3 that each APP listed a set of potential approaches to test vulnerabilities discovered during the literature review and device physical examination ("verification"), but also to possibly uncover other vulnerabilities ("characterization," which will be further investigated in exploration testing). All approaches used should be noted, as well as any additional vulnerabilities that were uncovered.

7.3.1 Verification/Characterization Test Scoping

The test scoping process is described in Section 6.1, and there is an example in Appendix A. The information captured per the example should be reported here. The down-selected verification test scope will necessarily be augmented by characterization testing.

7.3.2 Verification/Characterization Test Setup and Background

This section describes how the testing was set up and the tools used to perform the testing (Section 7.2.3). The report should cover things that were considered and ultimately incorporated during the V/C testing such how the testing components (power supply, network/communication interfaces, etc.) were set up, access methods that could be used for device configuration, vendor software used for device access and management, etc.

The test requirements, success/fail indicators, and an overview of the test plan should be described. This section should also discuss the cyber security tools used during the testing, such as vendor software, penetration testing, firmware analysis, network analysis, packet manipulation, etc.

Recall from Section 7.2.3 and Chapter 3 that the FDAM has numerous suggestions for testing approaches within each of the APPs that cover both verification and characterization. The final device report should note which of these approaches were used.

7.3.3 Verification/Characterization Testing Results

For each vulnerability or mitigation tested, the following information should be captured (referencing Section 5.3):

- Vulnerability description
- References (public or private/government material)
- Dates of discovery and last analysis (approximate if necessary)
- APP association, and goals supported
- Vulnerability risk score and ranking
- Testing approaches used
- Safety/damage impact given the application (if known)
  - Potential unexpected or unpredictable behavior during OT&E
  - Potential damage to underlying physical system components
  - Denial of service for local automated or manual control
  - Denial of service to supervisory control
- Additional notes and analysis, if any
- Additional vulnerabilities discovered (to be tested later during Exploration testing)

7.4 Exploration Testing

The focus of exploration testing is to further develop vulnerabilities/mitigations discovered as a result of completing V/C testing. Test procedures are similar to V/C testing. The only difference is that fewer vulnerability category types (only V3 and V4, described in Chapter 2) are covered because the goal of exploration testing is to test specific vulnerabilities. The vulnerabilities are scored and listed from lowest to highest score, along with their risk score ranking and vulnerability category.

The test scoping process is similar to the one used for V/C tests, and again the example in Appendix A is useful. Note that an exploration testing background and set up section may not be required, as it is expected to be similar to the V/C configuration (Section 6.3). The results will also be tabulated in the summary format, with the only exception being that all results should cover only vulnerabilities specific to the device or application (V3/V4).

7.5 Device Report Summary

The summary of the FDAM work for the specific device is critical to communicating the results. The first section covers the conclusions. A good approach would be to cover vulnerabilities, mitigations, and operational test safety concerns separately. One table will summarize the vulnerabilities, including the unique identifier (like the example ones in Section 5.3 or Appendix Aex), risk score and ranking, and whether they were tested. A second table includes mitigations, which vulnerabilities they would improve, if they were tested, and if they can be retrofitted to existing systems and designs (as some mitigations may need to be included at the design stage). Finally, the conclusions should include cover all of the potential OT&E test concerns, especially controls to avoid damage/safety risks, possibly as specific ROE.

Following the conclusions are the recommendations. Very likely, there will be broad caveats for deploying the tested field device and suggestions or requirements for further analysis. This section should also note the depth of analysis achieved for the current effort, and how additional work could logically build on the existing knowledge. If feasible, the discussion should leverage the vulnerability work to determine priority for risk mitigation. Also, the mitigations can be considered in terms of cost/benefit, and which are most likely to avoid unreasonable impacts to operations. Finally, the recommendations should also discuss what are other tests may be needed to fully characterize operational test safety.

References

[1] Director, Operational Test and Evaluation (DOT&E), "Cybersecurity Test and Evaluation Guidebook," Department of Defense (DOD), policy reference, July 2015.

[2] Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, "DOD Instruction 5000.02," DOD, Tech. Rep., January 2015.

[3] J. Michael Gilmore, "Procedures for Operational Test and Evaluation of Cybersecurity in Acquisition Programs," DOT&E, Washington, DC, memorandum, August 2014.

[4] Chief Information Officer, "DOD Cybersecurity Discipline," DOD, implementation plan, February 2016.

[5] Chief Information Officer, "Cybersecurity Activities Support to DOD Information Network Operations," DOD, policy instruction, March 2016.

[6] J. Mulder, M. Schwartz, M. Berg *et al.*, "Understanding Programmable Logic Controllers: Joint Control System Work (JCSW) LDRD Final Report," Sandia National Laboratories (SNL), Albuquerque, NM, Sandia Report SAND2011-7009, September 2011, document is Unclassified // Official Use Only.

[7] J. Mulder, M. Schwartz, M. Berg *et al.*, "WeaselBoard: Zero-Day Exploit Detection for Programmable Logic Controllers," SNL, Albuquerque, NM, Sandia Report SAND2013-8274, October 2013.

[8] L. Blinde, (2016) "PFP Cybersecurity announces DARPA award for Leveraging the Analog Domain," http://intelligencecommunitynews.com/pfp-cybersecurity-announces-darpa-award-for-leveraging-the-analog-domain/, online; accessed 07 SEP 2016.

Appendix A

FDAM Test Scoping Example

A.1 Verification/Characterization Testing Selection Process

The literature review and physical examination will result in a list of vulnerabilities and mitigations. The following example shows how the FDAM approach can be used to narrow down what should be tested in both verification and exploration testing.

For this example, assume the following:

- The literature review and physical examination uncovered 13 vulnerabilities and 8 mitigations
- Vulnerabilities are labeled V_A through V_M
- Mitigations are labeled M_A through M_H

All vulnerabilities need to be scored and ranked using the risk score calculations in Chapter 4. Recall that the risk scores have been prioritized as shown in Table A.1.

Rating	Risk Score
None	0.0 - 0.1
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Table A.1: Qualitative severity rating scale

To date, there is no mitigation risk score calculation that would be applicable for FDAM. Consequently, mitigations are initially categorized by associating them with the vulnerabilities they can address. Table A.2 shows the list of vulnerabilities to be considered (V_A through V_M), their risk score, the ranking according to Table A.1, and the associated mitigations. Note that some mitigations might be applicable to one or more vulnerabilities.

Vulnerability	**Risk**	**Rank**	**Associated Mitigations**
V_A	2.2	Low	M_A, M_B
V_B	3.2	Low	M_C, M_F, M_H
V_C	3.6	Low	M_G
V_D	3.6	Low	M_B, M_F
V_E	3.9	Low	M_A, M_B
V_F	5.6	Medium	M_B, M_E
V_G	6.1	Medium	M_D
V_H	6.9	Medium	M_B, M_H
V_I	8.1	High	M_D
V_J	8.3	High	M_D
V_K	8.7	High	M_H
V_L	9.1	Critical	M_E, M_H
V_M	9.2	Critical	M_A, M_E

Table A.2: The vulnerabilities found in the literature review and physical examination are listed in order of their scores; associated ranks and mitigations are also listed

The next step in the process is to associate the vulnerabilities with the appropriate access/procedure pairs (APPs). Recall from Chapter 3 that the APPs are presented in order of priority. In Table A.3, the prioritized APP list is divided into bands. Within each band is a rule that determines the ranks of the vulnerabilities to be tested.

Ordered AC-APPs	**Vulnerability Test Rule**
A2-P4 (G1, G4) A2-P1 (G1–G5) A1-P4 (G1, G5) A1-P1 (G1–G5)	Test All
A5-P1 (G1–G5) A4-P4 (G1, G4) A4-P1 (G1–G5)	Test Critical, High, Medium
A2-P2 (G3) A2-P3 (G2)	Test Critical & High

Table A.3: Listing APPs according to priority, and then dividing into sections that determine which vulnerabilities should be tested in each band

Based on Table A.3, vulnerabilities are associated with the appropriate APP, resulting in a table similar to Table A.4. Note that throughout this example the APPs listing order always remains the same – the vulnerabilities and mitigations changed positions. For illustrative purposes, the associated mitigations column has been moved off to the right side.

Ordered AC-APPs	**Vulnerability**	**Risk**	**Rank**	*Test Vulnerability?*	*Vulnerability Test Rule*	**Associated Mitigations**
A2-P4 (G1, G4)	V_E	3.9	Low	Yes	Test All	M_A, M_B
A2-P1 (G1–G5)	V_F	5.6	Medium	Yes		M_B, M_E
A2-P1 (G1–G5)	V_L	9.1	Critical	Yes		M_E, M_H
A1-P4 (G1, G5)	V_J	8.3	High	Yes		M_D
A1-P4 (G1, G5)	V_M	9.2	Critical	Yes		M_A, M_E
A1-P1 (G1–G5)	V_D	3.6	Low	Yes		M_B, M_F
A5-P1 (G1–G5)	V_H	6.9	Medium	Yes	Test Critical, High, Medium	M_B, M_H
A4-P4 (G1, G4)	V_C	3.6	Low	No		M_G
A4-P4 (G1, G4)	V_I	8.1	High	Yes		M_D
A4-P1 (G1–G5)	V_K	8.7	High	Yes		M_H
A4-P1 (G1–G5)	V_B	3.2	Low	No		M_C, M_F, M_H
A2-P2 (G3)	V_G	6.1	Medium	No	Test Critical & High	M_D
A2-P3 (G2)	V_A	2.2	Low	No		M_A, M_B

Table A.4: The vulnerabilities (Table A.2) are associated with the appropriate APPs and test criteria (Table A.3) to determine testing requirements; in this example, V_A, V_B, V_C, and V_G will not be tested

Based on Table A.4, the final cut of vulnerabilities to be tested is shown in Table A.5. Note that mitigations are not narrowed down at this point, but will be addressed in the next step.

Ordered AC-APPs	Vulner-ability	Risk	Rank	Test Vulnerability?	Associated Mitigations
A2-P4 (G1, G4)	V_E	3.9	Low	Yes	M_A, M_B
A2-P1 (G1–G5)	V_F	5.6	Medium	Yes	M_B, M_E
A2-P1 (G1–G5)	V_L	9.1	Critical	Yes	M_E, M_H
A1-P4 (G1, G5)	V_J	8.3	High	Yes	M_D
A1-P4 (G1, G5)	V_M	9.2	Critical	Yes	M_A, M_E
A1-P1 (G1–G5)	V_D	3.6	Low	Yes	M_B, M_F
A5-P1 (G1–G5)	V_H	6.9	Medium	Yes	M_B, M_H
A4-P4 (G1, G4)	V_I	8.1	High	Yes	M_D
A4-P1 (G1–G5)	V_K	8.7	High	Yes	M_H

Table A.5: The final vulnerabilities test set is determined; mitigations have not been down selected yet

The following step is to narrow down the mitigations to be tested. Clearly, only mitigations associated with vulnerabilities that are to be tested are considered. Although mitigations follow the vulnerabilities throughout the scoping process, in the final analysis, each one needs to be looked at carefully to determine if it is achievable, effective, and worthwhile.

- **Achievable:** Assess if the mitigation is too expensive (e.g. having guards stationed at far more locations), or if it something that ca actually be done at this stage (i.e. conformal coating is a design-stage issue, not suitable for retrofitting). As another example, replacing legacy equipment with new simply to get a security advantage might not be financially reasonable. However, testing a patch and then applying it is probably achievable.
- **Effective:** Assess if the mitigation could cause more problems than it solves. As an example, programmable logic controllers (PLCs) could all be completely isolated from other networks, but that could affect important data or mission needs. However, it might be perfectly acceptable to prevent web access to the PLCs and not impact their effectiveness. Consider if the mitigation would only protect against a very small or unlikely vulnerability. Should resources be used elsewhere to implement mitigations that can resolve several more pressing vulnerabilities?

In this example, assume that after careful consideration only thee mitigations have been selected for testing – M_D, M_E, and M_H. Adding that information to Table A.5 gives the list of final vulnerability/mitigation test requirements as shown in Table A.6.

Ordered AC-APPs	Vulner-ability	Mitigations to Test
A2-P4 (G1, G4)	V_E	none
A2-P1 (G1–G5)	V_F	M_E
A2-P1 (G1–G5)	V_L	M_E, M_H
A1-P4 (G1, G5)	V_J	M_D
A1-P4 (G1, G5)	V_M	M_E
A1-P1 (G1–G5)	V_D	none
A5-P1 (G1–G5)	V_H	M_H
A4-P4 (G1, G4)	V_I	M_D
A4-P1 (G1–G5)	V_K	*none*

Table A.6: The final vulnerabilities and mitigations to be tested in the V/C phase

This information informs potential test approaches to test the requirements set shown in Table A.6 (the "Verification" part) as well as opportunities to discover new vulnerabilities (the "characterization" part). The latter will be down-scoped for exploration testing.

A.2 Exploration Phase Testing Selections

During V/C testing, new vulnerabilities and mitigations may be discovered that need to be tested. Recall from Chapter 2 that vulnerabilities fall into four categories:

- V1: Common problems/networking and communications
- V2: Common problems/architecture and management
- V3: Equipment-specific problems
- V4: Context- or application-specific problems

All of the vulnerabilities uncovered in the V/C testing should fall into the V3 and V4 categories. There should not be any new V1 or V2 found because these are general and should have been uncovered during the literature review or physical examination. The goal of exploration testing is to test equipment specific vulnerabilities (V3 and V4).

Using the same example above, assume that eight new vulnerabilities were uncovered during the V/C testing. They are labeled V_N through V_U. Just as in the V/C testing, the risk scores for each needs to be calculated and the vulnerabilities are listed from lowest to highest score, along with their risk score ranking. The only additional step is to also list each vulnerability's category (V1-V4), as shown in Table A.7. As mentioned, they should all fall under the V3 and V4 categories – listing the category is simply a precautionary measure to ensure that this is the case.

Vulnerability	**Risk**	**Rank**	**V3 or V4**	**Associated Mitigations**
V_N	2.3	Low	V3	M_B, M_I
V_O	3.2	Low	V3	M_F, M_J
V_P	3.6	Low	V4	M_M
V_Q	6.6	Medium	V3	M_B, M_J
V_R	6.9	Medium	V3	M_A, M_I
V_S	8.3	High	V4	M_B, M_M
V_T	8.4	High	V4	M_E
V_U	9.3	Critical	V3	M_K, M_L

Table A.7: Vulnerabilities and mitigations discovered in V/C testing are listed, along with associated vulnerability categories

Subsequent steps proceed similar to prior down-selections. Mixing the vulnerabilities and associated mitigations in Table A.7 into the APPs (Table A.3) results in Table A.8.

Ordered AC-APPs	**Vulnerability**	**Risk**	**Rank**	**V3 or V4**	*Test Vulnerability?*	*Vulnerability Test Rule*	**Associated Mitigations**
A2-P4 (G1, G4)	V_T	8.4	High	V4	Yes	Test All	M_E
A2-P1 (G1–G5)	V_U	9.3	Critical	V3	Yes		M_K, M_L
A1-P4 (G1, G5)	V_N	2.3	Low	V3	Yes		M_B, M_I
A1-P1 (G1–G5)	V_Q	6.6	Medium	V3	Yes		M_B, M_J
A5-P1 (G1–G5)	none	N/A	N/A	N/A	N/A	Test Critical, High, Medium	none
A4-P4 (G1, G4)	V_P	3.6	Low	V4	No		M_M
A4-P1 (G1–G5)	V_O	3.2	Low	V3	No		M_F, M_J
A2-P2 (G3)	V_S	8.3	High	V4	No	Test Critical & High	M_B, M_M
A2-P3 (G2)	V_R	6.9	Medium	V3	No		M_A, M_I

Table A.8: Vulnerabilities and mitigations found during V/C testing will be scoped down and tested in exploration testing

According to the prior table, it is clear to see that the first four vulnerabilities will be tested, along with the second-to-last entry. In this example, assume also that all the associated mitigations are achievable and effective, and will also be tested. Thus, the final set of vulnerabilities and mitigations to be tested are listed in Table A.9.

Ordered AC-APPs	**Vulner-ability**	**Mitigations to Test**
A2-P4 (G1, G4)	V_T	M_E
A2-P1 (G1–G5)	V_U	M_K, M_L
A1-P4 (G1, G5)	V_N	M_B, M_I
A1-P1 (G1–G5)	V_Q	M_B, M_J
A2-P2 (G3)	V_S	M_B, M_M

Table A.9: The final vulnerabilities and mitigations to be tested in the exploratory phase

Appendix B

Revision History

Version	Date	Time	Name	Notes
0.1	20150810	1529	JES	Initial version
0.2	20150821	1428	JES	Improved tools and data sources
0.3	20150825	1327	JES	Modified framework analysis
0.4	20150924	1327	JES	Added sections on prioritization
0.5	20151019	1529	JES	Split into two volumes and reorganized
0.6	20151119	1004	JES	Integrated Q1 comments
0.7	20160111	0814	JES	Reorganized material in Volume I
0.8	20160126	1527	JES	Joined volumes (Vol. II is a “cheat sheet” appendix)
0.9	20160217	1655	JES	Completed “first feedback” circulation draft
0.9.1	20160322	1338	JES	Completed “team review” circulation draft
0.9.2	20160602	1612	JES	Integrated “team review” comments
0.9.3	20160627	1603	JES	Completed assessment process section
0.9.8	20160921	1409	JAS	Completed major revision
0.9.9	20160929	1200	JES	Finalized 2nd draft
1.0	20170309	1602	JES	Finalized first version

Appendix C

Contact Information

Name	Organization
Jason Stamp *SNL Technical Lead*	Sandia National Laboratories P.O. Box 5800 Albuquerque, NM 87185-0671 jestamp@sandia.gov
Nick Pattengale *SNL Project Manager*	Sandia National Laboratories P.O. Box 5800 Albuquerque, NM 87185-0671 ndpatte@sandia.gov
Zach Benz *SNL Program Manager*	Sandia National Laboratories P.O. Box 5800 Albuquerque, NM 87185-0671 zobenz@sandia.gov
Steve Gates *DOT&E Program Manager*	Office of the Director, Operational Test and Evaluation 4800 Mark Center Drive Suite 10E12 Alexandria, VA 22311 stephen.m.gates.civ@mail.mil

Report Distribution

1 Steve Gates
Office of the Director, Operational Test and Evaluation
4800 Mark Center Drive Suite 10E12
Alexandria, Virginia 22311

1 John Burns
Office of the Director, Operational Test and Evaluation
4800 Mark Center Drive Suite 10E12
Alexandria, Virginia 22311

1 MS0671 Zach Benz, 5624
1 MS0671 Chris Davis, 5624
1 MS0671 Jennifer Depoy, 5628
1 MS0671 Dan Fay, 5627
1 MS0671 Mitch Martin, 5627
1 MS0671 Nick Pattengale, 5624
1 MS0671 Neeta Rattan, 5623
1 MS0671 Jason Stamp, 5623
1 MS0757 Tricia Schulz, 6613
1 MS0899 RIM-Reports Management, 9532 (electronic copy)

Sandia National Laboratories